Treebeard: A Scalable and Fault Tolerant ORAM Datastore

Amin Setayesh, Cheran Mahalingam, Emily Chen, and Sujaya Maiyya
University of Waterloo

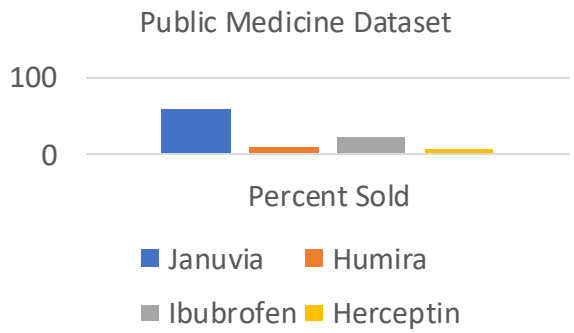
The Threat of Access Pattern Leakage

Encryption is not sufficient for data privacy. [1, 2, 3]

Adversary matches access patterns to public stats

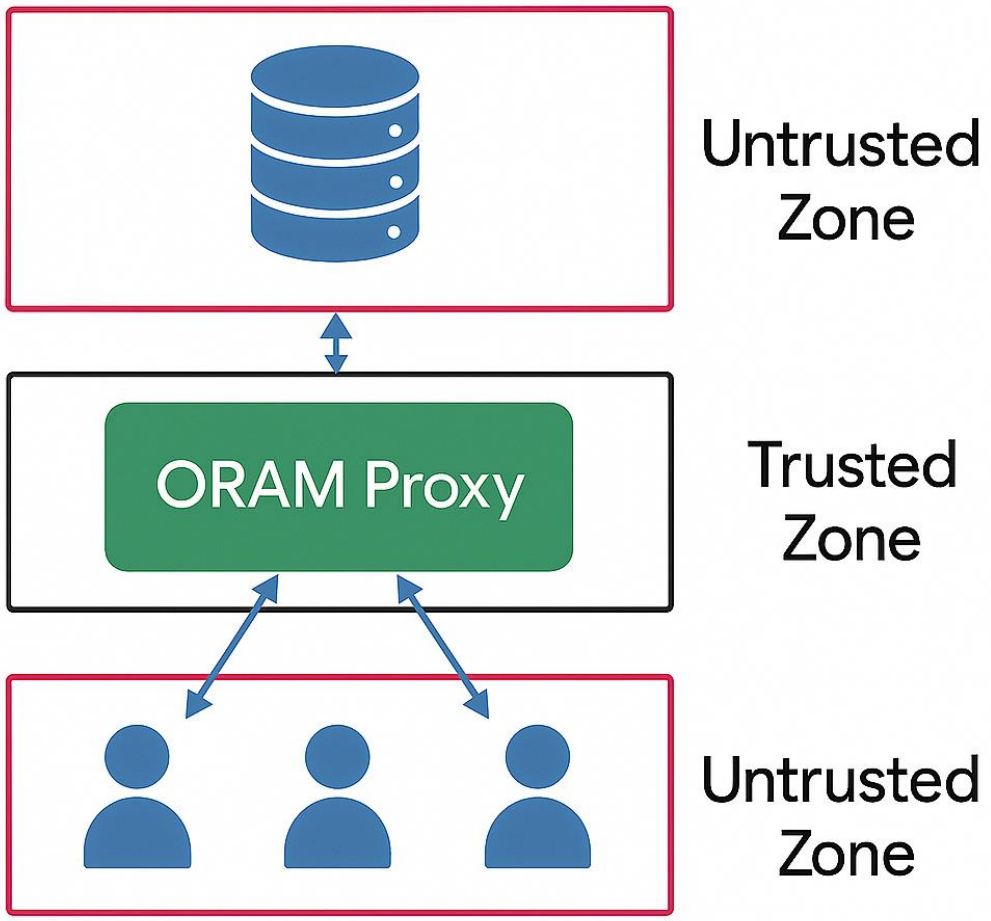
ID	Medicine
1	Januvia
2	Humira
3	Ibuprofen
4	Herceptin

ID	Medicine	
F4A	DSE…23	7
2SD	AEF…SO	1
D45	CME…UE	4
X12	GDE…1D	2



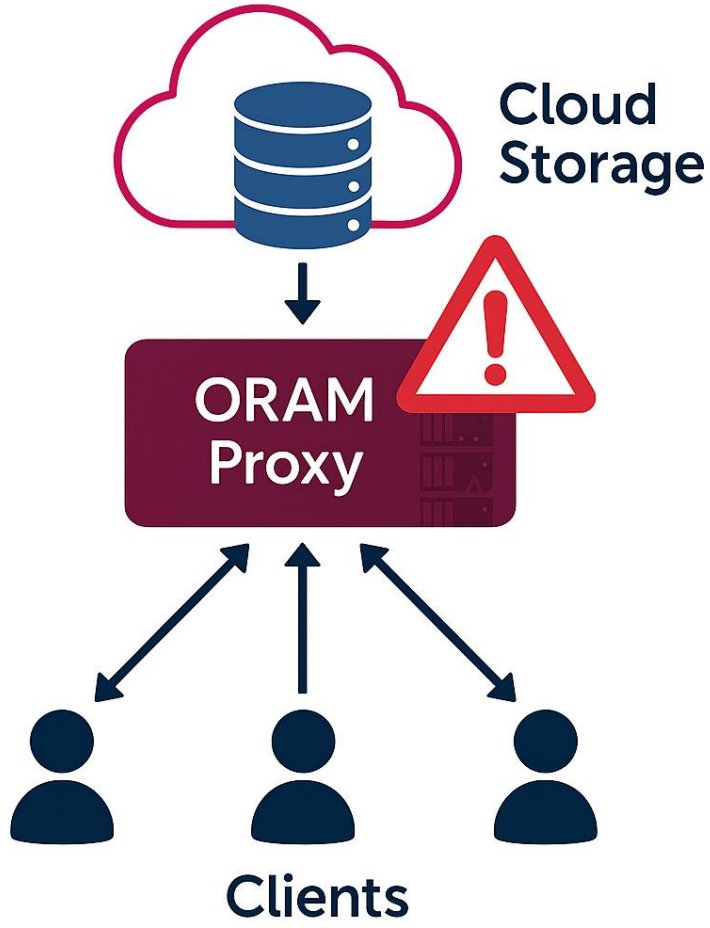
Oblivious RAM (ORAM)

