Cryptology 27th August 2015

Lecture 6: Computational Security

Instructor: Goutam Paul *Scribe: Subhadip Singha*

1 Vernam Cipher (1917)

Definition 1.1 *Vernam Cipher is also called One-Time Pad(OTP), because each message must be encrypted with a different key. The one-time pad encryption scheme is defined as follows:*

1. *Fix an integer $l > 0$. Then the message space $\mathcal{M}$, key space $\mathcal{K}$, and ciphertext space $\mathcal{C}$ are all equal to $\{0,1\}^l$.*
2. *The key-generation algorithm Gen works by choosing a string k from $\{0,1\}^l$ according to uniform distribution.*
3. *Encryption Enc works as follows: given a key $k \in \{0,1\}^l$ and a message $m \in \{0,1\}^l$, outputs $c := k \oplus m$.*
4. *Decryption Dec works as follows: given a key $k \in \{0,1\}^l$ and a ciphertext $c \in \{0,1\}^l$, outputs $m := k \oplus c$.*

Let m_i, c_i and k_i be the i^{th} bit of the message, ciphertext and key respectively.
$\forall b \in \{0,1\}$ *and* $\forall b' \in \{0,1\}$,

$$\begin{aligned}
Pr[m_i = b \mid c_i = b'] &= \frac{Pr[m_i = b] \cdot Pr[c_i = b' | m_i = b]}{Pr[c_i = b']} \\
&= \frac{Pr[m_i = b] \cdot Pr[c_i = b' | m_i = b]}{\sum_j Pr[m_i = b] \cdot Pr[c_i = b' | m_i = b]} \\
&= \frac{Pr[m_i = b] \cdot Pr[c_i = b' | m_i = b]}{Pr[m_i = 0] \cdot Pr[c_i = b' | m_i = 0] + Pr[m_i = 1] \cdot Pr[c_i = b' | m_i = 1]} \\
&= \frac{Pr[m_i = b] \cdot Pr[k_i = b \oplus b']}{Pr[m_i = 0] \cdot Pr[k_i = b'] + Pr[m_i = 1] \cdot Pr[k_i = b' \oplus 1]} \\
&= \frac{Pr[m_i = b] \cdot 1/2}{Pr[m_i = 0] \cdot 1/2 + Pr[m_i = 1] \cdot 1/2} \\
&= Pr[m_i = b]
\end{aligned}$$

This implies perfect secrecy of Vernam Cipher.

1.1 Problems Associated with Vernam Cipher

1. Each message must be encrypted with different key. Otherwise, suppose $c_1 := m_1 \oplus k$ and $c_2 := m_2 \oplus k$, where $m_1, m_2 \in \mathcal{M}$, $c_1, c_2 \in \mathcal{C}$ and $k \in \mathcal{K}$
   This implies $m_1 \oplus m_2 := c_1 \oplus c_2$. An adversary can compute $c_1 \oplus c_2$ for any two observed ciphertexts. If there are n ciphertexts known to the adversary, it can compute the $\binom{n}{2}$ possible values of $m_1 \oplus m_2$. Hence it can be easily broken.

2. Sender needs to communicate the secret key to the receiver before sending each of the messages. As the length of the key and the message are same, both the parties need to establish a common key even before sending the new key for new message. So, this problem becomes circular in nature.

1.2 Remedies

1. Instead sending the secret key, use Random Number Generator (RNG) to generate random sequence (Key) at both the sides. This solution faces synchronization problem, as the RNGs generate true random numbers, it will not be possible for two RNGs generating same numbers each time.

2. Pseudo Random Number Generator (PRNG) can be used to generate random sequence (Key) at both the sides. This solution does not face synchronization problem, as the two parties generate a common session key which is fed to the PRNG as a seed to generate random sequence.

1.3 Random Number Generator (RNG)

The name is self explanatory. We can visualize RNG as a black box which generates l bit random numbers which can be used as the secrect key of OTP. So, the problem of OTP (establishing a shared secrect key) reduces to generate random numbers. If the numbers are true random numbers, then there will be no synchronization between the two parties who want to communicate using one secret key as stated above.

