Asymmetric-key Encryption

Dawn Song
dawnsong@cs.berkeley.edu

Review

- **Introduction to cryptography**
- **Symmetric-key encryption**
- **One-time pad**
- **Block cipher**
  - **DES**
    - **Fiestel Networks**
  - **AES**

Today

- **Modes of operation for Block ciphers**
- **Administrative matters**
- **Modular Arithmetic**
- **Asymmetric-key encryption**

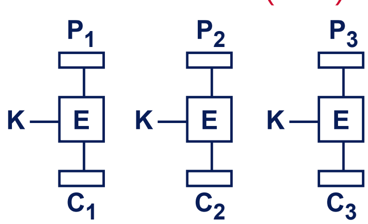
Block-cipher Modes of Operation

- Block-cipher has fixed block size
- How to encrypt arbitrary length msgs using a block cipher?
- How to ensure the same plaintext when encrypted/sent twice, will result in different ciphertexts?
- Different block-cipher modes of operation
  - Encryption scheme
    - Randomized, i.e., flips a coin
    - Stateful, i.e., depending upon state info
  - Decryption scheme
    - Neither randomized nor stateful
    - Why?

Examples of Block-Cipher Modes of Operation

- ECB: Electronic code book
- CBC: Cipher block chaining
- OFB: Output feedback
- CTR: Counter mode

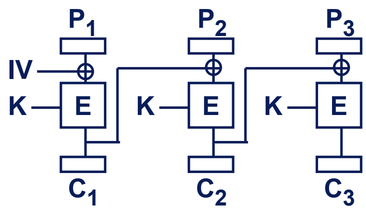
Electronic Code Book (ECB) Mode



- Disadvantages and issues to note
  - Same plaintext always corresponds to same ciphertext
  - Traffic analysis yields which ciphertext blocks are equal → know which plaintext blocks are equal

Cipher Block Chaining (CBC) Mode

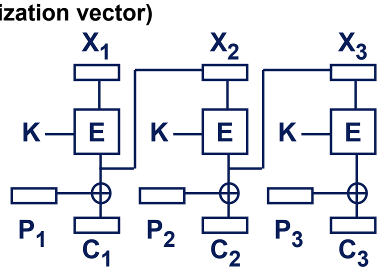
- $C_j = \{ P_j \oplus C_{j-1} \}_K$
- C_0 = IV (initialization vector)



- Interesting fact
  - Altered ciphertext only influences two blocks

Output Feedback (OFB) Mode

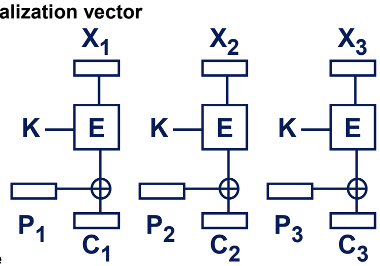
- X_1 = IV (initialization vector)
- $X_j = \{ X_{j-1} \}_K$
- $C_j = X_{j+1} \oplus P_j$



- Altered ciphertext only influences single block

Counter Mode (CTR)

- X_1 = IV called initialization vector
- $X_j = X_1 + j - 1$
- $C_j = \{ X_j \}_K \oplus P_j$



- Advantages
  - Easy to parallelize
- Altered ciphertext only influences single block

Adminstrivia

- **Waitlist**

10

Modular Arithmetic

- **a + b mod s**
- **a*b mod s**
- **a^b mod s**
  - **how to compute a^{25} mod s ?**
  - **Repeated squaring**
    - » **$a^{16} * a^8 * a^1$ mod s**

11

Modular Division

- **How to compute 1/a mod s?**
- **What does it mean?**
  - **$ax \equiv 1$ mod s**
- **Can it always be computed?**
  - **iff gcd (a,s) = 1**
- **How?**
  - **Extended Euclidean algorithm**

12

Euclidean Algorithm

- Compute gcd (a,b)
- Lemma If a > b, then gcd(a,b) = gcd (a mod b, b)
  - Why?
- Euclid algorithm:
  - $b \leq a$,
  - Euclid (a,b) = Euclid (b, a mod b) if $b \neq 0$
    or a if b = 0

Extended Euclidean Algorithm

- For any positive integers a, b, the extended Euclidean algorithm returns integers x, y such that ax + by = gcd (a,b)
- How to use it to compute x such that $ax \equiv 1 \bmod s$?
- gcd (a,s) = 1, thus can compute x, y s.t. ax + sy = 1
  - Thus, $ax \equiv 1 \bmod s$

