CSE 594 : Modern Cryptography 02/28/2017

Lecture 10: Symmetric Encryption

Instructor: Omkant Pandey *Scribe: Hyungjoon Koo, FNU Gaurav*

1 Symmetric Encryption

Assume that Alice and Bob share a secret $s \in \{0,1\}^n$ and Alice wants to send a private message m to Bob. We want to achieve both correctness and security, which means no eavesdropper could reveal their message but Alice and Bob. In this setting, Alice encodes m to c with s. Likewise, Bob decodes c using s to obtain the correct m. The definition of symmetric encryption can be set up as following.

$$\text{Gen}(1^n) \to s$$
$$\text{Enc}(s, m) \to c$$
$$\text{Dec}(s, c) \to m' \text{or } \perp$$

All algorithms are PPT in n, known as a security parameter. With the parameter of your choice separately, you can sit on the balance between security and efficiency. Sometimes you may want to run your system faster by adjusting the parameter.

For correctness of the definition, we can compute $\text{Dec}(s, \text{Enc}(s, m)) = m \quad \forall m \text{and} s$, where $s \leftarrow \text{Gen}(1^n)$. For security it is computationally indistinguishable. In other words, an adversary cannot tell if m_0 or m_1 was encrypted by looking at c.

Definition 1 *Indistinguishability security A symmetric encryption scheme (Gen, Enc, Dec) is secure if for all n.u. PPT adversaries A, there exists a negligible function $\mu(\cdot)$ s.t.*

$$Pr[s \leftarrow \text{Gen}(1^n), (m_0, m_1) \leftarrow A(1^n), b \leftarrow \{0,1\} : A(\text{Enc}(s, m_b)) = b] \leq \frac{1}{2} + \mu(\cdot)$$

In this scenario, we assume all system including encryption, decryption, and key generation algorithm, is known to our adversary but the key, secret.

Definition 2 *Indistinguishability security (alternative) A symmetric encryption scheme (Gen, Enc, Dec) is secure if $\forall m_0, m_1$:*

$$\{\text{Enc}(s, m_0) : s \leftarrow \text{Gen}(1^n)\} \approx \{\text{Enc}(s, m_1) : s \leftarrow \text{Gen}(1^n)\}$$

We can convert this definition to another that is computational analogue of perfect secrecy. These two definitions are equivalent with a normalization factor 2 in terms of "prediction advantage" versus "computational indistinguishability". Here one cannot tell the two distributions apart computationally within polynomial time.

With one-time pads, indistinguishability security can be written as following:

$$\text{Gen}(1^n) := s \leftarrow \{0,1\}^n$$
$$\text{Enc}(s, m) := m \oplus s$$
$$\text{Enc}(s \leftarrow \{0,1\}^n, m_0) \equiv \text{Enc}(s \leftarrow \{0,1\}^n, m_1)$$

2 Encryption using PRGs

Now how can we encrypt messages longer than n bits? With poly-stretch PRG, m can be polynomially long. Like one-time pads, first generate the key s from key generation algorithm. Next, compute encryption by *xoring* m with $PRG(s)$ instead of s. In this setting, the distinguisher D cannot tell two ciphers apart within polynomial time.

$$\text{Gen}(1^n) := s \leftarrow \{0,1\}^n$$

$$\text{Enc}(s, m) := m \oplus PRG(s)$$

$$\text{Enc}(s \leftarrow \{0,1\}^n, m_0) \approx \text{Enc}(s \leftarrow \{0,1\}^n, m_1)$$

Proof. We can prove the above via hybrids.

- $H_0 : \text{Enc}(s, m_0) = m_0 \oplus \text{PRG}(s)$
- $H_1 : \text{Enc}(s, m_0) \approx m_0 \oplus \text{R}$
- $H_2 : \text{Enc}(s, m_1) \approx m_1 \oplus \text{R}$
- $H_3 : \text{Enc}(s, m_1) = m_1 \oplus \text{PRG}(s)$

Let us replace PRG with R, a random string under one-time pads which is perfectly secure. When giving H_0 and H_1 to the distinguisher D, it can be distinguishable at most ϵ, security of PRF ($H_0 \approx H_1$) because of PRG $\approx$ R. Likewise, H_2 and H_3 are indistinguishable in a polynomial time ($H_2 \approx H_3$). Hence total prediction advantage is at most 2ϵ, which is negligible. Note that H_1 and H_2 are identical.

3 Stream Ciphers: Encryption using PRGs

So far, we have considered the encryption once with a single PRG. It only addresses the size of the key. How can we encrypt more than one message? It is called stream ciphers, which is another name for "encryption with a PRG (or PRNG)". Recall our PRG stretch construction looks like this: $G(s_0 = s) = b_1||s_1 \rightarrow G(s_1) = b_1||s_2 \rightarrow G(s_2) = b_3||s_3 \rightarrow ...$ However, the design of stream ciphers is different in practice so that it works much faster. Many stream ciphers have been broken or known weaknesses. For example, the RC4 was broken because of its biases in initial output. The CSS for DVD encryption was also badly broken. Yet SOSEMANUK and Salsa20 have not been broken.

