**CS255: Cryptography and Computer Security** **Winter 2002**

# Assignment #3

Due: Friday, March 8th, 2002 at 5pm.

**Problem 1** Parties $A_1, \ldots, A_n$ and $B$ wish to generate a secret conference key. All parties should know the conference key, but an eavesdropper should not be able to obtain any information about the key. They decide to use the following variant of Diffie-Hellman: there is a public prime $p$ and a public element $g \in \mathbb{Z}_p^*$ of order $q$ for some large prime $q$ dividing $p-1$. User $B$ picks a secret random $b \in [1, q-1]$ and computes $y = g^b \bmod p$. Each party $A_i$ picks a secret random $a_i \in [1, q-1]$ and computes $x_i = g^{a_i} \bmod p$. User $A_i$ sends $x_i$ to $B$. User $B$ responds to party $i$ by sending $z_i = x_i^b \bmod p$.

**a.** Show that party $i$ given $z_i$ (and $a_i$) can determine $y$.

**b.** Explain why (a hash of) $y$ can be securely used as the conference key. Namely, explain why at the end of the protocol all parties $A_1, \ldots, A_n$ and $B$ know $y$ and give a brief informal explanation why an eavesdropper cannot determine $y$.

**c.** Formally prove part (b). Namely, show that if there exists an efficient algorithm $\mathcal{A}$ that given the public values in the above protocol, outputs $y$, then there also exists an efficient algorithm $\mathcal{B}$ that breaks the Computational Diffie-Hellman assumption (using $p$ and $g$ as the public values). Use algorithm $\mathcal{A}$ as a subroutine in your algorithm $\mathcal{B}$. Note that algorithm $\mathcal{B}$ takes $g^a \bmod p$ and $g^b \bmod p$ as input and should output $g^{ab} \bmod p$.

**Problem 2** Let $N = pq$ be an RSA composite. Let $g \in [0, N^2]$ be an integer satisfying $g = aN + 1 \bmod N$ for some $a \in \mathbb{Z}_N^*$. Consider the following encryption scheme. The public key is $\langle N, g \rangle$. To encrypt a message $m \in \mathbb{Z}_N$ do: (1) pick a random $h \in \mathbb{Z}_{N^2}^*$, and (2) compute $C = g^m \cdot h^N \bmod N^2$. Our goal is to develop a decryption algorithm.

**a.** Show that the discrete log problem base $g$ is easy. That is, show that given $g$ and $B = g^x \bmod N^2$ there is an efficient algorithm to recover $x \bmod N$. Use the fact that $g = aN + 1$ for some integer $a \in \mathbb{Z}_N^*$.

**b.** Show that given $g$ and the factorization of $N$, decrypting $C = g^m \cdot h^N \bmod N^2$ can be done efficiently.
Hint: consider $C^{\varphi(N)} \bmod N^2$. Use the fact that by Euler's theorem $x^{\varphi(N^2)} = 1 \bmod N^2$ for any $x \in \mathbb{Z}_{N^2}^*$.

**c.** Show that this encryption scheme enables limited computation on ciphertexts. Let $a, b, c$ be integers in $[1, N]$. Show that given $N$ and $c$, and the encryption of $a$ and $b$ it is possible to construct the encryption of $a + b$ and the encryption of $c \cdot a$. More precisely, show that given $N$ and an integer $c$, and ciphertexts $C_1 = E[a]$, $C_2 = E[b]$, it is possible to construct the ciphertexts $C_3 = E[a + b]$ and $C_4 = E[c \cdot a]$.

**Problem 3** Rabin suggested a signature scheme very similar to RSA signatures. In its simplest form, the public key is a product of two large primes $N = pq$ and the private key is $p$ and $q$. The signature $S$ of a message $M \in \mathbb{Z}_N$ is the square root of $M$ modulo $N$. For simplicity, assume that the messages $M$ being signed are always quadratic residues modulo $N$. To verify the signature, simply check that $S^2 = M \bmod N$. Note that we did not include any hashing of $M$ prior to signing. Show that a chosen message attack on the scheme can result in a total break. More precisely, if an attacker can get Alice to sign messages chosen by the attacker then the attacker can factor $N$.

**Hint:** recall that a quadratic residue modulo $N = pq$ has four square roots in $\mathbb{Z}_N$. First show that there are two square roots of $M$ that enable the attacker to factor $N$ (use the fact that gcd's are easy to compute). Then show how using a chosen message attack the attacker can get a hold of such a pair of square roots with high probability. Note that proper hashing prior to signing prevents this attack.

