CS4236 Final Examination Cheat-sheet

1. CBC Encryption & MAC

1) *Cipher Block Chaining (CBC) mode:* $c_i = F_k(m_i \oplus c_{i-1})$ and $c_0 = IV$.
2) *IV requirements:* randomly selected, unpredictable & cannot be reused.
3) *Drawbacks:* cannot be parallelized, 1 additional ciphertext block (c_0, IV), F must be invertible (thus we cannot use PRP, pseudorandom permutation).
4) *Error propagation:* If a bit in block c_i is flipped in the transmission of the ciphertext, then p_i is garbled and the corresponding bit in p_{i+1} is flipped.
5) *Stateful CBC:* insecure because IV becomes predictable (SSL 2.0 BEAST attack, because IV is the last block of the previous ciphertext).
6) *CBC-MAC:* 1 $t_i = F_k(m_i \oplus t_{i-1})$ and $t_0 = 0^n$, only output the last block t_l.
7) *Concatenation attack:* possible for arbitrary length CBC-MAC (either use the length of the message as t_0 or encrypt the tag with another key).
8) *CBC-MAC cannot use random IV:* doing so (thus must send IV in clear with the message) is vulnerable because the attacker can change the same i^{th} bit in IV and the first block in message body m_1, without affecting the tag.

2. RSA Encryption

1) *Key pair:* public key (n, e) and private key (n, d).
2) *Derivation:* we have $n = p \cdot q$ and $\varphi(N) = (p-1)(q-1)$, then we could get $gcd(e, \varphi(N)) = 1$, $e \cdot d \equiv 1 \ mod\ \varphi(N)$.
3) *Encryption & decryption:* $c = m^e \ mod\ N$ and $m = c^d \ mod\ N$.
4) *Selection of p and q:* two large-enough primes of equal length. We could use Miller-Rabin test to generate large primes efficiently.
5) Textbook RSA is neither CPA-secure nor CCA-secure since deterministic.

Common-modulus attack 1
- several users share N; users need private encryption
- (e_i,d_i) to user i; $pk_i=(N,e_i)$ and $sk_i=(N,d_i)$
- user i compute $e_i d_i = 1 \bmod \phi(N)$ and solves $(X-p)(X-q)=0$

Common-modulus attack 2
- several users share N; suppose $gcd(e_1,e_2)=1$
- adversary sees $c_1 = m^{e1} \bmod N$, $c_2 = m^{e2} \bmod N$
- since $gcd(e_1,e_2)=1$, there exist X,Y s.t. $Xe_1 + Ye_2 = 1$
- adversary computes $c_1^X c_2^Y = m^{Xe1} m^{Ye2} = m^{Xe1+Ye2} = m \bmod N$

CCA attack 1
- obtains a user's ciphertext $c=[m^e \bmod N]$, picks r <-$- Z_N^* and creates forgery $c'=r^e c \bmod N$
- submits c' for decryption, obtains m' = decryption of c', and discovers $m = m'r^{-1} \bmod N$
  - $m'r^{-1}= (c')^d r^{-1}= (r^e m^e)^d r^{-1} = r^{ed} m^{ed} r^{-1} = rmr^{-1}= m \bmod N$

CCA attack 2
- obtains a user's ciphertext $c=[m^e \bmod N]$ of unknown m
- easy to generate c' that is an encryption of [2m mod N]
  - by setting $c' = [2^e c \bmod N] = 2^e m^e = (2m)^e \bmod N$

3. Diffie-Hellman Key Exchange

1) $\mathbb{Z}_N^*$ = invertible elements in $\{1, 2, \cdots, N-1\}$ under multipcation modulo N.
2) *Theorem:* b is invertible modulo N if and only if they are co-prime.
3) *Cyclic group:* given a finite group G of order m, G is cyclic if and only if there exists a generator g such that $\{g^0, g^1, \cdots\}$ represents all elements in G.
a. Any group of prime order is cyclic, any non-identity element is a generator;
b. Thus, if p is prime, $\mathbb{Z}_p^*$ (of order $p-1$) is cyclic.
4) *Order of element in cyclic group:* for all $x \in \mathbb{Z}_p^*$, the smallest positive integer such that $x^a \equiv 1 \ mod\ p$. In cyclic group $\mathbb{Z}_p^*$, the order of any element is a factor of $p-1$ (the order of the group $\mathbb{Z}_p^*$).
5) *Quadratic residue (QR):* an element $x \in \mathbb{Z}_p^*$ which has a square root in $\mathbb{Z}_p^*$.
a. Each element $x \in \mathbb{Z}_p^*$ has either 0 or 2 square root(s) in $\mathbb{Z}_p^*$;
b. Exactly half of the elements in $\mathbb{Z}_p^*$ are QR;
c. It is computationally feasible to compute square roots in $\mathbb{Z}_p^*$.
6) *Discrete log (DL):* given the generator g and an element x in a cyclic group, find e such that $x \equiv g^e \bmod N$. DL is hard relative to G for all PPT algorithms.
7) Diffie-Hellman (DH) problem: given a cyclic group G with its generator g, define $DH_g(h_1, h_2) = DH_g(g^x, g^y) = g^{xy}$.
a. Computational Diffie-Hellman (CDH): given g, h_1, h_2, find $DH_g(h_1, h_2)$;
b. Decision Diffie-Hellman (DDH): given g, h_1, h_2, and distinguish $DH_g(h_1, h_2)$ from a uniform element in G;
c. If DL is easy, then CDH problem is also easy;
d. DDH is only hard if h_1 and h_2 are QRs inside $\mathbb{Z}_p^*$.
8) *DH key exchange:* set up p & g, exchange g^x & g^y to get g^{xy} as key.
a. DH key exchange achieves <u>forward secrecy</u>;

b. DH key exchange is vulnerable to MITM attack, need authenticated channel.