4 Multi-message Secure Encryption

We now have an encryption scheme which shows indistinguishability, assuming PRGs exist. But there is one thing which is unrealistic about this scheme that how is communication supposed to happen with one message. In reality, one expects to exchange multiple messages over time. For that, we need an encryption technique that enables security of multiple messages while maintaining the principle of indistinguishability. It seems that our source of trouble is that we are using the same portion of the PRG output to encrypt multiple messages. One way to counter this is by using

some 'm' bits to encrpyt the first message, the next 'm' bits for second message and so on. But this solution requires us to maintain state at both encoding as well decoding levels i.e. to decode any j^{th} message we need to know the exact position so in order to start decoding at the correct place. Such a state may or may not be maintained. The solution to this is to design a stateless encryption scheme such that there is randomness supplementing our previous encryption scheme in order to beat the shortcomings of it being deterministic.

A symmetric encryption scheme (Gen, Enc, Dec) is multi-message secure if for all n.u. PPT adversaries A, for all polynomials $q(.)$, there exists a negligible function $\mu(.)$ s.t.:

$$Pr[s \xleftarrow{\$} Gen(1^n), \{(m_0^i, m_1^i)\}_{i=1}^{q(n)} \leftarrow A(1^n), b \xleftarrow{\$} \{0,1\} : A(\{Enc(m_b^i)\}_{i=1}^{q(n)}) = b] \leqslant 1/2 + \mu(n)$$

4.1 Necessity of Randomized (Probabilistic) Encryption

A multi-message secure encryption scheme cannot be deterministic and stateless. Randomized encryption is the use of randomness in an encryption algorithm, so that when encrypting the same message several times it will, in general, yield different ciphertexts. The term "randomized encryption" is typically used in reference to public key encryption algorithms, however various symmetric key encryption algorithms achieve a similar property (e.g., block ciphers when used in a chaining mode such as CBC). To be semantically secure, that is, to hide even partial information about the plaintext, an encryption algorithm must be randomized.

$$\{Enc(s, m_0) = c_0; Enc(s, m_0') = c_0'\} \leftarrow D \rightarrow \{Enc(s, m_1) = c_1; Enc(s, m_1') = c_1'\}$$

4.2 Encryption using PRFs (Pseudo Random Functions)

Let $f_s : \{0,1\}^n \rightarrow \{0,1\}^n$ be a family of PRFs where (Gen, Enc, Dec) is a multi-message secure encryption scheme:

- $Gen(1^n) : s \xleftarrow{\$} \{0,1\}^n$
- $Enc(s, m)$: Pick $r \xleftarrow{\$} \{0,1\}^n$; Output $(r, m \oplus f_s(r))$
- $Dec(s, (r, c))$: Output $(c \oplus f_s(r))$

4.2.1 Proof via hybrids

To prove - If (Gen, Enc, Dec) is $(s, 2\epsilon)$ is message indistinguishable, then that same scheme is $(s - nm, 2n\epsilon)$ message indistinguishable for n messages

- H_1: Real experiment with $m_0^1, ..., m_0^{q(n)}$ (i.e., b=0)
- H_2: Replace f_s with random function $f \xleftarrow{\$} F_n$
- H_3: Switch to one-time pad encryption
- H_4: Switch to encryption of $m_1^1, ..., m_1^{q(n)}$

- H_5: Use random function $f \xleftarrow{\$} F_n$ to encrypt
- H_6: Encrypt using f_s. Same as real experiment with $m_0^1, ..., m_0^{q(n)}$ (i.e., b=1)

5 Semantic Security

A symmetric encryption scheme (Gen, Enc, Dec) is semantically secure if for every A there exists a PPT algorithm S (the simulator) s.t. the following two experiments are computationally indistinguishable:

$$\left\{ \begin{array}{r} (m,z) \leftarrow A(1^n), \\ s \leftarrow Gen(1^n), \\ Output(Enc(s,m),z) \end{array} \right\} \overset{c}{\approx} \left\{ \begin{array}{c} (m,z) \leftarrow A(1^n), \\ OutputS(1^n,z) \end{array} \right\}$$

where A is an "adversarial" machine that samples a message from the message space and arbitrary auxiliary information.

