CHOSEN PLAINTEXT ATTACKS (CPA) CONCEPTS

Overview:

Chosen-plaintext attacks capture the ability of an adversary to exercise (partial) control over what the honest parties encrypt.

In the formal definition, we model chosen-plaintext attacks by giving the adversary A access to an encryption oracle, viewed as a "black box" that encrypts messages of A's choice using a key k that is unknown to A. That is, we imagine A has access to an "oracle" Enc k (·); when A queries this oracle by providing it with a message m as input, the oracle returns a ciphertext c ← Enc k (m) as the reply. (If Enc is randomized, the oracle uses fresh randomness each time it answers a query.) The adversary can interact with the encryption oracle adaptively, as many times as it likes.

DEFINITION 3.21 *A private-key encryption scheme* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *has* indistinguishable encryptions under a chosen-plaintext attack, *or is* CPA-

secure, *if for all probabilistic polynomial-time adversaries* $\mathcal{A}$ *there is a negligible function* negl *such that*

$$\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\Pi}(n) = 1\right] \leq \frac{1}{2} + \mathsf{negl}(n),$$

where the probability is taken over the randomness used by $\mathcal{A}$*, as well as the randomness used in the experiment.*

Code Construction:

The implementation of the CPA-secure encryption scheme code follows the given algorithm.

CONSTRUCTION 3.28

Let F be a pseudorandom function. Define a fixed-length, private-key encryption scheme for messages of length n as follows:

- Gen: on input 1^n, choose uniform $k \in \{0,1\}^n$ and output it.
- Enc: on input a key $k \in \{0,1\}^n$ and a message $m \in \{0,1\}^n$, choose uniform $r \in \{0,1\}^n$ and output the ciphertext

$$c := \langle r, F_k(r) \oplus m \rangle.$$

- Dec: on input a key $k \in \{0,1\}^n$ and a ciphertext $c = \langle r, s \rangle$, output the message

$$m := F_k(r) \oplus s.$$

It can be proven that If F is a pseudorandom function, then the above construction is a CPA-secure, fixed-length private-key encryption scheme for messages of length n.The formal proof is as follows

Proof:

THEOREM 3.29 *If F is a pseudorandom function, then Construction 3.28 is a CPA-secure, fixed-length private-key encryption scheme for messages of length n.*

PROOF Let $\widetilde{\Pi} = (\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{Enc}}, \widetilde{\mathsf{Dec}})$ be an encryption scheme that is exactly the same as $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ from Construction 3.28, except that a truly random function f is used in place of F_k. That is, $\widetilde{\mathsf{Gen}}(1^n)$ chooses a uniform function $f \in \mathsf{Func}_n$, and $\widetilde{\mathsf{Enc}}$ encrypts just like Enc except that f is used instead of F_k. (This modified encryption scheme is not efficient. But we can still define it as a hypothetical encryption scheme for the sake of the proof.)

Fix an arbitrary PPT adversary $\mathcal{A}$, and let $q(n)$ be an upper bound on the number of queries that $\mathcal{A}(1^n)$ makes to its encryption oracle. (Note that q must be upper-bounded by some polynomial.) As the first step of the proof, we show that there is a negligible function negl such that

$$\left|\Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\Pi}(n) = 1\right] - \Pr\left[\mathsf{PrivK}^{\mathsf{cpa}}_{\mathcal{A},\widetilde{\Pi}}(n) = 1\right]\right| \leq \mathsf{negl}(n). \tag{3.9}$$

We prove this by reduction. We use $\mathcal{A}$ to construct a distinguisher D for the pseudorandom function F. The distinguisher D is given oracle access to a function $\mathcal{O}$, and its goal is to determine whether $\mathcal{O}$ is "pseudorandom" (i.e., equal to F_k for uniform $k \in \{0,1\}^n$) or "random" (i.e., equal to f for uniform $f \in \mathsf{Func}_n$). To do this, D simulates experiment $\mathsf{PrivK}^{\mathsf{cpa}}$ for $\mathcal{A}$ in the manner described below, and observes whether $\mathcal{A}$ succeeds or not. If $\mathcal{A}$ succeeds then D guesses that its oracle must be a pseudorandom function, whereas if $\mathcal{A}$ does not succeed then D guesses that its oracle must be a random function.