VIEWS AND A CHOSEN PLAINTEXT ATTACK ON A PSEUDORANDOM FUNCTION

BURTON ROSENBERG
UNIVERSITY OF MIAMI

Contents

1. Proof of indistinguishability for the ideal block cipher

The adversarial indistinguishability game is the interaction of two probabilistic polynomial time machines, the protocol Π and the adversary $\mathcal{A}$. A manner of thinking about that interaction is to consider the sequence of messages exchanged as a random variable, called the *transcript*. In the $[\Pi, \mathcal{A}]$ interaction, Π is in effect a sampling algorithm choosing from the distribution of possible transcripts. When we take this from the adversary's point of view, it is called the adversary's *view*, and the draw from the space of transcripts conditioned on all of the adversary's messages.

Refer to Figure 1 in the description of the protocol and the notational definitions that follow.

The protocol encrypts using a random function,

$$f : U_{l(n)} \to U_{l(n)},$$

where $U_{l(n)}$ is the space of strings over $l(n)$ bits with the uniform distribution, and $l(n)$ is a polynomial that gives the block size. The distinction between key size n and block size $l(n)$ has no meaning here, but is retained to agree with notation for pseudo-random functions F_k, which take an n bit key k.

Date: October 8, 2021.

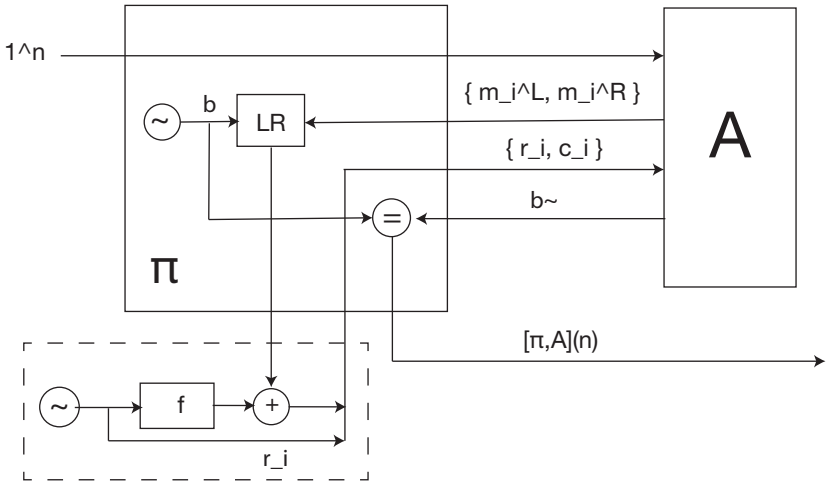

FIGURE 1. The CPA game on an Ideal Block Cipher

The transcript of messages is the sequence of message pairs from the adversary to the protocol, during a protocol run,

$$M = \langle\, m_i^L, m_i^R \,\rangle_i.$$

The transcript of responses by the protocol is the sequence of encryptions in the return message,

$$T = \langle\, r_i, c_i \,\rangle_i.$$

We will also refer to the sequence of random choices of the protocol, which is not sent to the adversary,

$$\tilde{T} = \langle\, r_i, f(r_i) \,\rangle_i.$$

We argue that the protocol is a sampling procedure for a random variable taking values in on of two spaces, depending on the coin flip. We will ultimately argue that the two random variables are identically distributed, so the protocol does not need to flip a coin in order to interact faithfully with the adversary. Therefore the adversary cannot predict the coin b, as the value of the coin is irrelevant for the interaction.

1.1. **A note on notation.** The angle signifies a sequence. For instance, a sequence $s_1, s_2, \ldots$ of elements from S is denoted $\langle\, s_i \,\rangle_i$. The subscript of the bracket helps identify which variable is the indexing variable, and reminds the reader that the shown element is just an example of a sequence of elements.

1.2. **A note on time bounds.** Both the protocol and the adversary are polynomial bound. The counting of time for the protocol will not include the encryption requests.

Despite any implications of the drawing, the adversary has oracle access to the LR–encryption, and the encryption is polynomial bound in the length of the presented messages.

In this way, the time bound of Π can be stated independently of the time bound of $\mathcal{A}$. Π provides the encryption oracle with a key of length n, and the single bit b. It receives from $\mathcal{A}$ the single bit $\tilde{b}$. The adversary's queries are all charged to its own data communication with the encryption oracle, and the encryption oracle charges one unit to the adversary for its use. The encryption oracle itself is polynomial time bound. In the case of the random function, the random function charges only the bit complexity of its input and output for its use.