4. Hash & Digital Signature

1) Collision resistance ⇒ second-preimage resistance ⇒ preimage resistance.
2) *Digital signature:* public verifiability, transferable & non-repudiation.
3) *RSA signature:* textbook version is not secured.

No-message attack
- given pk=<N,e>, choose any $\sigma \in Z_N^*$
- compute $m = [\sigma^e \bmod N]$
- output a forgery (m, σ); σ is a valid signature of m
- m may have no semantic meaning, but can be dangerous in some cases, thus this shouldn't be allowed

Forging a signature on arbitrary message
- adversary chosen message $m \in Z_N^*$
- finds m_1 and m_2, $m=m_1m_2$, that are likely accepted by signer
- obtains σ_1 and σ_2 for m_1 and m_2
- $\sigma^e = (\sigma_1 \sigma_2)^e = (m_1^d m_2^d)^e = m_1^{ed} m_2^{ed} = m_1m_2=m \bmod N$

4) Digital Signature Algorithm (DSA):

p, q: two primes s.t. q|p-1 (details out of scope)

g: a generator in a prime-order subgroup of Z_p^* having order q (hence, g is a generator of a group of size q)

h: $\{0,1\}^* \to Z_q$ (a collision resistant hash function)

private key: *randomly* chosen $x \in Z_q^*$

public key: p, q, g, y where $y = g^x \bmod p$

Sign: given a message $m \in \{0,1\}^*$,
(1) choose $k \in Z_q^*$ uniformly at random
(2) compute $r = (g^k \bmod p) \bmod q$ (in Z_q)
(3) compute $s = (h(m) + x \cdot r) \cdot k^{-1} \bmod q$ (in Z_q)
(4) the signature for m is: (r, s)

Vrfy: given a message $m \in \{0,1\}^*$ and a signature (r, s),
(1) compute $u_1 = h(m) \cdot s^{-1} \bmod q$
(2) compute $u_2 = r \cdot s^{-1} \bmod q$
(3) output **YES** iff $r = ((g^{u_1} y^{u_2} \bmod p) \bmod q)$

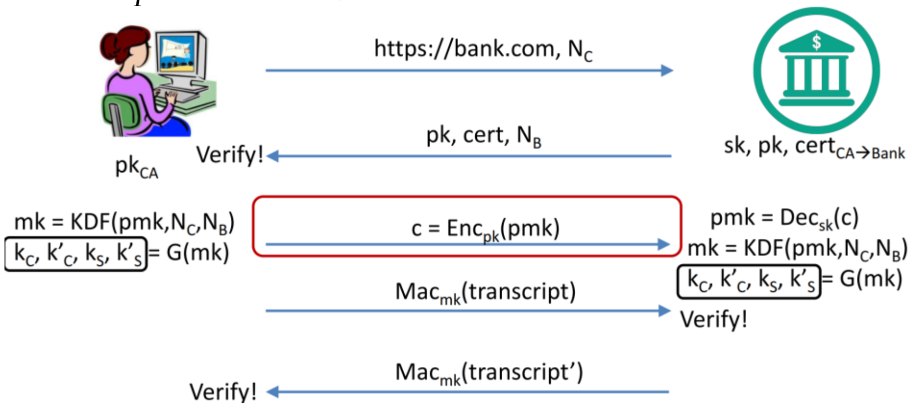
a. k must have high entropy, be unpredictable and unique (cannot reuse);
b. Signing can be fast (by pre-computing k, k^{-1} and r).

5. El Gamal Encryption

1) *Key pair:* private key – x, public key – generator g and $h = g^x$.
2) *Encryption* – $E(m) = \langle mh^r, g^r \rangle$ for random r, *decryption* – $D(\langle a, b \rangle) = ab^{-x}$.
3) *Security implications with different assumptions:*
a. If DL can be solved, one can derive private key from public key;
b. If CDH can be solved, one can get plaintext from ciphertext and public key;
c. If DDH can be solved, El Gamal is not CPA-secure;
d. For El Gamal with $\mathbb{Z}_p^*$, one should pick g that is a QR from $\mathbb{Z}_p^*$.

6. Transport Layer Security (TLS)

1) *Handshake protocol in TLS:*



2) *Record-layer protocol in TLS:* use k_c and k_c' to encrypt/authenticate all messages from the client, use k_s and k_s' to encrypt/authenticate all messages from the server. Use sequence number to prevent replay attack. Use two pairs of keys (i.e., 4 independent keys) to prevent reflection attack.
3) *Key exchange methods in TLS:*
a. RSA-based: pervasive surveillance (no forward secrecy);
b. Fixed DH: no forward secrecy, no authentication for C;
c. Ephemeral DH: forward secrecy (due to fresh pre-master key).

7. Homomorphic Encryption

1) *Homomorphic scheme:* $E(m_1\ op_1\ m_2) = E(m_1)\ op_2\ E(m_2)$.
2) Holomorphicity implies malleability, which means CCA-insecure.
3) For both unpadded RSA and El Gamal, $E(m_1 \cdot m_2) = E(m_1) \cdot E(m_2)$.

8. Secret Sharing

1) *Shamir threshold scheme:* dealer randomly picks a polynomial f of degree $t-1$ (on a finite field) such that $f(0) = k$. Participant P_i gets the value $f(i)$.
2) *Feldman's verifiable scheme:* similarly, given polynomial $f(x) = \sum_{i=t-1}^{0} a_i x^i$, send $f(i)$ to P_i. Dealer broadcasts $g^K \bmod P$, $g^{a_1} \bmod P$, $\cdots$, $g^{a_{t-1}} \bmod P$.

 Good Luck!