**Problem 4** Let's explore why in the RSA public key system each person has to be assigned a different modulus $N = pq$. Suppose we try to use the same modulus $N = pq$ for everyone. Each person is assigned a public exponent $e_i$ and a private exponent $d_i$ such that $e_i \cdot d_i = 1 \bmod \varphi(N)$. At first this appears to work fine: to encrypt a message to Bob, Alice computes $C = M^{e_{bob}}$ and sends $C$ to Bob. An eavesdropper Eve, not knowing $d_{bob}$ appears to be unable to decrypt $C$. Let's show that using $e_{eve}$ and $d_{eve}$ Eve can very easily decrypt $C$.

**a.** Show that given $e_{eve}$ and $d_{eve}$ Eve can obtain a multiple of $\varphi(N)$.

**b.** Show that given an integer $K$ which is a multiple of $\varphi(N)$ Eve can factor the modulus $N$.
Hint: Consider the sequence $g^K, g^{K/2}, g^{K/4}, \ldots g^{K/\tau(N)} \bmod N$ where $g$ is random in $\mathbb{Z}_N$ and $\tau(N)$ is the largest power of 2 dividing $K$. Use the the left most element in this sequence which is not equal to $1 \bmod N$.

**c.** Deduce that Eve can decrypt any RSA ciphertext encrypted using the modulus $N$ intended for Alice or Bob (at this point this should be obvious).

**Problem 5** Recall that a simple RSA signature $S = H(M)^d \bmod N$ is computed by first computing $S_1 = H(M)^d \bmod p$ and $S_2 = H(M)^d \bmod q$. The signature $S$ is then found by combining $S_1$ and $S_2$ using the Chinese Remainder Theorem (CRT). Now, suppose a CA is about to sign a certain certificate $C$. While the CA is computing $S_1 = H(C)^d \bmod p$, a glitch on the CA's machine causes it to produce the wrong value $\tilde{S}_1$ which is not equal to $S_1$. The CA computes $S_2 = H(C)^d \bmod q$ correctly. Clearly the resulting signature $\tilde{S}$ is invalid. The CA then proceeds to publish the newly generated certificate with the invalid signature $\tilde{S}$.

**a.** Show that any person who obtains the certificate $C$ along with the invalid signature $\tilde{S}$ is able to factor the CA's modulus.
Hint: Use the fact that $\tilde{S}^e = H(C) \bmod q$. Here $e$ is the public verification exponent.

**b.** Suggest some method by which the CA can defend itself against this danger.

**Extra credit:** In the lecture we defined the Decision Diffie-Hellman problem (DDH) as follows: let $G$ be a group of prime order $q$. An algorithm $\mathcal{A}$ $\epsilon$-solves the DDH problem in $G$ if:

$$\left|\Pr[\mathcal{A}(g, g^a, g^b, g^{ab}) = \text{“yes''}] - \Pr[\mathcal{A}(g, g^a, g^b, g^c) = \text{“yes''}]\right| \geq \epsilon$$

where $g \neq 1$ is uniform in $G$ and $a, b, c$ are uniform in $\mathbb{Z}_q^*$. In other words, $\mathcal{A}$ is able to distinguish between a distribution of Diffie-Hellman tuples and a distribution of random tuples.

We also said that an algorithm $\mathcal{B}$ $\epsilon$-breaks the semantic security of a public-key encryption scheme $\mathcal{E}$ if $\mathcal{B}$ wins the following game with probability at least $\frac{1}{2} + \epsilon$:
(1) $\mathcal{B}$ is given a public-key generated by the key generation algorithm of $\mathcal{E}$,
(2) $\mathcal{B}$ outputs two messages $M_0, M_1$,
(3) $\mathcal{B}$ is given the public key encryption of $M_b$ under the public key from step (1) where $b$ is random in $\{0, 1\}$,
(4) $\mathcal{B}$ returns a $b' \in \{0, 1\}$ and wins the game if $b = b'$.

Consider the original ElGamal encryption scheme where the encryption of a message $M \in G$ is $C = [g^r,\ M \cdot y^r]$ where $\langle g, y \rangle \in G$ is the public key and $r$ is random in $\mathbb{Z}_q$. Show that this ElGamal encryption scheme is semantically secure assuming DDH in $G$ is hard. In other words, show that if an algorithm $\mathcal{B}$ $\epsilon$-breaks semantic security of ElGamal in $G$ then there is an algorithm $\mathcal{A}$ with approximately the same running time as $\mathcal{B}$ that $\epsilon$-breaks DDH in $G$. (your goal is to design algorithm $\mathcal{A}$ for DDH in $G$ that uses $\mathcal{B}$ as a subroutine).