Class 8: Mining

Schedule

Wednesday, September 23: Checkup 2 (was originally scheduled for Monday, September 21). See previous class notes for details on what is covered.
Friday, October 9: Problem Set 2 (moved from original deadline of October 2). Problem Set 2 will be posted later this week.
Monday, October 19: Midterm Exam

Reminders

You can subscribe to the course calendar. This has updated information on deadlines and office hours.

If you want to receive course website updates by email, you can subscribe to the RSS feed using a RSS reader or an emailing service like feedmyinbox.com or Blogtrottr. The feed address is http://bitcoin-class.org/index.xml.

We generally will avoid sending out emails to the class, and will assume you are observing the website closely.

Exploring Blocks

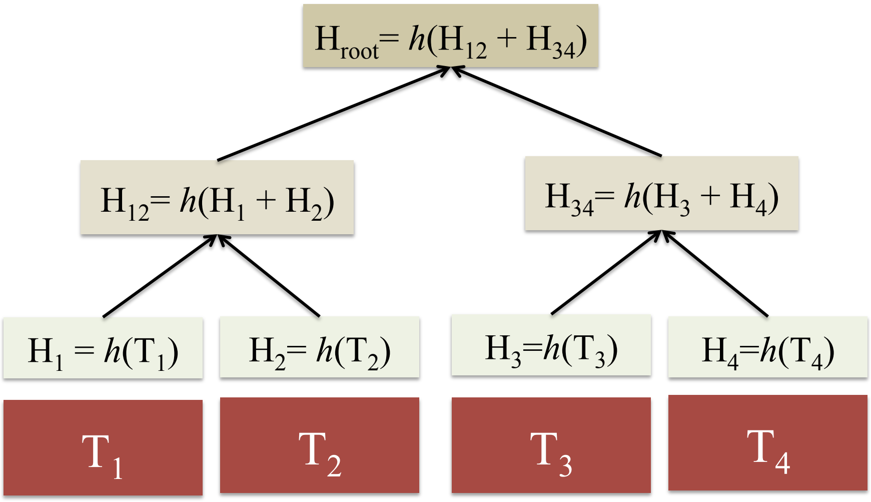
Label	Bytes	Description
`version`	4	Block version information
`prev_block`	32	Hash of the previous block
`merkle_root`	32	Hash of Merkle tree of all transactions
`timestamp`	4	When block was created (overflows in 2106)
`bits`	4	Difficulty target used for this block
`nonce`	4	Nonce found to generate this block

Merkle Trees

Ralph Merkle, *Publishing a New Idea*. Includes his cs244 project proposal ("Discussion: No, I am not joking.") and ACM rejection letter ("I am sorry to have to inform you that the paper is not in the main stream of present cryptography thinking and I would not recommend that it be published...").

https://github.com/btcsuite/btcd/blob/master/blockchain/merkle.go (some comments removed)

```
// HashMerkleBranches takes two hashes, treated as the left and right tree
// nodes, and returns the hash of their concatenation.
func HashMerkleBranches(left *btcwire.ShaHash, right *btcwire.ShaHash) *btcwire.ShaHash {
   var sha [btcwire.HashSize * 2]byte
   copy(sha[:btcwire.HashSize], left.Bytes())
   copy(sha[btcwire.HashSize:], right.Bytes())
   newSha, _ := btcwire.NewShaHash(btcwire.DoubleSha256(sha[:]))
   return newSha
}

func BuildMerkleTreeStore(transactions []*btcutil.Tx) []*btcwire.ShaHash {
    nextPoT := nextPowerOfTwo(len(transactions))
    arraySize := nextPoT*2 - 1
    merkles := make([]*btcwire.ShaHash, arraySize)

    // Create the base transaction shas and populate the array with them.
    for i, tx := range transactions { merkles[i] = tx.Sha() }

    // Start the array offset after the last transaction and adjusted to the
    // next power of two.
    offset := nextPoT
    for i := 0; i < arraySize-1; i += 2 {
        switch {
           case merkles[i] == nil:
              merkles[offset] = nil

           case merkles[i+1] == nil:
              newSha := HashMerkleBranches(merkles[i], merkles[i])
              merkles[offset] = newSha

           default:
              newSha := HashMerkleBranches(merkles[i], merkles[i+1])
              merkles[offset] = newSha
       }
       offset++
    }
    return merkles
}
```



What is needed to verify T_2 in H_{root}?

What must be recomputed if T_3 is replaced?

What must be computed if a new node, T_5, is added?

How many SHA-256 hashes must be computed to verify Block 375474?

Mining Cost

The measured time to compute one SHA-256 hash on my EC2 node (2.5 GHz processor) is 750 ns. Approximately *how many instructions execute* to compute on SHA-256 hash?

Assumption. SHA-256 produces a uniform random output. (We know this is not really true, but it is a reasonable approximation, and necessary for the analysis.) So, we can model SHA-256 on any (new) input x as drawing randomly from 2256 possible outputs:

$$\text{SHA-256}(x) \leftarrow [0, 2^{256})$$

Target = T_{max} / Difficulty
$T_{max} \approx 2^{224}$

Current Bitcoin Difficulty = 59,335,351,234

Energy

Why is energy/hash so much less for custom ASICs?

In an ASIC, it is possible to build an XOR using 4 transistors. How many transistors have to flip to do an XOR on a general purpose processor like an Intel i7?

Mining Hardware - current ASIC miners achive >5 Billion hashes per seconds and over 1500 Million hashes per Joule (the energy required to lift an apple one meter).

Inside a Chinese Bitcoin Mine, The Coinsman, 11 August 2014.

> The first thing you notice as you approach the warehouse is the noise. It begins as soon as you step out of the car, at which point it sounds like massive swarm of angry bees droning away somewhere off in the distance. It becomes louder and louder the closer you get to the building, and as you step through the doors it becomes a deafening and steady roar. . .

Our nearest nuclear plant, the North Anna Power Station (Lake Anna) generates 1892 MW, "enough to power 450,000 homes" or about 9x the amount needed to power the current bitcoin network (only counting the miners themselves; additional power needed for cooling, etc.)

How does the energy use of bitcoin compare to what is used by the current financial infrastructure for comparable service? (This is a very difficult question to answer, not something to answer in the blank below!)