- ORAM makes all data accesses look random [4]
- ORAM hides the following from the cloud provider:
  - Which object was accessed by a user
  - Which object was last accessed
  - Access pattern (skewed vs. uniform)
  - Whether a client reads or writes data



Problems with Current ORAM Systems

- **Centralized proxy**
- **Single point of failure.**
- **Require specialized hardware.**



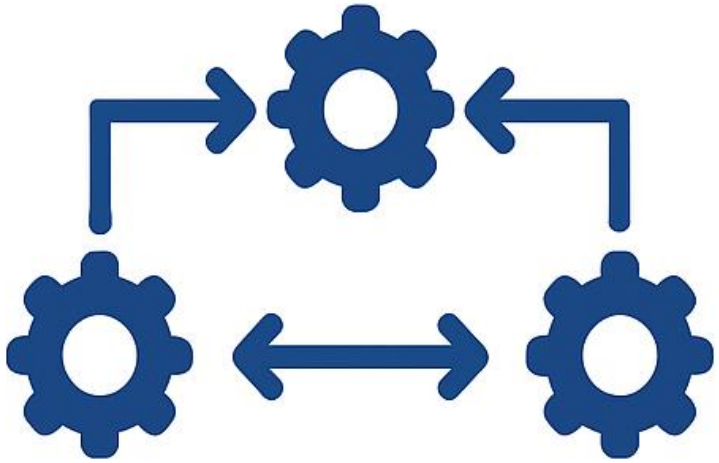
ORAM Challenges

Achieving scalability, fault tolerance, and parallelism while preserving security is fundamentally hard.