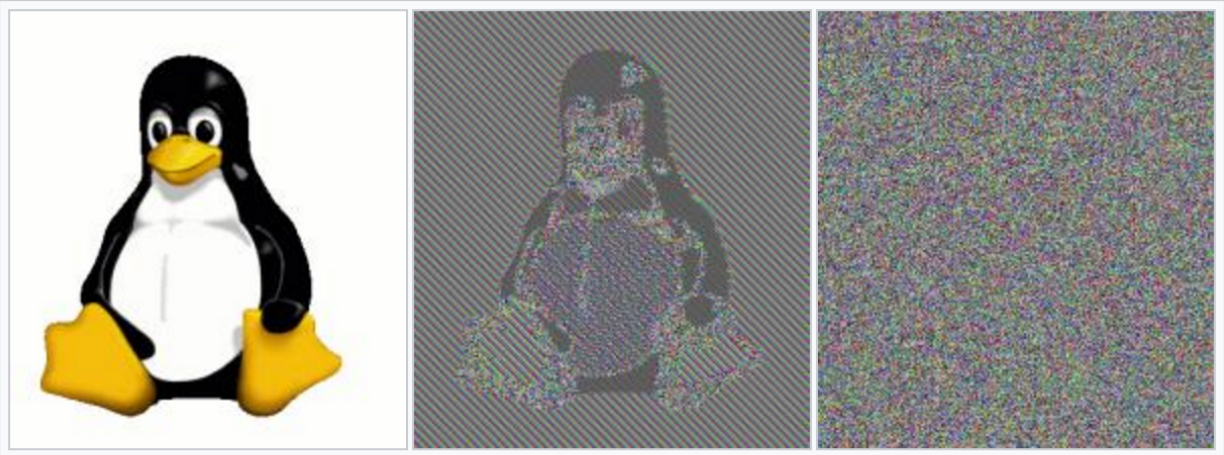
In other words, knowledge of the ciphertext (and length) of some unknown message does not reveal any additional information on the message that can be feasibly extracted. This concept is the computational complexity analogue to Shannon's concept of perfect secrecy. Perfect secrecy means that the ciphertext reveals no information at all about the plaintext, whereas semantic security implies that any information revealed cannot be feasibly extracted.

Indistinguishability security $\Leftrightarrow$ Semantic security

6 Block Ciphers

In cryptography, a block cipher is a deterministic algorithm operating on fixed-length groups of bits, called blocks, with an unvarying transformation that is specified by a symmetric key. Block ciphers operate as important elementary components in the design of many cryptographic protocols, and are widely used to implement encryption of bulk data.

- Encrpyt blocks (say 64-bit) instead of bits as in stream ciphers
- AES is a block cipher
- Block ciphers does not yield encryption directly
- The cipher comes with many "encryption modes" to encrypt arbitrarily long messages
- Weakest example: ECB (Electronic Code Book) as there are identifiable patterns in this cipher as seen in figure 1
- Other examples: CBC (Cipher Block Chaining) as seen in figure 2, PCBC (Propagating Cipher Block Chaining - type of CBC), CFB (Cipher Feedback), OFB (Output Feedback), CTR (Counter) etc.



Original image | Encrypted using ECB mode | Modes other than ECB result in pseudo-randomness

The image on the right is how the image might appear encrypted with CBC, CTR or any of the other more secure modes—indistinguishable from random noise. Note that the random appearance of the image on the right does not ensure that the image has been securely encrypted; many kinds of insecure encryption have been developed which would produce output just as "random-looking".

Figure 1: ECB Shortcoming - Source: `https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation`

6.1 Cipher Block Chaining

CBC has been the most commonly used method. Its main drawbacks are that encryption is sequential (i.e., it cannot be parallelized), and that the message must be padded to a multiple of the cipher block size. One way to handle this last issue is through the method known as ciphertext stealing. Note that a one-bit change in a plaintext or IV (Initialization Vector) affects all the following ciphertext blocks.

If the first block has index 1, the mathematical formula for CBC encryption is
$C_i = E_K(P_i \oplus C_{i-1})$,
$C_0 = IV$.
while the mathematical formula for CBC decryption is
$P_i = D_K(C_i) \oplus C_{i-1}$,
$C_0 = IV$.

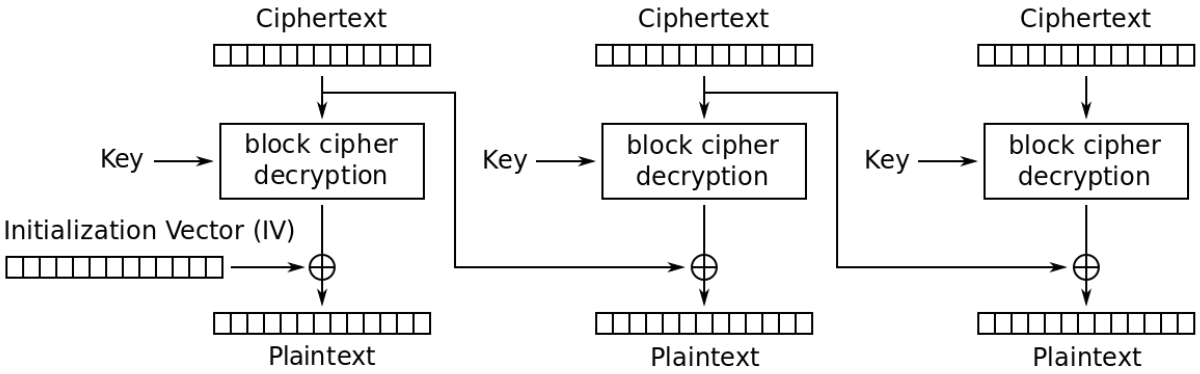


Figure 2: Cipher Block Chaining - Source: `https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation`