Signal – Homework

Please answer the following questions. **We don't expect formal proofs: rather, just a high-level argument from intuition.** Please submit your answers as a **PDF** to Gradescope. Collaboration is allowed and encouraged, but you must write up your own answers and acknowledge your collaborators in your submission.

Due Date: *Monday, February 9th.*

1 Computational Security

Recall the distinction between perfect (information-theoretic) security and computational security that we discussed in class.

(1) In your own words, explain what is computational security and why we introduced this notion.

(2) When we say a cryptosystem sets the computational security parameter as $\lambda = 128$ (or achieves 128-bit security), what does it mean?

2 Cryptographic Schemes

(1) Why isn't the plain RSA encryption scheme CPA-secure?

(2) Why is the ElGamal encryption scheme CPA-secure?

(3) Why isn't the plain RSA signature scheme CMA-secure?

(4) Why does adding a cryptographic hash function to the plain RSA signature scheme make it CMA-secure?

3 Authenticated Encryption

Given a CPA-secure symmetric-key encryption scheme $\Pi_1 = (\mathsf{Gen}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ and a strongly CMA-secure MAC scheme $\Pi_2 = (\mathsf{Gen}_2, \mathsf{Mac}_2, \mathsf{Vrfy}_2)$, we describe three constructions for an authenticated encryption scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ below.

- Encrypt-and-MAC:
  - $\mathsf{Gen}(1^\lambda)$: run $k_1 \leftarrow \mathsf{Gen}_1(1^\lambda)$ and $k_2 \leftarrow \mathsf{Gen}_2(1^\lambda)$, and output $k = (k_1, k_2)$;
  - $\mathsf{Enc}_k(m)$: compute $c_1 \leftarrow \mathsf{Enc}_1(k_1, m)$ and $t_2 \leftarrow \mathsf{Mac}_2(k_2, m)$, and output $c = (c_1, t_2)$;
  - $\mathsf{Dec}_k(c = (c_1, t_2))$: compute $m := \mathsf{Dec}_1(k_1, c_1)$ and $b := \mathsf{Vrfy}_2(k_2, (m, t_2))$. If $b = 1$, then output m; otherwise output $\perp$.
- MAC-then-Encrypt:
  - $\mathsf{Gen}(1^\lambda)$: run $k_1 \leftarrow \mathsf{Gen}_1(1^\lambda)$ and $k_2 \leftarrow \mathsf{Gen}_2(1^\lambda)$, and output $k = (k_1, k_2)$;
  - $\mathsf{Enc}_k(m)$: compute $t_2 \leftarrow \mathsf{Mac}_2(k_2, m)$ and $c \leftarrow \mathsf{Enc}_1(k_1, m \| t_2)$, and output c;
  - $\mathsf{Dec}_k(c)$: compute $m \| t_2 := \mathsf{Dec}_1(k_1, c)$ and $b := \mathsf{Vrfy}_2(k_2, (m, t_2))$. If $b = 1$, then output m; otherwise output $\perp$.
- Encrypt-then-MAC:
  - $\mathsf{Gen}(1^\lambda)$: run $k_1 \leftarrow \mathsf{Gen}_1(1^\lambda)$ and $k_2 \leftarrow \mathsf{Gen}_2(1^\lambda)$, and output $k = (k_1, k_2)$;
  - $\mathsf{Enc}_k(m)$: compute $c_1 \leftarrow \mathsf{Enc}_1(k_1, m)$ and $t_2 \leftarrow \mathsf{Mac}_2(k_2, c_1)$, and output $c = (c_1, t_2)$;
  - $\mathsf{Dec}_k(c = (c_1, t_2))$: compute $m := \mathsf{Dec}_1(k_1, c_1)$ and $b := \mathsf{Vrfy}_2(k_2, (c_1, t_2))$. If $b = 1$, then output m; otherwise output $\perp$.

(1) Why is Encrypt-and-MAC *not* necessarily CPA-secure?

(2) (Extra Credit) Why is Encrypt-and-MAC unforgeable?

(3) Why is MAC-then-Encrypt *not* necessarily CCA-secure?

(4) (Extra Credit) Why is MAC-then-Encrypt unforgeable?

(5) (Extra Credit) Why is Encrypt-then-MAC both CCA-secure and unforgeable?

4 Potential Attacks

In this question, we exploit potential attacks on the Signal project.

(1) **Man-in-the-Middle (MitM) Attack.** Our protocol ensures that two parties can establish shared secret keys, but it does *not* ensure that they know exactly who they are talking to. Indeed, an adversary could pretend to be who they are talking to. Describe a man-in-the-middle attack that compromises the security.

(2) **Replay Attack.** Our protocol is *not* entirely secure against replay attacks. In particular, once a secure channel is established, the same encrypted messages could be sent multiple times (before the parties switch to a new key). For instance, consider an application that upon receiving a suitable message, will send a dollar to charity. Describe a replay attack that exploits this system, and propose a mechanism for protecting against this attack.

5 Cryptographic Notions

In your own words, explain the following cryptographic notions.

(1) What is a cryptographic hash function?

(2) What is a pseudorandom generator (PRG)?

(3) What is a pseudorandom function (PRF)? What is a pseudorandom permutation (PRP)?