CS255: Cryptography and Computer Security **Winter 2000**

Assignment #2

Due: Wednesday, February 9th, 2000.

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 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 $\mathcal{B}$ to break the Diffie-Hellman protocol (using p and g as the public values). Use algorithm $\mathcal{A}$ as a subroutine in your algorithm $\mathcal{B}$ for breaking the Diffie-Hellman protocol. 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 To achieve fast encryption it is desirable to make the public exponent e in the RSA cryptosystem as small as possible. Consider the case when $e = 3$. Show that given the public key an attacker can easily recover the half-most-significant bits of the private exponent d. In other words, show that when N is n bits long, an attacker can recover the $n/2$ most significant bits of d just given the modulus N. Is this a sufficiently serious threat that $e = 3$ should not be used? (just state your opinion)

A. Since $ed = 1 \bmod \varphi(N)$ there exists an integer k such that $ed - k\varphi(N) = 1$. First show that when $e = 3$ we must have $k = 2$. Recall that d is in the range $0 < d < \varphi(N)$.

B. Next show that given k and N it is easy to construct a number $\hat{d}$ that is very close to d — sufficiently close so as to match d on the $n/2 - 4$ most significant bits (with very high probability).
Hint: Observe that since p and q are on the order of $\sqrt{N}$ we have that $\varphi(N)$ is very close to N.

Problem 3 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 multiple of $\varphi(N)$ Eve can easily recover Bob's private key d_{bob} from his public key e_{bob}.

C. Deduce that Eve can decrypt any message encrypted using the modulus N (at this point this should be obvious).

Problem 4 Suppose one uses a modulus p and a *generator* g of $\mathbb{Z}_p^*$ as parameters for the ElGamal public key system. Show that the system leaks one bit of information about the plaintext. Namely, given the public key and a ciphertext, an eavesdropper can learn the Legendre symbol of the plaintext.