How to Authenticate any Data Structure

Andrew Miller, Michael Hicks, Jonathan Katz,Elaine Shi
University of Maryland
{amiller,mwh,jkatz,elaine}@cs.umd.edu

ABSTRACT

Hash-based Authenticated Data Structures (ADS) are a classic technique in cryptography (beginning with Merkle's authenticated binary trees), and used widely in computer security applications (including BitTorrent, Amazon Dynamo, and Bitcoin, just to name a few). An ADS allows a client to outsource storage of a data structure to an untrusted server; the client can efficiently query the data structure remotely (without having to fetch all the data) and can verify that the query result is correct. We give a thoroughly generic treatment of this technique using programming language theory: from any ordinary (pure functional) data structure definition, we obtain a corresponding authenticated data structure protocol [1]. This also leads to a practical implementation of our language, **λ•**, based on OCaml: our compiler takes as input an ordinary data structure definition (annotated with the "auth" type operator, •, as well as coercions **auth** and **unauth**), and outputs a correct-by-construction protocol implementation, with performance comparable to hand-optimized code.

To illustrate by way of example, the following **λ•** code defines an authenticated binary-search-tree data type:

```
type tree = Tip | Bin of (•tree × Int × •tree)
```

and the following code defines a lookup query:

```
lookup :: • tree → Int → bool
lookup tree x = case unauth tree of
      | Tip → false
      | Bin(l, x, r) | x == y → true
                     | x < y → lookup l x
                     | x > y → lookup r x
```

BODY

*In our new language, **λ•**, every data structure has an authenticated "merkle-ized" variant, safe to store on untrusted servers.*

REFERENCES

[1] Andrew Miller, Michael Hicks, Jonathan Katz, and Elaine Shi. *Authenticated Data Structures, Generically.* In Proceedings of the 41st annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages. ACM, 2014.

Volume 3 of Tiny Transactions on Computer Science