Asymmetric-key Crypto

- Symmetric cryptography: both parties share the same key
  - Secret key (or shared key) only known to communicating parties
- Asymmetric cryptography: each party has a public and a private key
  - Public key known to everyone
  - Private key only known to owner
- Requirements for secure communication
  - Symmetric crypto: key is secret and authentic
  - Asymmetric crypto: private key is secret and public key is authentic

Advantage of Public-Key Crypto

- Consider N parties, how can any pair of them establish a secret key?
  - To use symmetric-key crypto, requires secret and authentic channel to set up shared secret key
  - Need $O(N^2)$ keys
  - Key management is challenging
- Public-key crypto advantage
  - Each party only needs to know N-1 authentic public keys

Asymmetric-key Encryption

- encryption-Key ≠ decryption-Key
- Alice has public key: pub_key, private key: priv_key
- Bob wants to send Alice message M
- C = E(pub_key, M);
- M = D(priv_key, C)

Asymmetric cryptography

- encryption-Key ≠ decryption-Key
- We cannot simply run operations backwards
- Some things are hard to reverse
  - Multiplication
    - Easy to multiply two large primes
    - Hard to factor
    - Factoring up to 663 bits (200 digits) now demonstrated
      - Intensive computing; record set in May 2005
    - More efficient factoring methods unknown

Using hard problems to make crypto

- Gauss (building on work by Fermat) proved
  - If p and q are primes and
  - If m is not a multiple of p or q
  - Then $m^{(p-1)(q-1)} = 1 \bmod pq$
- Example, p=3, q=5, pq = 15, (p-1)(q-1) = 8
  - $1^8 = 1 = 1 \bmod 15$
  - $2^8 = 256 = 1 \bmod 15$
  - $4^8 = 65536 = 1 \bmod 15$
  - $7^8 = 5764801 = 1 \bmod 15$
  - $8^8 = 16777216 = 1 \bmod 15$
  - $11^8 = 214358881 = 1 \bmod 15$
  - $13^8 = 815730721 = 1 \bmod 15$
  - $14^8 = 1475789056 = 1 \bmod 15$

RSA

- Rivest, Shamir, Adleman (1978 – published 1979)
- Idea:
  - Let p, q be large secret primes, N = pq
  - Given e, find d, such that $ed \equiv 1 \bmod \phi(N)$, where $\phi(N)=(p-1)(q-1)$
  - public key: e, N
  - private key: d, p, q
  - Encryption: $c = E(m) = m^e \bmod pq$
  - Decryption: $D(c) = c^d \bmod pq$
  - So $D(E(m)) = m^{ed} \bmod pq = m^{K(p-1)(q-1)+1} \bmod pq = m$

5-min Break

- Is RSA encryption secure?

Discussion (I)

- **Mallory knows e, so why doesn't she simply compute the e^{th} root to recover the plaintext? E.g., $(M^e \bmod N)^{1/e} = M$?**
- **What if Mallory can find $\phi(N)$?**
  - Then she can compute secret value d
- **Is finding $\phi(N)$ equivalent to factoring?**
  - Yes! Consider the equation $(X-p)(X-q) = 0$
  - Note: $N - \phi(N) + 1 = p+q$
  - $X^2 - (p+q)X + pq = X^2 - (N - \phi(N) + 1)X + N$
  - p and q can be found by solving quadratic equation
- **RSA assumption: finding e-th root mod N is hard when factorization of N is unknown**

Discussion (II)

- **Short plaintext attack:**
  - Consider RSA with n of size 1024 bits, e=3
  - Let's encrypt AES key, secure?
    - » No! 128-bit AES key raised to third power only results in 384-bit #, mod n does not reduce the result, attacker can simply compute cube root over integers
- **What other security issues does RSA have?**
  - E.g., deterministic, same plaintext always encrypt to same ciphertext

How to Fix?

- **Padding:**
  - Pad short plaintext to block size
  - Add randomness
- **Can't just do random padding**
  - E.g., given data D, pad message m to be m= 00 | 02 | r | 00 | D, where r is a random number of appropriate length
  - Bleichenbacher found an attack (1998)
- **Standard: OAEP (Optimal Asymmetric Encryption Padding)**
  - With a formal proof of security