6.857
NETWORK AND COMPUTER SECURITY
LECTURE 5

LECTURER: RONALD RIVEST
SCRIBE: JOHN WANG

1. Hash Function Applications

1.1. **Password Storage.** System stores $h(pw)$ rather than pw itself. System might also store username, salt, etc.

1.2. **File Modification Detector.** You want to monitor to detect when files have been changed. For each file, store $h(F)$ securely. You can check to see if the files have been modified by recomputing the hash. Provides detection (not prevention).

1.3. **Digital Signatures (hash and sign).**

- PK_A is Alice's public key (for signature verification).
- SK_A is Alice's secret key (for signing).
- Signing: $\sigma = sign(SK_A, m)$ and σ is Alice's signature on message.
- Verification: $verify(M, \sigma, PK_A) \in \{true, false\}$.

Idea: computing $h(m)$ is fast, so sign $h(m)$ instead of signing m. We do $sign(m, SK_A) = sign(m', SK_A)$ if $h(m) = h(m')$.

Problem is that if $h(m) = h(m')$, then asking Alice to sign m, her signature σ is also a signature for m'.

1.4. **Commitments.** Alice has value x which is her bid. She computes $C(x)$ and gives auctioneer $C(x)$, which is her sealed bid. When bidding is over, Alice should be able to open $C(x)$ to reveal x.

Want these properties:

- Binding: Alice should not be able to open $C(x)$ in more than one way.
- Secrecy: Anyone seeing $C(x)$ should have no information about x.
- Non-malleability: Anyone seeing $C(x)$ shouldn't be able to come up with a related bid, e.g. $C(x+1)$.

Let's try $C(x) = h(\text{username}||x)$.