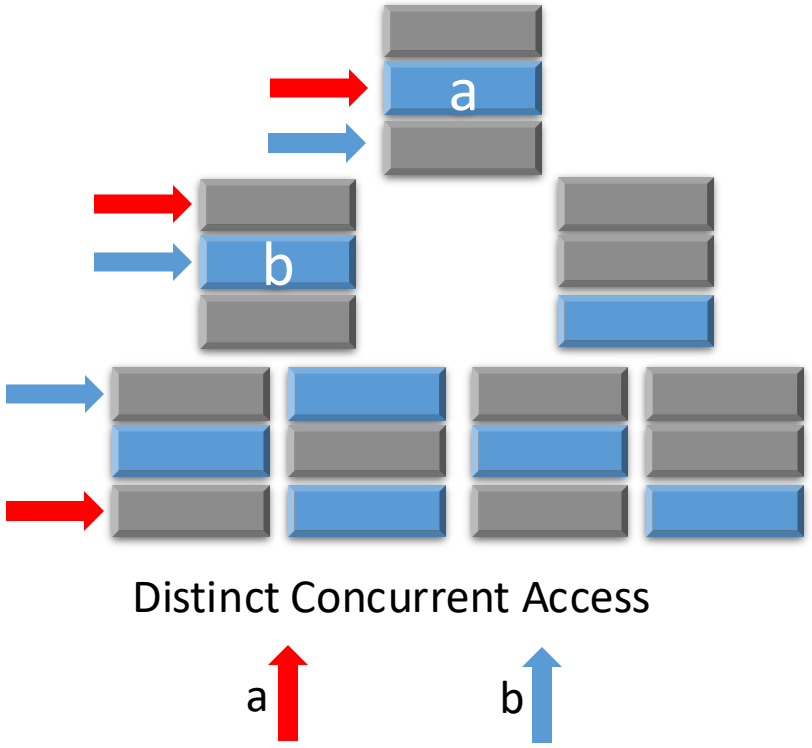
Scalability

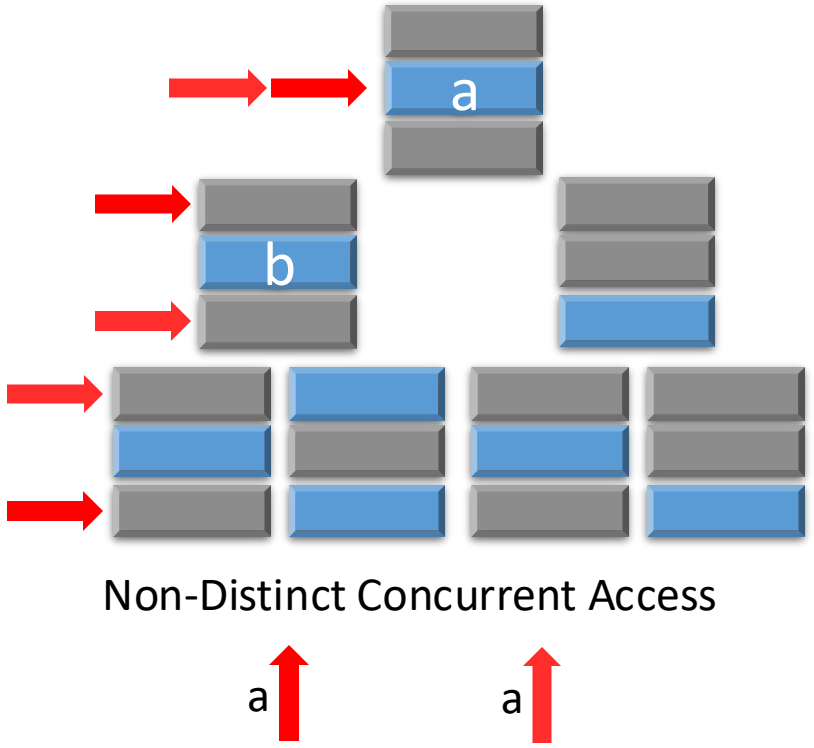
Fault Tolerance

Parallelism



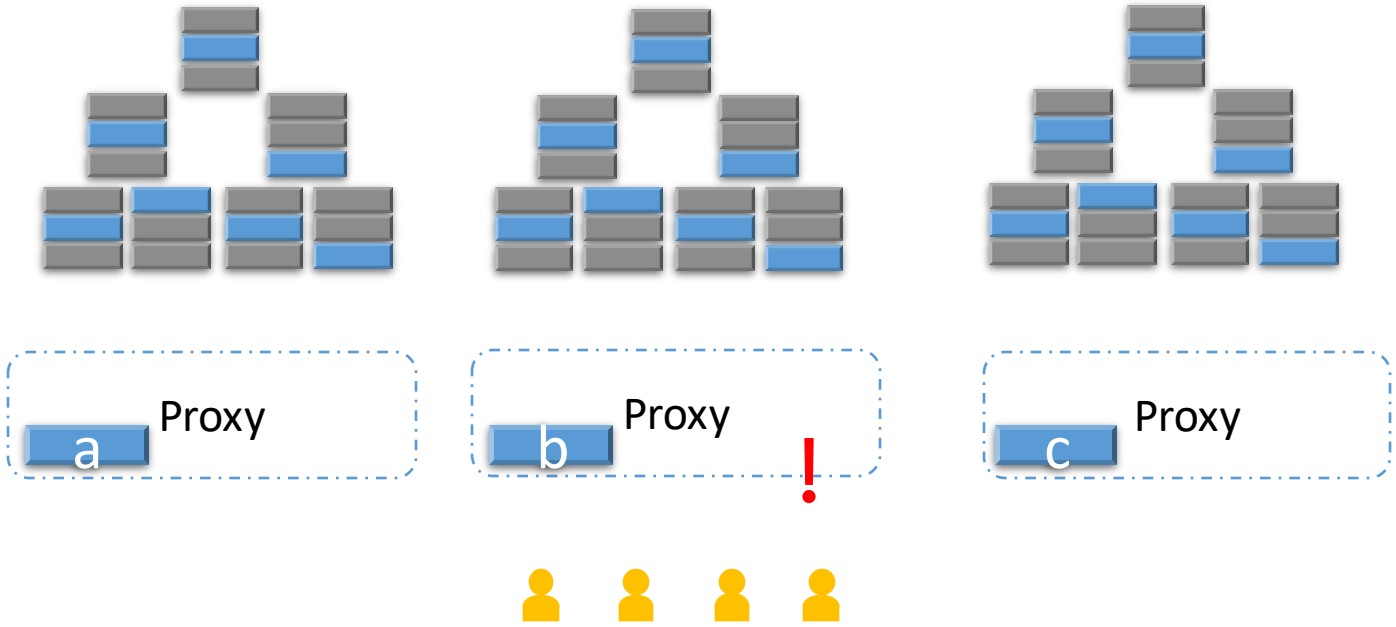
Challenges – Parallel & Secure



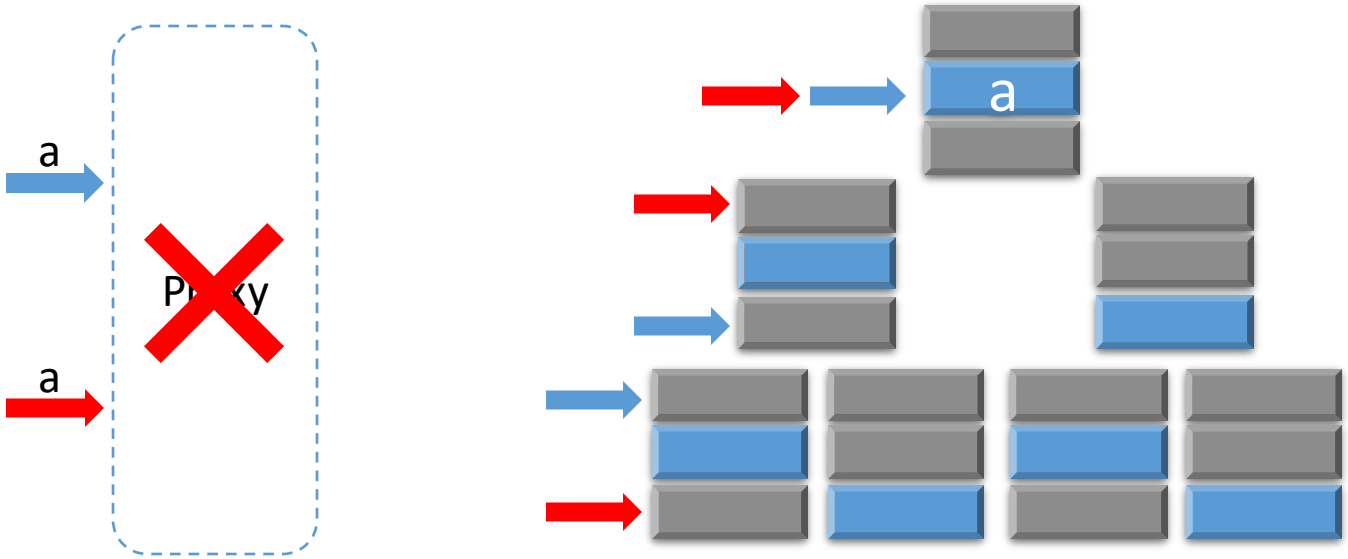


Challenges – Scalable & Secure

- **Static mappings**
- **Inter-proxy coordination**

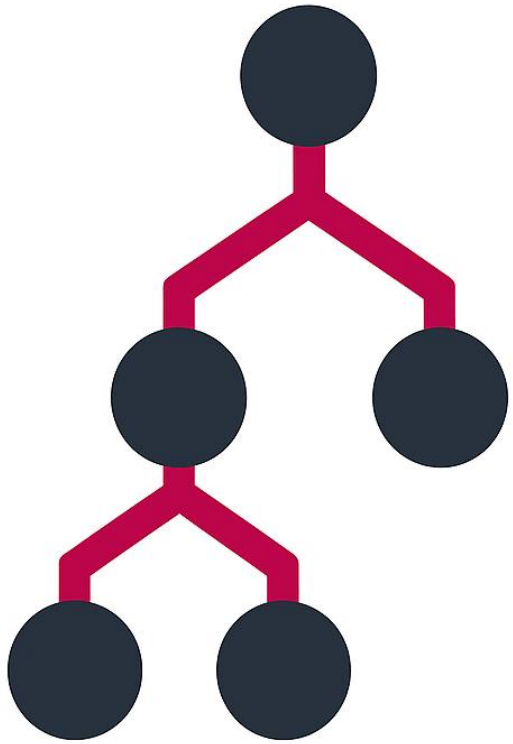


