Authenticated private information retrieval

Simone Colombo, *EPFL;* Kirill Nikitin, *Cornell Tech;* Henry Corrigan-Gibbs, *MIT;*
David J. Wu, *UT Austin;* Bryan Ford, *EPFL*

https://www.usenix.org/conference/usenixsecurity23/presentation/colombo

This paper is included in the Proceedings of the
32nd USENIX Security Symposium.

August 9–11, 2023 • Anaheim, CA, USA

978-1-939133-37-3

Open access to the Proceedings of the
32nd USENIX Security Symposium
is sponsored by USENIX.

Authenticated private information retrieval

Simone Colombo *EPFL*
Kirill Nikitin *Cornell Tech*
Henry Corrigan-Gibbs *MIT*
David J. Wu *UT Austin*
Bryan Ford *EPFL*

Abstract. This paper introduces protocols for *authenticated* private information retrieval. These schemes enable a client to fetch a record from a remote database server such that (a) the server does not learn which record the client reads, and (b) the client either obtains the "authentic" record or detects server misbehavior and safely aborts. Both properties are crucial for many applications. Standard private-information-retrieval schemes either do not ensure this form of output authenticity, or they require multiple database replicas with an honest majority. In contrast, we offer multi-server schemes that protect security as long as at least one server is honest. Moreover, if the client can obtain a short digest of the database out of band, then our schemes require only a single server. Performing an authenticated private PGP-public-key lookup on an OpenPGP key server's database of 3.5 million keys (3 GiB), using two non-colluding servers, takes under 1.2 core-seconds of computation, essentially matching the time taken by unauthenticated private information retrieval. Our authenticated single-server schemes are 30-100× more costly than state-of-the-art unauthenticated single-server schemes, though they achieve incomparably stronger integrity properties.

1 Introduction

Private information retrieval (PIR) [29] enables a client to fetch a record from a database while hiding from the database server(s) which specific record(s) the client retrieves. PIR has numerous privacy-protection uses, such as in metadata-private messaging [5, 6], certificate transparency [62, 80], video streaming [50], password-breach alerting [4, 59, 83], retrieval of security updates [22], public-key directories [63], and private SQL-like queries on public data [72, 88].

Most PIR protocols, however, do not ensure data authenticity in the presence of malicious servers. In many multi-server PIR schemes [17, 29], a single adversarial server can flip any subset of bits in the client's recovered output. In all single-server PIR schemes we know of (c.f., [1, 4, 5, 18, 20, 31, 36, 45, 51, 56, 61, 65, 70, 74, 76] for a non-exhaustive list), a malicious server can choose the exact output that the client will receive by substituting all the database records with a chosen record before processing the client's request. In applications where data integrity matters, such as a PGP public-key directory, unauthenticated PIR is inadequate.

This paper introduces *authenticated private information retrieval*, which augments the standard privacy properties of classic PIR with strong authenticity guarantees. In the multi-server setting, we propose authenticated-PIR schemes for:

- *Point queries*, in which a client wants to fetch a particular database record. For example, "What is the public key for `user@usenix.org`?"
- *Predicate queries*, where a client wants to apply an aggregation operator – such as `COUNT`, `SUM`, or `AVG` – to all records matching a predicate. For example, "How many keys are registered for email addresses ending in `@usenix.org`?"

Our corresponding authenticated-PIR schemes guarantee integrity in the *anytrust* model [90]: as long as at least one of the PIR servers is honest. In contrast, prior work that deals with malicious or faulty PIR servers in the multi-server setting either requires a majority or supermajority of servers to be honest [11, 12, 38, 48] or requires expensive public-key cryptography operations [94]. Our schemes use only fast symmetric-key cryptography in the multi-server setting.

In the single-server setting, we offer authenticated-PIR schemes for point queries which provide authentication as long as the client can obtain a short digest of the database via out-of-band means (Fig. 1). Prior work for the single-server setting [56, 89, 95] ensures only that the server truthfully answers the query with respect to *some* database—not necessarily the database the client queried. Table 2 summarizes prior work and Section 8 gives the complete discussion.

New definitions. Our first contribution is a new definition of integrity for private information retrieval. In our multi-server PIR schemes, a client communicates with several database servers, and client privacy holds as long as at least one server is honest. In this multi-server setting, we say that a PIR scheme satisfies integrity if, whenever the client accepts the servers' answers, the client's output is consistent with an honest server's view of the database.

Defining integrity in the single-server setting is more tricky: If the single database server is malicious, who is to say what the "right" database is? Our approach assumes that the client can obtain a short digest of the database via some out-of-band means. A single-server PIR protocol satisfies integrity if the client accepts the protocol's output only if the output is consistent with the database that the digest represents. In some applications of PIR, the client could obtain this database digest via a gossip mechanism, as in CONIKS [64], or from a collective authority [81], or from a signature-producing blockchain [71]. In other applications of PIR such as video streaming [50], a

The full version of this paper is available at https://eprint.iacr.org/2023/297.