CS 435: Introduction to Cryptography **Spring 2020**

Homework 7

Professor Somesh Jha **Due:** May 6

1. **Exercise 10.3**

   Describe a man-in-the-middle attack on the Diffie-Hellman protocol where the adversary shares a key k_A with Alice and a (different) key k_B with Bob, and Alice and Bob cannot detect that anything is wrong.

   Solution:

   Consider the following scheme where Oscar is the adversary:

   (1) Alice picks $x \leftarrow \mathbb{Z}_q$ and sends g^x to Bob

   (2) Oscar gets g^x from Alice, picks $x' \leftarrow \mathbb{Z}_q$ and sends $g^{x'}$ to Bob

   (3) Bob picks $y \leftarrow \mathbb{Z}_q$ and sends g^y to Alice

   (4) Oscar gets g^y from Bob, picks $y' \leftarrow \mathbb{Z}_q$ and sends $g^{y'}$ to Alice

   Bob thinks the key is $k_B = (g^{x'})^y$ and Alice thinks the key is $k_A = (g^{y'})^x$.

2. Consider the following public-key encryption scheme. The public key is $(\mathbb{G}, q, g, h)$ and the private key is x, generated exactly as in the El Gamal encryption scheme. In order to encrypt a bit b, the sender does the following:

   (a) If $b = 0$ then chose uniformly $y \in \mathbb{Z}_q$ and compute $c_1 := g^y$ and $c_2 := h^y$. The cipher text is $\langle c_1, c_2 \rangle$.

   (b) If $b = 1$ then choose independent uniform $y, z \in \mathbb{Z}_q$, compute $c_1 := g^y$ and $c_2 := g^z$ and set the ciphertext equal to $\langle c_1, c_2 \rangle$.

   Show that it is possible to decrypt efficiently given knowledge of x.

   Solution:

   If $\frac{(c_1)^x}{c_2} = 1$ output $b = 0$ else output $b = 1$.

   When encrypting bit 0, $c_2 := h^y = (g^x)^y$ and $c_1 = g^y$. Thus $\frac{(c_1)^x}{c_2} = 1$.

   When encrypting bit 1, $c_2 := g^z$ and $c_1 = g^y$. Thus $\frac{(c_1)^x}{c_2} \neq 1$. Unless $z = xy$ which has negligible probability.

3. How can CRT be used to speed up RSA decryption?

   Solution:

   We wish to solve the equation $x \equiv c^d \pmod{N}$. Assuming exponentiation modulo an l-bit integer takes $\gamma \cdot l^3$ operations for some constant γ. If p, q are each n bits long, then naively computing $c^d \mod N$ takes $\gamma \cdot (2n)^3 = 8\gamma \cdot n^3$ steps (because $\|N\| = 2n$).

Using the uniqueness property of CRT we can solve the following equivalent system instead:

$$\begin{cases} x \equiv c^d \pmod{p} \\ x \equiv c^d \pmod{q} \end{cases}$$

Let us write d as $(p-1)l + m$ where m is the remainder $(d \mod (p-1))$.

Now we can write $c^d \equiv (c^{p-1})^l \cdot c^m \pmod{p}$.

Since p is prime, we can use Fermat's little theorem to conclude $c^d \equiv c^m \pmod{p}$ or in other words $c^d \equiv c^{d \mod (p-1)} \pmod{p}$. Using the same argument for q we get the following equivalent system of equations:

$$\begin{cases} x \equiv c^{d \mod (p-1)} \pmod{p} \\ x \equiv c^{d \mod (q-1)} \pmod{q} \end{cases}$$

Notice that $d \mod (p-1)$ and $d \mod (q-1)$ are independent of c and need to be computed only once. We then compute $c^{d \mod (p-1)} \mod p$ and $c^{d \mod (q-1)} \mod q$ both of which take $\gamma \cdot n^3$ steps (because $\|p\| = \|q\| = n$), let us call them a_p and a_q respectively. Now we are left with the following equivalent system of equations:

$$\begin{cases} x \equiv a_p \pmod{p} \\ x \equiv a_q \pmod{q} \end{cases}$$

Using CRT we know that the solution for this is $a_p \cdot q \cdot q^{p-2} + a_q \cdot p \cdot p^{q-2}$ (use Fermat's little theorem). Thus, we have successfully reduced the time complexity from $8\gamma \cdot n^3$ to $2\gamma \cdot n^3$.

4. **Exercise 10.4**

Consider the following key-exchange protocol:

(a) Alice chooses uniform $k, r \in \{0,1\}^n$, and sends $s := k \oplus r$ to Bob.
(b) Bob chooses uniform $t \in \{0,1\}^n$, and sends $u := s \oplus t$ to Alice.
(c) Alice computes $w := u \oplus r$ and sends w to Bob.
(d) Alice outputs k and Bob outputs $w \oplus t$.

Show that Alice and Bob output the same key. Analyze the security of the scheme (i.e., either prove its security or show a concrete attack).

Solution:

The adversary has knowledge of the communication between Alice and Bob. In particular the adversary has knowledge of

(a) $s := k \oplus r$,
(b) $u := s \oplus t = k \oplus r \oplus t$, and
(c) $w := u \oplus r = k \oplus t$.

Alice and Bob both output the same key k ($k = w \oplus t$). Observe that $k = s \oplus u \oplus w$, which tells us that the adversary can also compute the key k. Since an eavesdropper can compute the key from the transcript of communication between the two parties, the scheme is not secure.