Challenges – Fault-tolerant & Secure



Treebeard

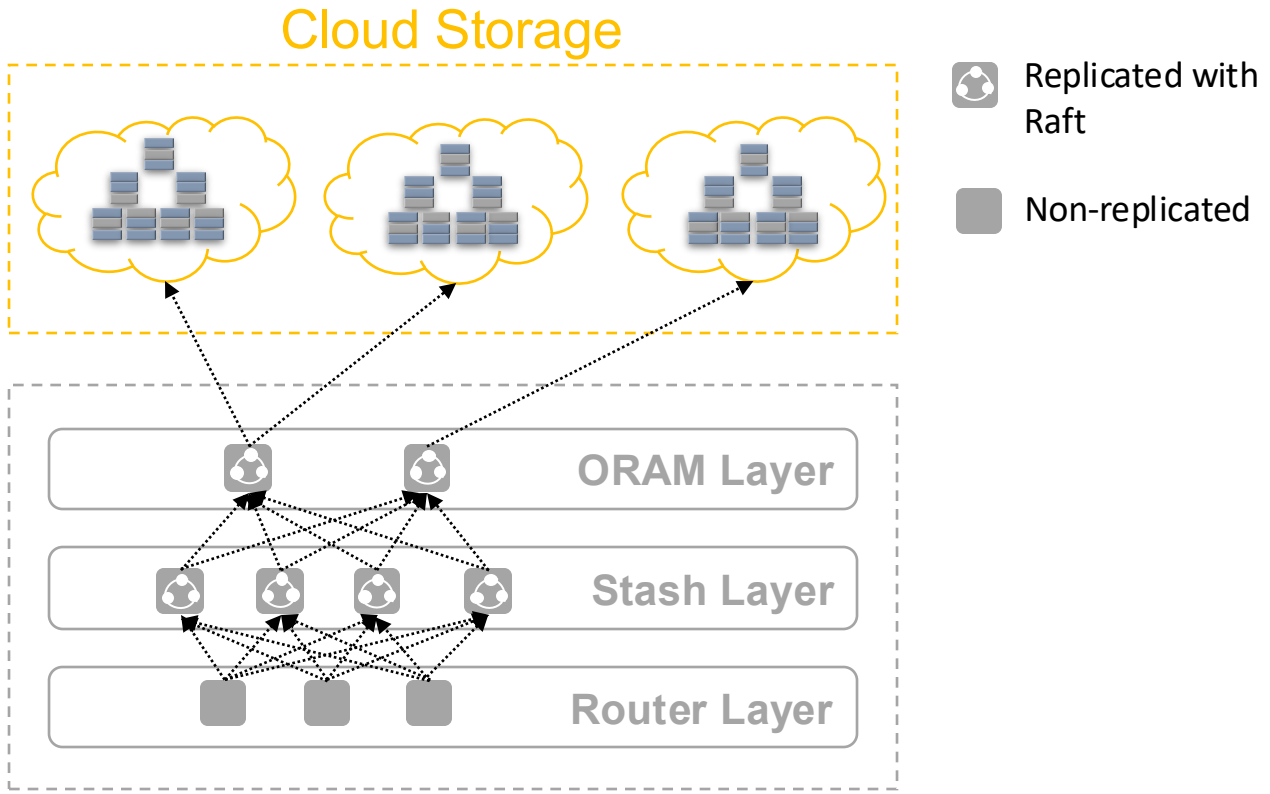
A modular ORAM system
designed for scale,
resilience, and performance.



In Submission to USENIX Security 2025

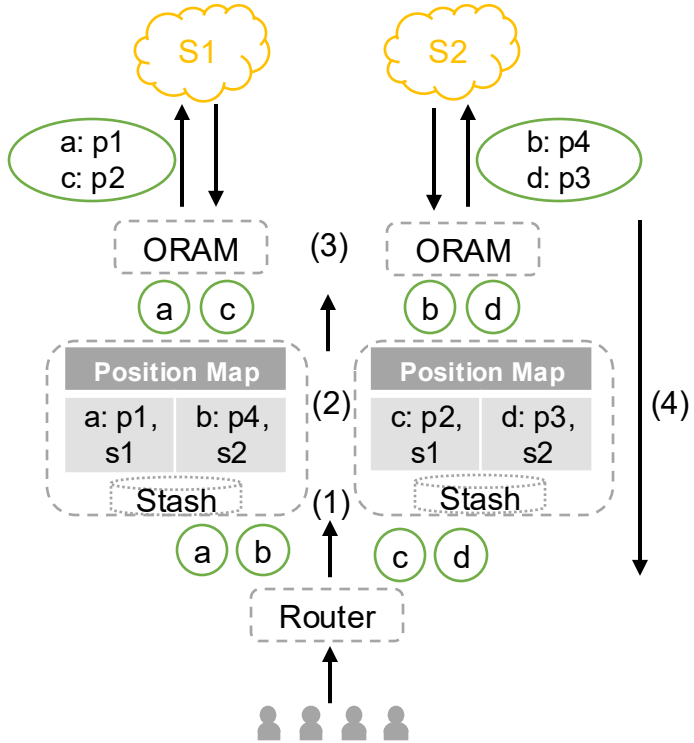
Treebeard Overview

- Modular design
- Parallelism and horizontal scaling
- Layer scaling & fault tolerance