1.4 Pseudo Random Number Generator (PRNG)

A pseudorandom number generator (PRNG), also known as a deterministic random bit generator , is an algorithm for generating a sequence of numbers whose properties approximate the properties of sequences of random numbers. Loosely speaking, a pseudorandom string is a string that looks like a uniformly distributed string.

1.5 Secure PRNG

A secure pseudo-random number generator is a pseudo-random number generator (PRNG) with properties that make it suitable for use in cryptography. We assume the adversary is a probabilistic polynomial tme (PPT) algorithm, which has to distinguish between a random and a pseudo random string or number.

Definition 1.2 *A PRNG 'G' is said to be secure iff for all PPT adversary D, that*

$$Pr[\ D(G) = 1\ ] - \ Pr[\ D(R) = 1\ ]\ \leq\ \mathcal{E} \tag{1}$$

where $\mathcal{E}$ is a negligible function in 'n' and R is any RNG.

Definition 1.3 *A function f(n) is called negligible in n if for all polynomial p(n)*

$$f(n) < 1/p(n) \tag{2}$$

Equivalently, A function f(n) is called negligible in n if

$$\forall c > 0,\ \exists N,\ s.t.\ \forall n \geq N,\ f(n) < 1/n^c \tag{3}$$

2 Test of (Non)Randomness

Randomness tests are used to analyze the distribution pattern of a set of data. A good PNRG should pass these tests. National Institute of Standards and Technology (NIST) has 16 standard statistcal tests available on their website `www.nist.gov`

3 Comparison between Information Theoretic and Computational notion of Cryptography

1. *The efficiency or running time of an adversary is unlimited in case of Information Theoretic notion but Probabilistic Polynomial time in case of Computational notion.*
2. *The success probability in the indistinguishability game is equal to the randomness associated as per Information Theoretic notion but in case of Computational Notion, it is negligible from the associated randomness.*

4 A Computationally Secure Encryption

$$Pr\ [\ Adversary\ wins\ the\ game\ ] \leq 1/2 + \mathcal{E} \tag{4}$$

where $\mathcal{E}$ is a negligible function in 'n'.

Note: Crypto System can be categorized into two broad categories, e.g. Private (Symmetric) and Public (A-symmetric) key cryptography and Symmetric key cryptography can be devided into Block Cipher and Stream Cipher. Block Ciphers are Pseudo Random Permutations (PRP) and Stream Ciphers are Pseudo Random Number Generators (PRNG).

5 Pseudo Random Permutation (PRP)

Definition 5.1 *Let F be a mapping $\{0,1\}^n \times \{0,1\}^k \rightarrow \{0,1\}^n$, F is a PRP if*

1. $\forall\ K \in \{0,1\}^k$ *, F is a bijection from* $\{0,1\}^n$ *to* $\{0,1\}^n$.
2. $\forall\ K \in \{0,1\}^k$*, there is an 'efficient' algorithm to evaluate* $F_k(x)$.

A pseudorandom permutation family is a collection of pseudorandom permutations, where a specific permutation may be chosen using a key.

Note: As per the definition, the cardinality of the set of all permutation is $\frac{(2^n)!}{}$ *but based on the secret key, only one of* 2^k *prmutations will be selected. For a good PRP, it will be difficult for an adversary to distinguish between a pseudo random permutation and a true random permutation.*

6 Pseudo Random Function (PRF)

Definition 6.1 *Let F be a mapping* $\{0,1\}^n \times \{0,1\}^k \rightarrow \{0,1\}^n$*, F is a PRF if*

1. $\forall\ K \in \{0,1\}^k$ *, F is a function from* $\{0,1\}^n$ *to* $\{0,1\}^n$ *.*
2. $\forall\ K \in \{0,1\}^k$*, there is an 'efficient' algorithm to evaluate* $F_k(x)$*.*

A pseudorandom function family is a collection of pseudorandom functions, where a specific function may be chosen using a key.

Note: As per the definition, the cardinality of the set of all function is l^l *(where* $l = 2^n$*) but based on the secret key, only one of* 2^k *functions will be selected. For a good PRF, it will be difficult for an adversary to distinguish between a pseudo random function and a true random function.*