CS255: Cryptography and Computer Security	Winter 2011

Final Exam

Instructions:

– **Answer all five questions.**
– The exam is open book and open notes. Wireless devices are not allowed.
– Students are bound by the Stanford honor code.
– You have two hours.

Problem 1. Questions from all over.

a. Can a symmetric cipher that uses deterministic encryption (with no nonce) be semantically secure under a chosen plaintext attack? If so, explain why. If not, describe a chosen plaintext attack.

b. When building a CBC-MAC from AES one has to properly handle messages whose length is not a multiple of 16 bytes. Describe one method to do so that results in a secure MAC.

c. Suppose Alice bought a certificate from certificate authority X. Alice intends to use the certificate to issue signatures in her name (e.g. to sign code that Alice develops). If X is malicious, can it forge Alice's signature on rogue malware? More precisely, can X fool a verifier into believing that a certain rogue malware was written by Alice? If so explain how, if not explain why not. You may assume the verifier has not seen signatures from Alice before.

d. Describe a concrete attack that is prevented by challenge-response authentication, but is not prevented by authentication based on one-time passwords. Please be specific when describing how an attacker defeats the one-time password scheme.

Problem 2. Let F be a secure PRF defined over $(\mathcal{K}, \mathcal{X}, \{0,1\}^n)$. Which of the following is a secure PRF? Justify your answer.

a. $F_1(\ (k_1, k_2),\ x) := F(k_1, x) \oplus F(k_2, x)$.

b. $F_2(\ (k_1, k_2),\ x) := F(k_1, x) \wedge F(k_2, x)$. Here $u \wedge v$ is the bit-wise and of u and v.

c. $F_3(k, x) := F(k, x)|_{1\ldots4}$. (for $y \in \{0,1\}^n$, $y|_{1\ldots4}$ are the 4 least significant bits of y)

d. $n = 128$ and $F_4(\ (k_1, k_2),\ x) := \begin{cases} F(k_1, x) & \text{if } F(k_2, x) = 0, \text{ and} \\ F(k_2, x) & \text{otherwise} \end{cases}$

e. Same as part (d), but with $n = 1$.

f. Treat $\{0,1\}^n$ as the integers $\{0, \ldots, 2^n - 1\}$ with multiplication modulo 2^n. Let $n = 128$ and define $F_5(\ (k_1, k_2),\ x) := F(k_1, x) \cdot F(k_2, x)$.

Problem 3. In this question we look at concrete security of CBC and counter modes.

a. Let F be a secure PRF defined over $(\mathcal{K}, \{0,1\}^{32}, \mathcal{Y})$, namely F has domain is $\{0,1\}^{32}$. Suppose we construct a symmetric cipher from this F using randomized counter mode. We plan to use this cipher to encrypt two movies with the same key, where each movie contains 2^{32} blocks of F. Will the cipher provide semantic security under a chosen plaintext attack in these settings (i.e. where the attacker sees the encryption of two messages of his choice, each 2^{32} blocks long)? If so, explain why. If not, describe a chosen plaintext attack that breaks semantic security.

Note: if you describe a chosen plaintext attack, the attacker should query for the encryption of one message of his choice and then use that to solve a semantic security challenge. In total the attacker is given two ciphertexts.

b. Let π be a secure PRP defined over $(\mathcal{K}, \{0,1\}^{64})$. Suppose we construct a symmetric cipher from this π using randomized CBC mode (CBC mode with a random IV). As before, we plan to use this cipher to encrypt two movies with the same key, where each movie contains 2^{32} blocks of π. Will the cipher provide semantic security under a chosen plaintext attack in these settings (i.e. where the attacker sees the encryption of two messages of his choice, each 2^{32} blocks long)? If so, explain why. If not, describe a chosen plaintext attack that breaks semantic security using the note from part (a).

Hint: consider the effect of the birthday paradox.