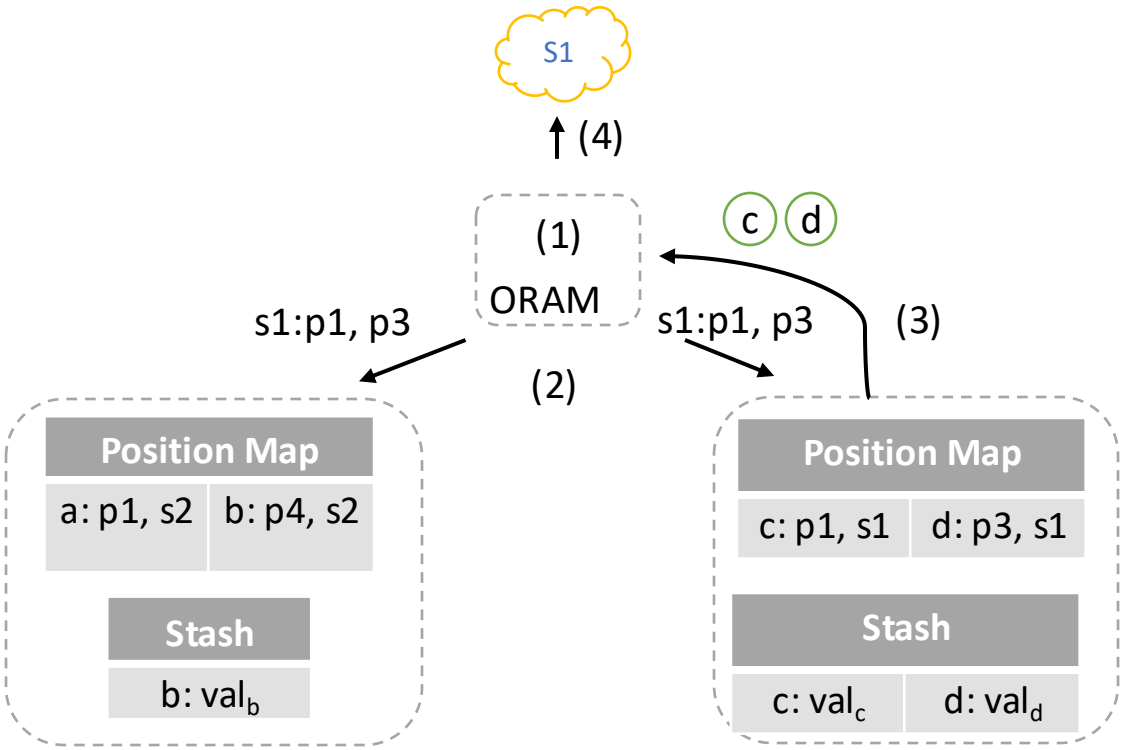
Treebeard Overview – Read Operation

1. Router request batching
2. Stash process forwarding
3. Storage access
4. Response



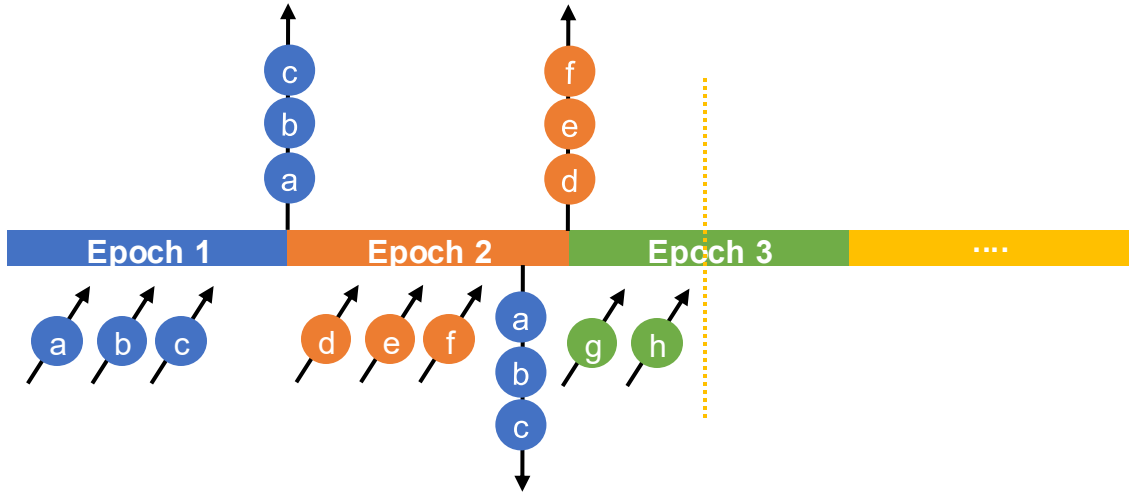
Treebeard Overview – Eviction

1. Eviction initiation
2. Block request from Stash layer
3. Stash response
4. Storage writes

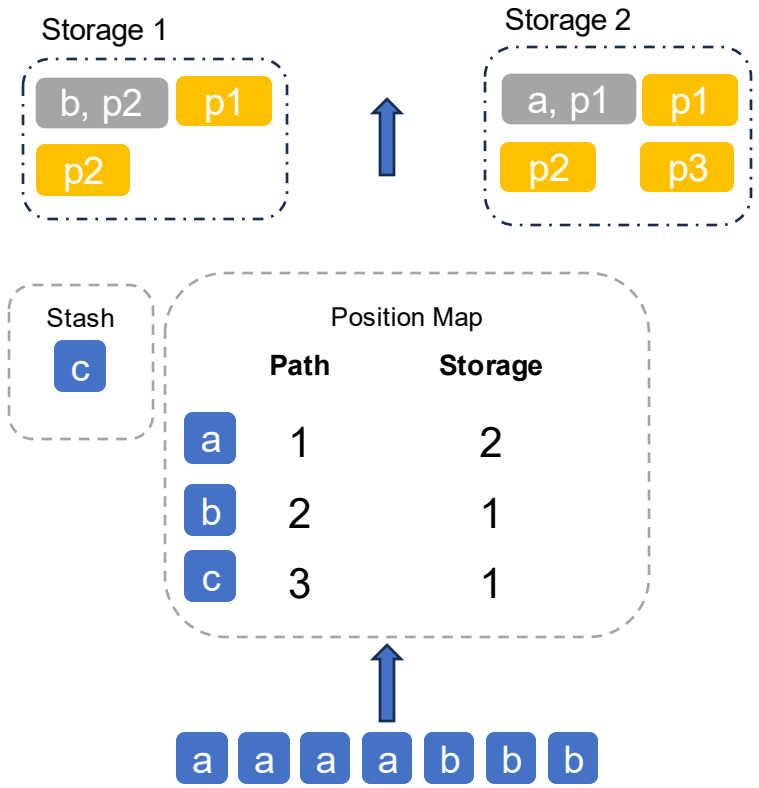


Router Layer - Overview

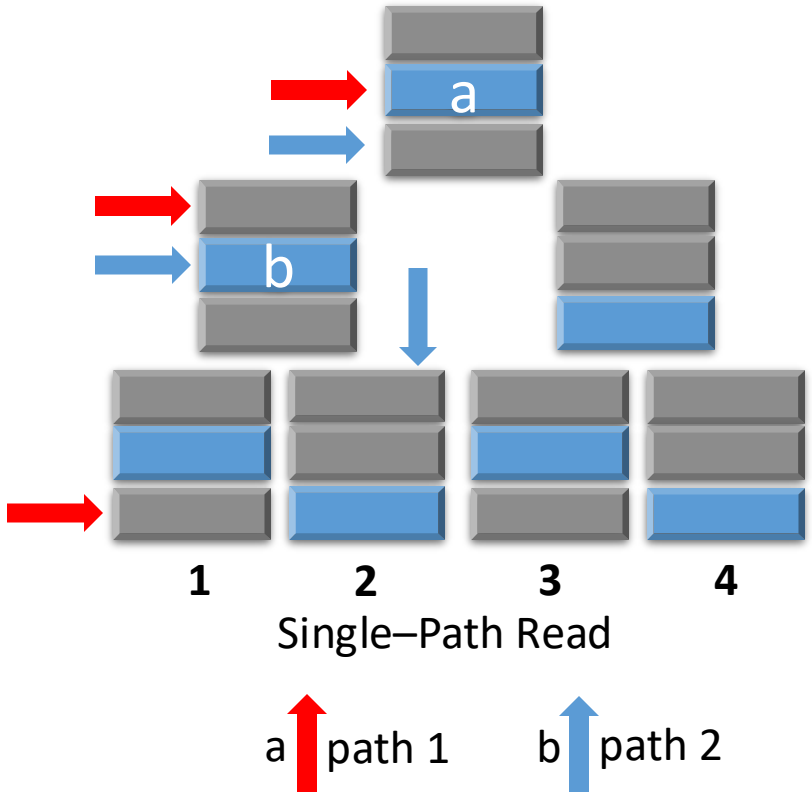
- Stateless
- Epoch-based batching
- Hash-based mapping to Stash nodes
- No replication

