Hash, MAC, and Digital Signature

Dawn Song
dawnsong@cs.berkeley.edu

Review

- **Asymmetric-key encryption**
  - **RSA encryption**
  - **ElGamal encryption**

Hash Function Properties

- **Hash function: a function h with properties**
  - **Compression: h maps an input x of arbitrary length to an output h(x) of a fixed length**
  - **Ease of computation: given h and x, it's easy to compute h(x)**
- **Additional important properties**
  - **Preimage resistance**
  - **2nd-preimage resistance**
  - **Collision resistance**

Three Properties

- **Preimage resistance**
  - For any y (in the range of h) for which a corresponding input is not known, it is computationally infeasible to find any input x such that h(x) = y.
- **2^{nd}-preimage resistance**
  - It is computationally infeasible to find any second input which has the same output as any specified input, i.e., given x, to find $x' \neq x$ s.t. h(x) = h(x')
- **Collision resistance**
  - It is computationally infeasible to find any two distinct inputs x and x' which has to the same output, i.e., h(x) = h(x')

Examples

- **RSA-based one-way function**
  - $f(x) = x^e \bmod N$, where factorization of N is unknown
  - Under RSA assumption, f(x) is preimage resistant
  - What about 2^{nd}-preimage resistance?
- **DES-based one-way fucntion**
  - $f(x) = E(k, x) \oplus x$, for any fixed known key k.
  - Under the assumption that E is a random permutation, f(x) is preimage resistant

Relationships btw Properties (I)

- **Does collision resistance imply 2^{nd}-preimage resistance?**
  - yes
- **Does preimage resistance imply 2^{nd}-preimage resistance?**
  - No
- **Does 2^{nd}-preimage resistance imply preimage resistance?**
  - No

Relationships btw Properties (II)

- **Does collision-resistance imply preimage resistance?**
  - E.g., let g be a hash function which is collision resistant and maps arbitrary-length inputs to n-bit outputs. Consider function h:

    h(x) = 1 || x, if x has bitlength n
    0 || g(x), o.w.
  - Is h collision resistant?
  - Is h preimage resistant?
- **Different applications need different properties**

Cryptographic Hash Functions

- **MD5**
  - Output 128-bit
  - Designed by Ron Rivest, 1991
  - Xiaoyun Wang et. al. found collision in one hour using IBM p690 cluster, 2004
  - Klima find collision with one minute on a notebook computer, using tunneling, 2006
- **SHA-1**
  - Output 160-bit
  - Designed by NSA, adopted by NIST, 1993
  - Xiaoyun Wang et. al. found attack on SHA-1, 2005
    - Requiring fewer than 2^{69} operations to find a collision, whereas brute force would require 2^{80} operations
  - More improvements on attacks
- **NIST is looking for new hash functions**
  - Similar competition as in AES
  - Submissions due Oct 31, 2008

Administrative Matters

- **Group sign-up**
  - During the break
  - Each group has a representative to fill in the sheet
    - Need login info
- **Class accounts**
  - You can pick it up from TAs if you haven't
- **Communication**
  - cs161-spring08 mailing list is only for announcements
  - Do not send your questions there
  - Post your general class-related questions to newsgroup

5-min Break

Message Authentication Code (MAC)

- Encryption: secrecy/confidentiality
- What if Mallory tries to change the message?
- Can encryption alone help?
- Message authentication code (MAC)
  - Provides assurance of source & integrity of msg (data origin authentication)
  - $f(k, M) = f_k(M)$, k is secret key
  - Unforgeability:

    For any fixed value of k unknown to adversary, given a set of values $(x_i, f_k(x_i))$, it is computationally infeasible to compute $f_k(x)$ for any new input x.
- Sample construction: HMAC
  - HMAC(x)= h(k||p||h(k||q||x))
  - Proof of security assuming underlying compression function is PRF

Digital Signatures

- MACs
  - Only parties who have the shared key can verify data integrity & origin
  - Symmetric-key model
- Digital signatures
  - Asymmetric-key model
  - Sender has public/private key
  - Anybody with public key can verify data integrity & origin---non-repudiation

Security Properties of Signature Schemes

1. **Types of attacks**

- **Selective forgery**
  - **Adversary is able to forge a signature for a particular message chosen a priori**
- **Existential forgery**
  - **Adversary is able to forge a signature for at least one message**

2. **Resources of adversary**
   - **Known-message attack: adversary has signatures for messages which are known to adversary but not chosen by him**
   - **Adaptive chosen message attack: adversary can choose which messages to get signatures on using the signer as an oracle**

Conclusion

- **Hash functions**
  - **Properties?**
  - **Relationship between properties?**
- **Message authentication codes**
  - **What security properties is it designed to provide?**
- **Digital signatures?**
  - **What security properties is it designed to provide?**