Problem 4. One-time signatures from discrete-log. Let $\mathbb{G}$ be a cyclic group of prime order q with generator g. Consider the following signature system for signing messages m in $\mathbb{Z}_q$:

KeyGen: choose $x, y \xleftarrow{R} \mathbb{Z}_q$, set $h := g^x$ and $u := g^y$.
output $\mathrm{sk} := (x, y)$ and $\mathrm{pk} := (g, h, u) \in \mathbb{G}^3$.

$\mathsf{Sign}(\mathrm{sk}, m)$: output s such that $u = g^m h^s$.

$\mathsf{Verify}(\mathrm{pk}, m, s)$: output '1' if $u = g^m h^s$ and '0' otherwise.

a. Explain how the signing algorithm works. That is, show how to find s using sk.

b. Show that the signature scheme is weakly one-time secure assuming the discrete-log problem in $\mathbb{G}$ is hard. That is, suppose there is an adversary $\mathcal{A}$ that asks for a signature on a message $m \in \mathbb{Z}_q$ and in response is given the public key pk and a signature s on m. The adversary then outputs a signature forgery (m^*, s^*) where $m \neq m^*$. Show how to use $\mathcal{A}$ to compute discrete-log in $\mathbb{G}$. This will prove that the signature is secure as long as the adversary sees at most one signature.

Hint: Your goal is to construct an algorithm $\mathcal{B}$ that given a random $h \in \mathbb{G}$ outputs an $x \in \mathbb{Z}_q$ such that $h = g^x$. Your algorithm $\mathcal{B}$ runs adversary $\mathcal{A}$ and receives a message m from $\mathcal{A}$. Show how $\mathcal{B}$ can generate a public key $\mathrm{pk} = (g, h, u)$ so that it has a signature s for m. Your algorithm $\mathcal{B}$ then sends pk and s to $\mathcal{A}$ and receives from $\mathcal{A}$ a signature forgery (m^*, s^*). Show how to use the signatures on m^* and m to compute the discrete-log of h base g.

c. Show that this signature scheme is not 2-time secure. Given the signature on two distinct messages $m_0, m_1 \in \mathbb{Z}_q$ show how to forge a signature for any other message $m \in \mathbb{Z}_q$.

d. Explain how you would extend this signature scheme to sign arbitrary long messages rather than just messages in $\mathbb{Z}_q$.

Problem 5. In class we showed a collision resistant hash function from the discrete-log problem. Here let's do the same, but from the RSA problem. Let n be a random RSA modulus, e a prime relatively prime to $\varphi(n)$, and u random in $\mathbb{Z}_n^*$. Show that the function

$$H_{n,u,e} : \mathbb{Z}_n^* \times \{0, \ldots, e-1\} \to \mathbb{Z}_n^* \qquad \text{defined by} \qquad H_{n,u,e}(x, y) := x^e u^y \quad \in \mathbb{Z}_n$$

is collision resistant assuming that taking e'th roots modulo n is hard.

Suppose $\mathcal{A}$ is an algorithm that takes n, u as input and outputs a collision for $H_{n,u,e}(\cdot, \cdot)$. Your goal is to construct an algorithm $\mathcal{B}$ for computing e'th roots modulo n.

a. Your algorithm $\mathcal{B}$ takes random n, u as input and should output $u^{1/e}$. First, show how to use $\mathcal{A}$ to construct $a \in \mathbb{Z}_n$ and $b \in \mathbb{Z}$ such that $a^e = u^b$ and $0 \neq |b| < e$.

b. Clearly $a^{1/b}$ is an e'th root of u (since $(a^{1/b})^e = u$), but unfortunately for $\mathcal{B}$, it cannot compute roots in $\mathbb{Z}_n$. Nevertheless, show how $\mathcal{B}$ can compute $a^{1/b}$. This will complete your description of algorithm $\mathcal{B}$ and prove that a collision finder can be used to compute e'th roots in $\mathbb{Z}_n^*$.
Hint: since e is prime and $0 \neq |b| < e$ we know that b and e are relatively prime. Hence, there are integers s, t so that $bs + et = 1$. Use a, u, s, t to find the e'th root of u.

c. Show that if we extend the domain of the function to $\mathbb{Z}_n^* \times \{0, \ldots, e\}$ then the function is no longer collision resistant.