1172/DCP1323 Introduction to Cryptography, Spring 2019 **Due:2019/5/20 (Mon.)**

Homework 5

Instructor: Prof. Wen-Guey Tseng Scribe: Lyue Li

1.

a. Consider the Davies and Price hash code scheme described in Section 11.4 and assume that DES is used as the encryption algorithm:

$$H_i = H_{i-1} \oplus E(M_i, H_{i-1})$$

Recall the complementarity property of DES (Problem 3.14): If $Y = E(K, X)$, then $Y' = E(K', X')$. Use this property to show how a message consisting of blocks $M_1, M_2, \ldots, M_N$ can be altered without altering its hash code.

b. Show that a similar attack will succeed against the scheme proposed in [MEYE88]:

$$H_i = M_i \oplus E(H_{i-1}, M_i)$$

Answer.

a. For clarity, we use overbars for complementation. We have:

$$E(\overline{M_i}, \overline{H_{i-1}}) = \overline{E(M_i, H_{i-1})} \oplus \overline{H_i} = E(M_i, H_{i-1}) \oplus H_i$$

Therefore, the hash function of message M with initial value I is the same as the hash function for message N with initial value I for any given I, where

$M = M_1 \| M_2 \| \ldots \| M_n$; $N = M_1 \| M_2 \| \ldots \| M_n$

b. The same line of reasoning applies with the Ms and Hs reversed in the derivation.

2. Now consider the opposite problem: using an encryption algorithm to construct a one-way hash function. Consider using RSA with a known key. Then process a message consisting of a sequence of blocks as follows: Encrypt the first block, XOR the result with the second block and encrypt again, etc. Show that this scheme is not secure by solving the following problem. Given a two-block message B1, B2, and its hash

$$RSAH(B1, B2) = RSA(RSA(B1) \oplus B2)$$

Given an arbitrary block C1, choose C2 so that $RSAH(C1, C2) = RSAH(B1, B2)$.
Thus, the hash function does not satisfy weak collision resistance.

Answer.

$$\begin{aligned} RSAH(C1, C2) &= RSA(\, RSA(C1) \oplus C2\,) \\ &= RSA(\, RSA(C1) \oplus RSA(C1) \oplus RSA(B1) \oplus B2\,) \\ &= RSA(\, RSA(B1) \oplus B2\,) \\ &= RSA(\, B1, B2\,) \end{aligned}$$

Therefore, choose $C2 = RSA(C1) \oplus RSA(B1) \oplus B2$

3. DSA specifies that if the signature generation process results in a value of s = 0, a new value of k should be generated and the signature should be recalculated. Why?

Answer.

A user who produces a signature with $s = 0$ is inadvertently revealing his or her private key d via the relationship:

$$s = 0 = k^{-1}[H(m) + dr] \bmod q$$

$$d = \frac{-H(m)}{r} \quad mod\ q$$

4. It is tempting to try to develop a variation on Diffie–Hellman that could be used as a digital signature. Here is one that is simpler than DSA and that does not require a secret random number in addition to the private key.

Public elements:	q	prime number
	α	$\alpha < q$ and α is primitive root of q
Private key:	X	$X < q$
Public key:	Y	$= \alpha^X \bmod q \bmod q$

To sign a message M, compute $h = H(M)$, which is the hash code of the message. We require that $\gcd(h, q-1) = 1$. If not, append the hash to the message and calculate a new hash. Continue this process until a hash code is produced that is relatively prime to $(q-1)$. Then calculate Z to satisfy $Z \equiv X \times h (\bmod q-1)$. The signature of the message is $\sigma = \alpha^Z$. To verify the signature, a user compute t such that $t \times h = 1 \bmod (q-1)$ and verifies $Y = \sigma^t \bmod q$. Show that the scheme is unacceptable by describing a simple technique for forging a user's signature on an arbitrary message.

Answer.

$$\sigma^t\ mod\ q = (\alpha^Z)^t\ mod\ q = \alpha^{x \times h \times t\ mod\ (q-1)}\ mod\ q = \alpha^x\ mod\ q = Y$$

5. Assume a technique for a digital signature scheme using a cryptographic one-way hash function (H) as follows. To sign an n-bit message, the sender randomly generates in advance 2n 64-bit cryptographic keys:

$$k_1, k_2, \dots, k_n, k_1', k_2', \dots, k_n'$$

which are kept private. The sender generates the following two sets of validation parameters which are made public.

$$v_1, v_2, \dots, v_n\ and\ v_1', v_2', \dots, v_n'$$

where

$$v_i = H(k_i||0), \qquad v_i' = H(k_i'||1)$$

The user sends the appropriate k_i or K_i' according to whether M_i is 0 or 1 respectively.
For example, if the first 3-bits of the message are 011, then the first three keys of the signature are k_1, k_2', k_3'

a. How does the receiver validate the message?

Answer.

The receiver validates the signature as follows. If ith bit of M is 0, it computes $H(k_i||0)$ otherwise $H(k_i'||1)$ as the sender knows both the private keys (k_i and k_i'), who originally avails v_i and v_i' . The receiver can verify correctly with vi or vi' based on the value of M_i (0 or 1).

b. Is the technique secure?

Answer.

An opponent wants to discover the k_i from v_i and v_i' is computationally infeasible as H is a one-way hash function. So, the scheme is secure.

c. How many times can the same set of secret keys be safely used for different messages?

Answer.

It can be used once only. As the keys are disclosed after once execution.

d. What, if any, practical problems does this scheme present?

Answer.

The scheme seems impractical as for an n-bit message n keys, thus n×64-bit length signature is required.