2. Proof of security

We can fix the coin, and consider the queries of the adversary as part of the domain of a random variable. For the case the where protocol will always answer by encrypting the left message,

$$\begin{array}{rcccc} X_L & : & \Omega \times M & \rightarrow & T, \\ & & \omega, \langle\, m_i^L, m_i^R \,\rangle_i & \mapsto & \langle\, r_i, f(r_i) \oplus m_i^L \,\rangle_i \end{array}$$

and for the case where the protocol will always answer by encrypting the right message,

$$\begin{array}{rcccc} X_R & : & \Omega \times M & \rightarrow & T, \\ & & \omega, \langle\, m_i^L, m_i^R \,\rangle_i & \mapsto & \langle\, r_i, f(r_i) \oplus m_i^R \,\rangle_i \end{array}$$

Finally there is a random variable for the workings of the protocol,

$$\begin{array}{rcccc} X_{\tilde{T}} & : & \Omega \times M & \rightarrow & \tilde{T}, \\ & & \omega, \langle\, m_i^L, m_i^R \,\rangle_i & \mapsto & \langle\, r_i, f(r_i) \,\rangle_i \end{array}$$

Condition on the event $\mathcal{R}$ that each r_i is distinct, so that the random function f acts as a sequence of independent choices. Then,

$$Pr(X_{\tilde{T}} = \langle\, r_i, f(r_i) \,\rangle \,|\, R) = \prod_i Pr(U_{l(n)} = r_i) \wedge Pr(U_{l(n)} = f(r_i)).$$

The probability on X_L conditioned on event $\mathcal{R}$ is provided by the probability preserving bijection to $X_{\tilde{T}}$,

$$\begin{array}{rcccc} \phi_L & : & \tilde{T} \times M & \rightarrow & T, \\ & & \langle\, r_i, f(r_i) \,\rangle, \langle\, m_i^L, m_i^R \,\rangle & \mapsto & \langle\, r_i, f(r_i) \oplus m_i^L \,\rangle \end{array}$$

and ϕ_R is defined similarly.

Theorem 2.1.

$$Pr([\Pi, \mathcal{A}](n)) \leq 1/2 + \mathit{negl}(n)$$

.

Proof. Π samples either X_L or X_R. Conditioned on event $\mathcal{R}$, these are the same distributions, therefore $\mathcal{A}$ has no information on the bit b. For instance, Π can provide $\mathcal{A}$ with a perfectly correct interaction without even flipping the coin or looking at the messages, by returning samples from $X_{\tilde{T}}$. Hence,

$$Pr([\Pi, \mathcal{A}](n) \,|\, \mathcal{R}) = 1/2.$$

The result follows by the rule of total probability, and that $Pr(\sim \mathcal{R})$ is negligible,

$$\begin{aligned} Pr([\Pi, \mathcal{A}](n)) &= Pr([\Pi, \mathcal{A}](n) \,|\, \mathcal{R})\, Pr(\mathcal{R}) + Pr([\Pi, \mathcal{A}](n)) \,|\, \sim \mathcal{R})\, Pr(\sim \mathcal{R})) \\ &\leq (1/2)\,(1 - Pr(\sim \mathcal{R})) + Pr(\sim \mathcal{R}) \\ &\leq 1/2 + \mathit{negl}(n) \end{aligned}$$

□

3. For the Nonce

The proof did not need to contemplate the case of repeated blinding factors r_i, since the probability of this occurring is negligible. However, it the value is repeated, $r_i = r_j$ for distinct i and j, it is highly likely the adversary can win the game. Given $r_i = r_j$ the adversary will know for which i and j the encryption will cancel,

$$c_i \oplus c_j = m_i^b \oplus f(r_i) \oplus m_j^b \oplus f(r_j) = m_i^b \oplus m_j^b$$

If it happens that,

$$m_i^L \oplus m_j^L \neq m_i^R \oplus m_j^R$$

then the adversary definitely wins the game.

In practice, it is often required that the r_i be unique, and is called a *nonce*.[1] This can be achieved by building the number out of components whose combination is sure to be unique (assuming honest players!). For instance, an ethernet address, a time and a process identification number.

In terms of *concrete security*, a birthday attack is possible if the block size is too small, so that the r_i so repeat, out of chance. For an n bit number, a repeat is likely after $\sqrt{2^n}$ samples. For instance, a cipher with 64 bit block size would seem secure, but after only four billion packets we should begin to see collisions on the r_i, and hence a break in security.

[1] Wiktionary doubts that the etymology of the word is from "number used once", and thinks it is from Middle English "for the nonce", meaning "for the once".