Security II: Cryptography – exercises

Markus Kuhn

Lent 2014 – Part II

Exercise 1: Show that an encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ over a message space $\mathcal{M}$ is *perfectly secret* if and only if

(a) for every probability distribution over $\mathcal{M}$, every message $M \in \mathcal{M}$, and every ciphertext $C \in \mathcal{C}$ with $P(C) > 0$ we have

$$P(C|M) = P(C).$$

(b) for every probability distribution over $\mathcal{M}$, every message pair $M_0, M_1 \in \mathcal{M}$, and every ciphertext $C \in \mathcal{C}$ with $P(C) > 0$ we have

$$P(C|M_0) = P(C|M_1).$$

Exercise 2: In the CBC mode of operation, the initial vector (IV) is chosen uniformly at random, using a secure source of random bits. Show that CBC would not be CPA secure if the initial vector could be anticipated by the adversary, for example because it is generated instead using a counter or a time-stamp.

Exercise 3:

Show that CTR mode is not CCA secure.

Exercise 4: Your colleagues have invented a new authenticated encryption scheme that they call AES-CBC+CMAC. Their key generating function outputs a 128-bit AES key K, and their encryption function outputs $C \| T = \mathsf{Enc}_K(M) \| \mathsf{Mac}_K(M)$, where $\mathsf{Enc}_K(M)$ shall be the AES-CBC encryption of M with key K (with random IV each time), and $\mathsf{Mac}_K(M)$ shall be the AES-CMAC of M with key K. Show that this construct lacks CPA security.

Exercise 5: Use Euclid's algorithm to calculate $\gcd(36, 24)$.

Exercise 6: Use Euler's theorem to calculate the inverse

(a) $5^{-1} \bmod 7$

(b) $5^{-1} \bmod 12$

(c) $5^{-1} \bmod 15$

Exercise 7: Given an abelian group $(\mathbb{G}, \bullet)$, let $\mathbb{H}$ be the set of its quadratic residues, that is $\mathbb{H} = \{g^2 \mid g \in \mathbb{G}\}$. Show that $(\mathbb{H}, \bullet)$ is a subgroup of $(\mathbb{G}, \bullet)$.

Exercise 8: With RSA encryption, it is common practice to choose e as a small number (e.g., 3, 17, $2^{16} + 1$).

(*a*) How does this affect the speed of encryption?

(*b*) If you wanted to make decryption faster, could you simply set d to one of these three values instead?

(*c*) How else can RSA decryption be calculated more efficiently using the Chinese Remainder Theorem and Fermat's little theorem?

Exercise 9: In the textbook RSA encryption scheme, with $n = pq$ being a product of two different primes and $ed \bmod \varphi(n) = 1$, the identity $m^{ed} \bmod n = m$, which states that we obtain the same plaintext m after encryption and decryption, is only guaranteed by Euler's theorem for any $m \in \mathbb{Z}_n^*$, that is if $\gcd(n, m) = 1$.

(*a*) Show that it actually also holds for any $m \in \mathbb{Z}_n$. [Hint: CRT]

(*b*) Conversely, if we instead had chosen $n = p^2$ being the square of a prime number (i.e., $p = q$), show a simple example for the fact that in this case $ed \bmod \varphi(n) = 1$ does *not* imply $m^{ed} \bmod n = m$ for all $m \in \mathbb{Z}_n$.