zkproof 2020: Bulletproofs and ZKPs for Blockchains

[authors anonymized]

2 February 2020

The following is an extended abstract for a talk at the ZKProof workshop in April 2020. It falls more into the SoK category rather than the proposal category, but it is a talk without an associated paper. The authors hope this is a useful submission to the workshop.

Bulletproofs[1] were developed in 2018 primarily as a zero-knowledge rangeproof for use in Confidential Transactions[2], a cryptosystem which replaces amounts in blockchain transactions with Pedersen commitments, effectively hiding them from verifiers. However, Bulletproofs are more general than rangeproofs, allowing the creation of zero-knowledge proofs of arbitrary arithmetic circuits.

In this talk we use Bulletproofs to illustrate the general design requirements of zero-knowledge proof schemes for certain blockchain applications such as rangeproofs. In particular, Bulletproofs have

- No trusted setup.
- No cryptographic assumptions beyond ECDL / random oracle, which are assumptions already used by every deployed blockchain
- Small proof sizes (logarithmic in circuit size with constants implying 1-2Kb proofs for any realistic circuit size)
- Tolerable proving and verification time, at least on PCs and where batch verification is possible (for example, when transactions have multiple similarly-sized proofs)

These attributes make Bulletproofs attractive for very small proofs, such as 64-bit rangeproofs (in fact, for such proofs the authors believe Bulletproofs have the fastest proving and verification time of any scheme in the literature, and the smallest proofs which have no trusted setup.

However, in extending beyond small rangeproofs to general circuits with thousands of multiplication gates (as would be needed, say, to replace input references with zero-knowledge lookups in large Merkle trees), the limitations of Bulletproofs become clear. When scaling to millions of gates (as would be needed to replace something like Bitcoin Script with a zero-knowledge proof system) Bulletproofs become completely intractable. Specifically,

- Verification time scales linearly with the size of circuits, resulting in multiple seconds to verifiy multi-1000-gate circuits, and hours to verify multi-million gate circuits (though batch verification gains us a factor of roughly 10 here). Ultimately a zero-knowledge proof scheme for advanced blockchain applications would need sublinear verification time to be useful.
- Proving time also scales linearly, and with worse constants than verification.
- Further, it is a logistical requirement that proving be doable with secure hardware (such as a Ledger or Trezor hardware wallet). Such hardware is typically extremely weak, a factor of 1000 or more slower than commodity PCs.

Having said this, we highlight some simple applications which can be implemented with very small circuits, for which Bulletproofs appear to be the optimal choice of zero-knowledge proof scheme.

[1] `https://crypto.stanford.edu/bulletproofs/`

[2] `https://link.springer.com/chapter/10.1007%2F978-3-662-58820-8_4`