Cryptology ePrint Archive: Report 2011/384

Multi-Server Oblivious RAM

Steve Lu and Rafail Ostrovsky

Abstract: Secure, cloud-based storage has attracted considerable attention, both from a theoretical and practical perspective. To access data in a private manner, "Oblivious RAM" techniques have been employed in the past to store and retrieve data from an untrusted storage in such a way that no information regarding "usage statistics" is revealed to a computationally bounded adversary monitoring it.

In this paper, we introduce a very natural setting where there are two or more non-communicating servers (such as Yahoo cloud and Amazon AWS, which, in light of competitive nature are unlikely to collaborate) and ask if more efficient Oblivious RAM simulations are possible than what are currently known for a single server with sublinear work at each server. Surprisingly, we show the following:

To support n reads and writes, our two-server oblivious RAM protocol requires $O(n)$ memory for the servers, $O(1)$ memory for the client, and $O(\log n)$ amortized read/write overhead for data access. The constants in the big-O notation are tiny, and we show that the storage and data access overhead of our solution concretely compares favorably to the state-of-the-art single-server schemes. Furthermore, these parameters asymptotically match the lower bound for any single-server solution and point out the fascinating possibility that two-server solutions may even beat the single-server lower bound.

In addition, our protocol enjoys an important feature from a practical perspective as well. At the heart of almost all previous single-server Oblivious RAM solutions, a crucial but inefficient process known as oblivious sorting was required. In our two-server model, we describe a novel technique to bypass oblivious sorting, and show how this can be carefully blended with existing techniques to attain a more practical Oblivious RAM protocol in comparison to all prior work.

Category / Keywords: cryptographic protocols / Oblivious RAM, Cloud Computing, Multi-Server Model, Software Protection

Date: received 14 Jul 2011

Contact author: steve at stealthsoftwareinc com