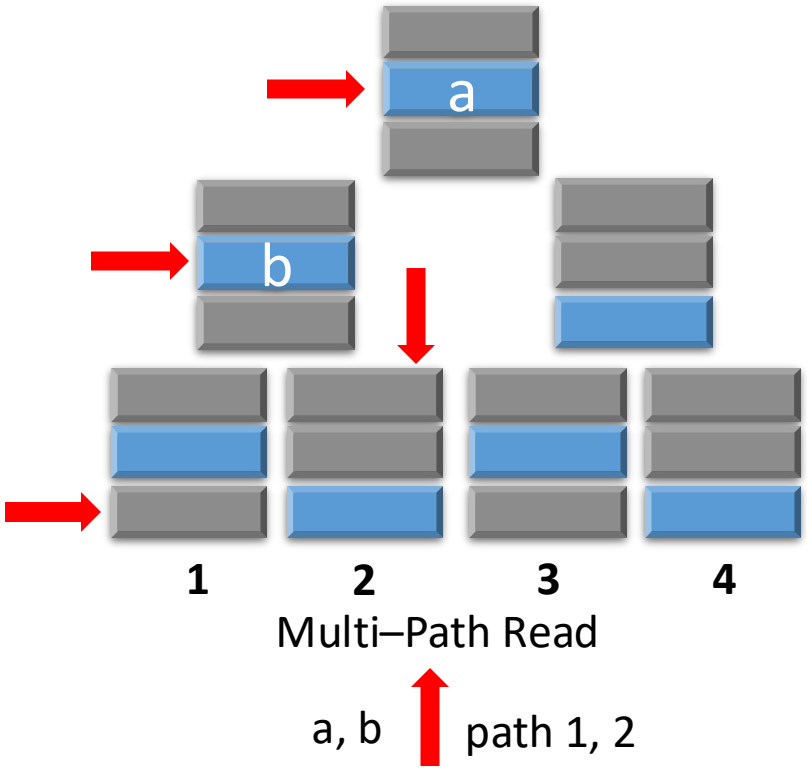


Stash Layer – Read Operation



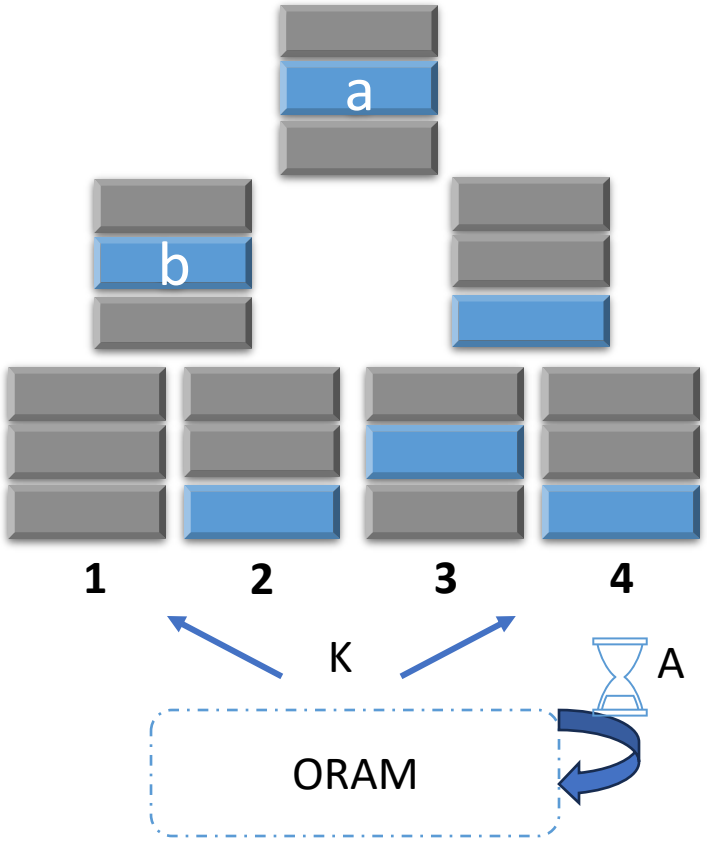
ORAM Layer – Multi Path Reads





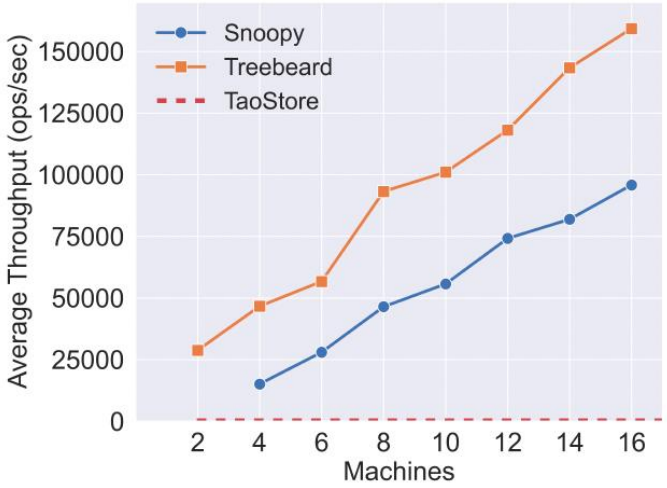
Multi-path eviction

- A: Eviction rate
- K: Path count

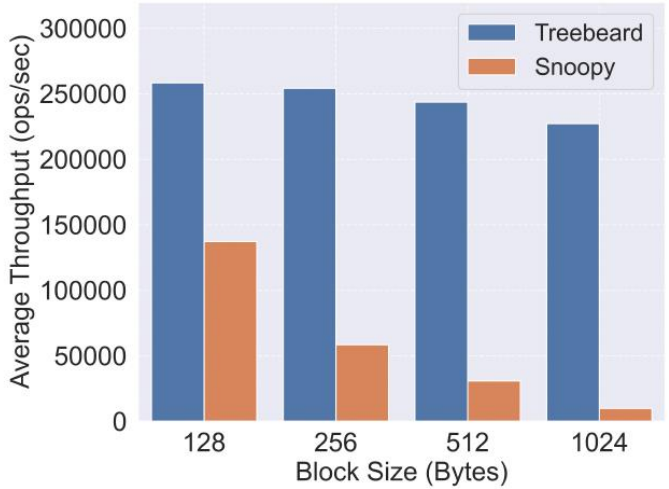


Implementation & Evaluation

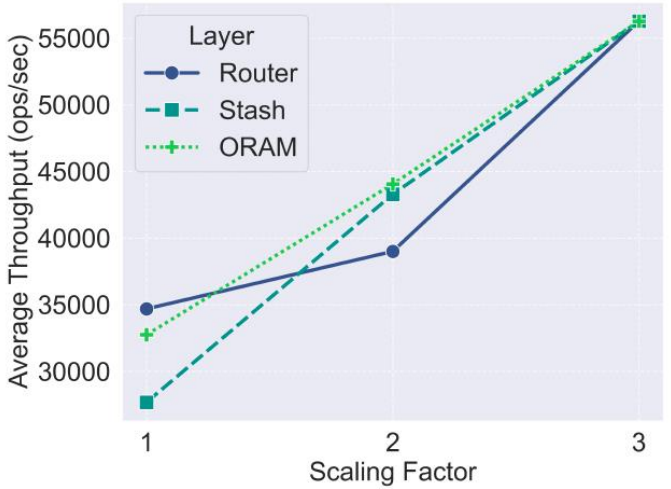
Scalability Analysis



(a) Scaling Throughput

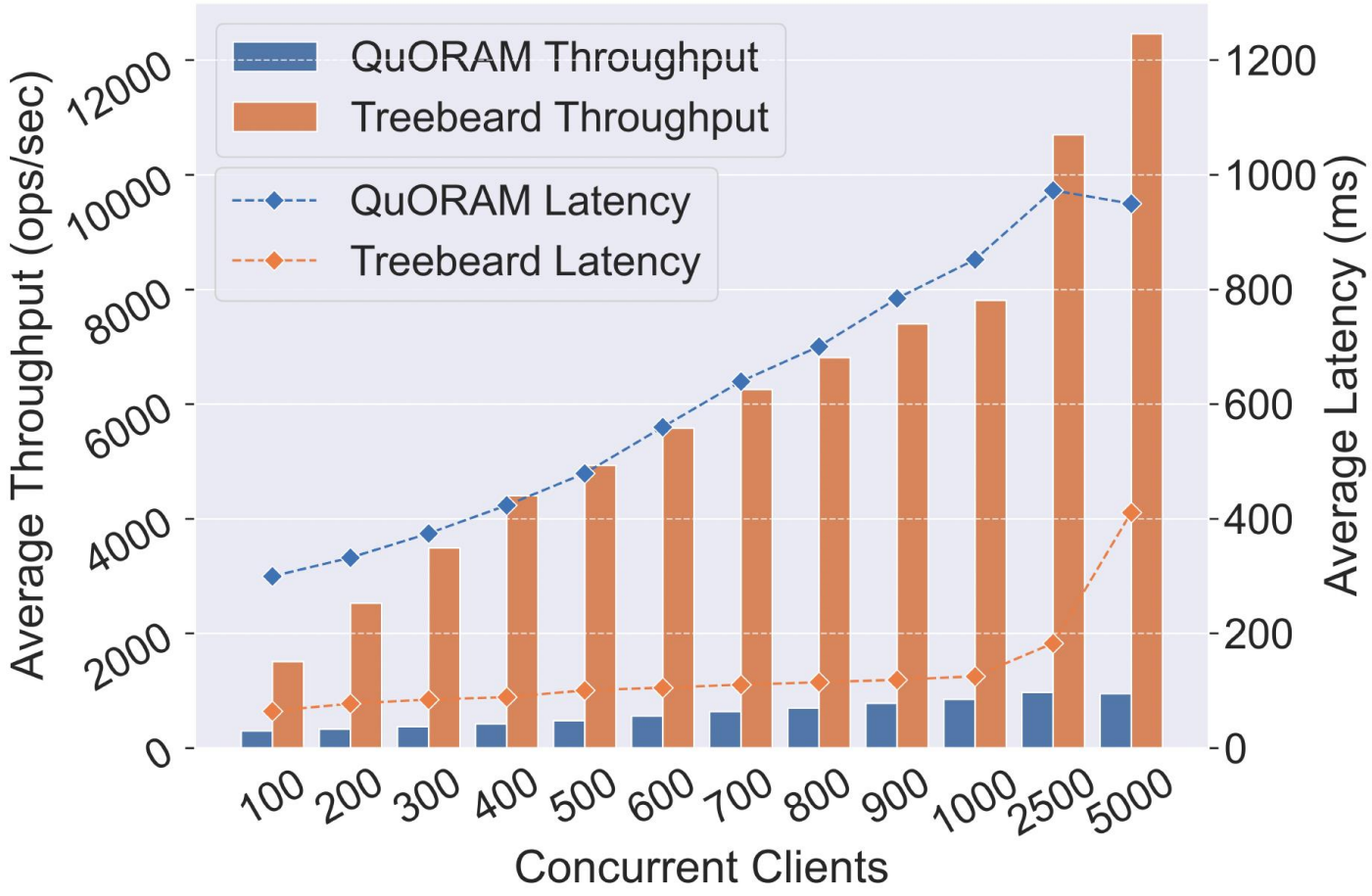


(b) Impact of Block Size

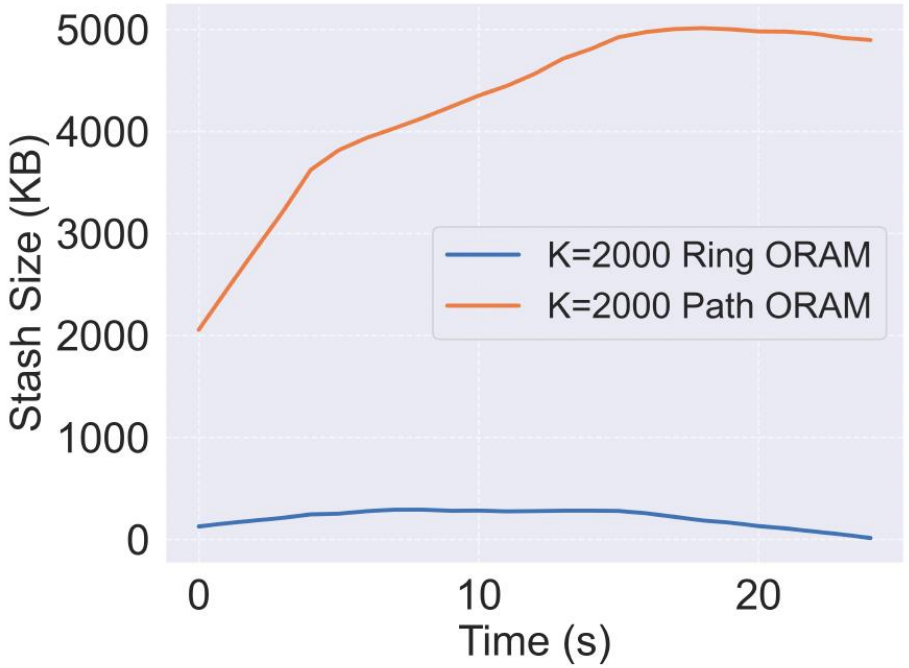


