Questions from SampleFinal2 not covered in this term's exam: 7, 8, 11

C&O 487/687 Final Examination

April 15, 2013

Total 100 marks. This exam has TWO pages. No calculators, aids, notes, or special materials permitted. Justify all answers and show all work.

1. [8 marks] Define what it means for a MAC scheme to be *secure*.

2. [8 marks] Explain why SHA-256 and AES-Small are said to have the same *security level*. (Recall that SHA-256 is a hash function with 256-bit hash values, and AES-Small is a block cipher with secret keys of bitlength 128.)

3. [8 marks] Alice is given two hash functions $F : \{0,1\}^* \to \{0,1\}^{160}$ and $G : \{0,1\}^* \to \{0,1\}^{160}$. She is told that one of these functions is collision resistant (and the other one is not collision resistant), but she does not know which is which. Alice wishes to use F and G to create a new hash function H which is definitely collision resistant. She defines the hash function $H : \{0,1\}^* \to \{0,1\}^{320}$ as follows: $H(x) = F(x) \| G(x)$ (where $\|$ denotes concatenation). Either prove that H is collision resistant, or provide a counterexample which shows that H is not necessarily collision resistant.

4. [8 marks] Why is encryption exponent $e = 3$ commonly used in deployments of the RSA encryption scheme and the RSA signature scheme? Why not use $e = 2$?

5. [8 marks] Recall that in the Diffie-Hellman key agreement protocol, Alice selects $x \in_R [0, q-1]$ and sends $g^x \bmod p$ to Bob. Similarly, Bob selects $y \in_R [0, q-1]$ and sends $g^y \bmod p$ to Alice. Their shared secret key is $k = H(K)$ where $K = g^{xy} \bmod p$. (Here, p is a prime, q is a prime divisor of $p-1$, g is an element of order q in $\mathbb{Z}_p^*$, and H is a hash function.) Show that the protocol is insecure if the communications channel between Alice and Bob is not authenticated.

6. [8 marks] We recall the DSA signature scheme. The public domain parameters consist of a 1024-bit prime p, a 160-bit prime divisor q of $p-1$, and an element $g \in \mathbb{Z}_p^*$ of order q. SHA-1 is a 160-bit hash function. Alice's private key is $a \in_R [0, q-1]$, while her public key is $h = g^a \bmod p$. To sign a message $M \in \{0,1\}^*$, Alice does the following:

   (i) Select $k \in_R [1, q-1]$.
   (ii) Compute $m = \text{SHA-1}(M)$.
   (iii) Compute $r = (g^k \bmod p) \bmod q$, and check that $r \neq 0$.
   (iv) Compute $s = k^{-1}(m + ar) \bmod q$, and check that $s \neq 0$.
   (v) Alice's signature on M is (r, s).

   To verify A's signature (r, s) on M, Bob does the following:

   (i) Obtain an authentic copy of Alice's public key h.
   (ii) Check that $1 \leq r, s \leq q-1$.
   (iii) Compute $m = \text{SHA-1}(M)$.
   (iv) Compute $u_1 = ms^{-1} \bmod q$ and $u_2 = rs^{-1} \bmod q$.
   (v) Accept iff $r = (g^{u_1} h^{u_2} \bmod p) \bmod q$.

   Show that DSA is existentially forgeable under a key-only attack if SHA-1 is not preimage resistant. (A *key-only* attack means that the attacker knows Alice's public key, but does not have access to a signing oracle.)

7. [8 marks] Let p be an odd prime, and let g be a generator of $\mathbb{Z}_p^*$. Recall that the DLP is: given p, g and $h = g^a \bmod p$ (where $a \in [0, p-2]$), find a. Define the problem COMPUTE-C(x) as follows: Given p, g and $x \in \mathbb{Z}_p^*$, determine $C(x)$ where

   $$C(x) = \begin{cases} 0, & \text{if } \log_g x \text{ is even,} \\ 1, & \text{if } \log_g x \text{ is odd.} \end{cases}$$

   Prove that COMPUTE-C(x) $\leq_P$ DLP. ($A \leq_P B$ means A can be solved if B can be solved.)

   BONUS (2 marks): Does DLP $\leq_P$ COMPUTE-C(x)?

8. [8 marks] Define what it means for a pseudorandom bit generator (PRBG) to be *cryptographically strong*. Define what it means for a PRBG to *pass the next-bit test*.

9. [8 marks] Explain why ECDSA (the elliptic curve analogue of DSA) may be advantageous over DSA.

10. [8 marks] What is *certificate revocation*? Why is it important to have certificate revocation in a public-key infrastructure (PKI)?

11. **Offline electronic cash protocol** (20 marks)
Recall the following *offline* electronic cash protocol. The Bank has an RSA public key (n, e) for signing \$100 dollar bills; this public key is known to Alice (a customer) and Bob (a store).
Withdrawal protocol
i) Alice prepares a message M=(This is a \$100 bill, #serial number).
ii) Alice selects $r \in_R \mathbb{Z}_n^*$.
iii) Alice computes $m' = H(M)r^e \bmod n$.
iv) Alice requests a \$100 withdrawal from the Bank, and gives m' to the Bank.
v) The Bank debits Alice's account by \$100, and gives Alice $s' = (m')^d \bmod n$.
vi) Alice computes $s = s'r^{-1} \bmod n$. The coin is (M, s).
Payment protocol
i) Alice gives the coin (M, s) to Bob.
ii) Bob verifies that s is the valid signature for M.
iii) Bob completes the transaction with Alice.
Deposit protocol
i) Bob forwards the coin (M, s) to the Bank.
ii) The Bank verifies that s is the valid signature for M.
iii) The Bank verifies that the coin has not already been spent.
iv) The Bank enters the coin in a spent-coin database.
v) The Bank credits Bob's account with \$100.

    (a) [4 marks] Prove that s is the Bank's RSA signature on M.

    (b) [8 marks] Show that this offline cash scheme provides *payer anonymity* and *payment untraceability* (i.e., neither Bob nor the Bank can tell whose money was used in a particular payment).

    (c) [8 marks] Show that this offline cash scheme *detects* double spending, but does not *prevent* double spending.