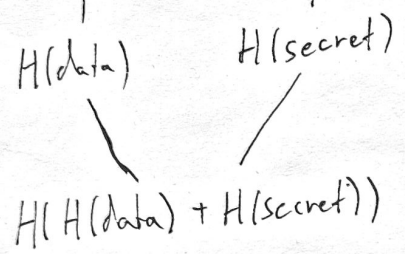
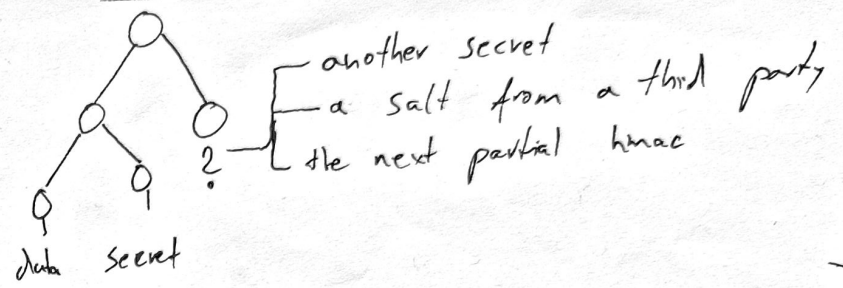


May 2, 2021

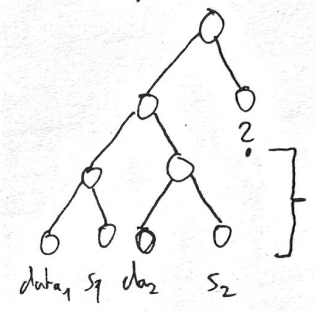
$$\text{hash}([data] + [secret]) \approx \text{hmac}$$



In the proposed hmac authentication, can an attacker replay that spends to ?
 It is unclear if replay protection is possible when root was publicly revealed. If anyone could replay



expanded hmac tree

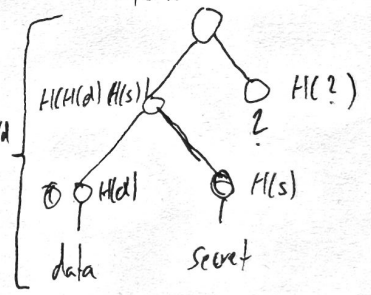


Imagine that each right leaf would be fillable by the owner of the secret with the highest order!

Could Merkle trees consume themselves? (or not?)

$\approx \text{hmac}$ as a hash tree

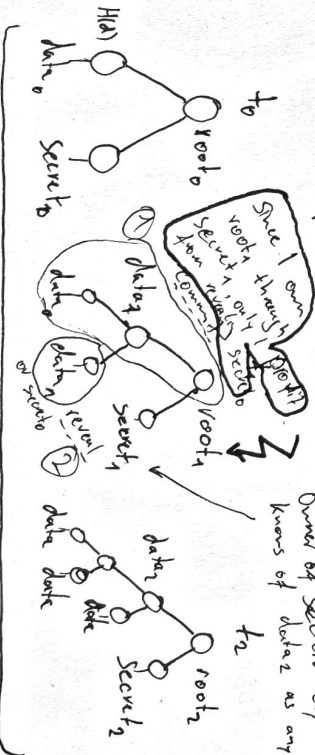
$$H(H(H(H(H(s))H(d))H(s))H(d)) \approx \text{root}$$



- ① We commit the data and the root
- ↳ If we share the secret with another person, they could claim by resolving for the root!
- ⚠ However, this transaction is replayable by copying our tx with a higher gas price! ⚠

WHAT IF?

- ② was a nonce?
- the old merkle tree "ate" the old one?



- Say we want to make a transfer, in what order would we send these bits of data to the contact?

It seems to work!
 Owner of secret only knows of data as anybody else.

- What is public here? A: data, root
 " private " A: secret

How to spend to ?

- ① Commit (tx, root)
- ① reveal (root) which becomes data, commitment is applied.
- ② Send secret to another person along with $H(data)$.

Useful as alternative authentication scheme for signatures.

- generating takes in the EVM is cheap
- It seems only three 32 bytes values are necessary
- Construction is simple to evaluate and implement

① commit } To avoid a
 ② reveal } replay attack!