(c) Impact of Scaling Each Layer

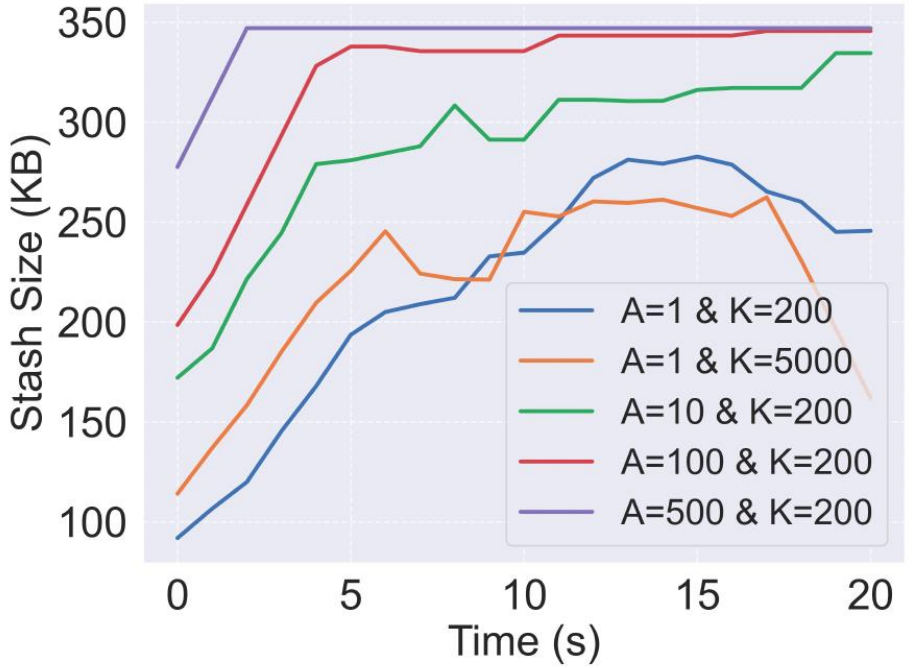
Fault Tolerance Analysis



Stash Experiments



(a) Treebeard with Path ORAM vs. Ring ORAM stash size



(b) Impact of A & k on Treebeard's stash size

References

[1] Mohammad Saiful Islam, Mehmet Kuzu, and Murat Kantarcioglu. Access pattern disclosure on searchable encryption: ramification, attack and mitigation. In 19th Annual Network and Distributed System Security Symposium (NDSS), 2012.

[2] Paul Grubbs, Marie-Sarah Lacharit´e, Brice Minaud, and Kenneth G Paterson. Learning to reconstruct: Statistical learning theory and encrypted database attacks. In 2019 IEEE Symposium on Security and Privacy (SP), pages 1067–1083, 2019.

[3] Georgios Kellaris, George Kollios, Kobbi Nissim, and Adam O'neill. Generic attacks on secure outsourced databases. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS), pages 1329–1340, 2016.

[4] Oded Goldreich and Rafail Ostrovsky. Software protection and simulation on oblivious RAMs. J. ACM, 43(3):431–473, 1996.