Data Authenticity

Friday, February 2, 2018 12:16 PM

Public Key Cryptography

- Asymmetric
  - First example = RSA ~2048 bits (to be equivalent to AES-128)
    - RSA Lab standard = PKCS#1
  - Elliptic Curve Cryptography (ECC) ~ 300bits
- Used largely for confidentiality
  - Usually the encryption key is made public
  - Reduces the number of keys to be stored (from n^2 to n)
  - Authenticity: private key applied first
- Public key + CT cannot give Private key or PT
- Private key cannot be easily derived from public key
  - May be partially public

Issues with RSA

- Significantly slower than AES
  - Used to transmit symmetric keys (actual file encrypted with AES)
- Getting private key from public key is as difficult as factorisation
  - Unknown about getting PT from CT + Public Key (assumes this, the RSA Problem, is difficult)
  - Factorisation: currently at 773 bits

Post-quantum cryptography: based on NP completeness

Hash

- Fixed-size digest/ hash value
- Algorithm is public
- Collision-resistant
- One-way (very easy to compute in one direction)
  - P = NP (polynomial time vs. non-deterministic time)
  - E.g. verification vs. solving of factorisation problem
  - Collision resistance implies one-way
- An issue of integrity (has the file been modified? Verify the digest)
- Rainbow Table: time-memory trade-off

Insecure Algorithms

- Taking selected bits (e.g. first 160 bits)
- CRC (cyclic redundancy check)
  - Check value is redundant
  - Proprietary = private, algorithm not revealed
  - Obfuscation = make message difficult to understand with confusing and ambiguous language
  - Reverse engineering

Keyed-hash

- Uses a secret key to output a fixed size mac (message authentication code)
- MAC algorithm -> HMAC
- Anti-forgery: key needed to forge mac
  - Mac sent along with file (also called a tag)

- An issue of authenticity (need the key to produce the digest)

Popular Hashes

- SHA-3 (Keccak), Secure Hash Algorithm
- MD5 (Rivest), Message Digest
- CBC-mac
    - Code re-use: AES algorithm for hashing to save space

Digital Signature

- Signing = private key (needed to forge mac)
- Verification = public key
- Non-repudiation: cannot revoke signature (private key)

Signature Schemes

- RSA-based more popular
- DSA = more secure

Attacks

Birthday attack

- Minimum number of random messages needed to achieve collision with > 0.5 probability
- Length of digest has to be at least 2x length of recommended key length

Electronic Code Book (ECB)

- Different IV for each block -> large overhead in terms of space
- Small IV -> collision chance is high

**Encryption ensures confidentiality
**mac ensures authenticity

- Use security measures

Strong Authentication

- "freshness" of data received

Protecting Password Files

- Hash passwords
- Include userID to minimise collision and guesses on common passwords
    - Salted password
    - Salt (like IV) displayed in clear
- Facebook: passwords already hashed with MD5
    - Cannot apply SHA-3 hash to PT pw
    - Apply SHA-3 to MD5-hashed pw