Introduction to Modern Cryptography
Summary

WS20/21

Valentin Knappich

January 26, 2021

Contents

1 Symmetric encryption

Kerkhoffs Principle: The security of a system should only depend on whether the actual key is secret, not on the system itself. The whole system is assumed to be public. No "Security by obscurity".

1.1 Scenario 1

One message with constant length

1.1.1 Cryptosystems

A cryptosystem is a tuple $\mathcal{S} = (X, K, Y, e, d)$ with

- X: set of plaintexts
- K: finite set of keys
- Y: set of ciphertexts
- e: encryption function
- d: decryption function

Perfect correctness: $d(e(x,k),k) \forall x \in X, k \in K$

No unnecessary ciphertexts: $Y = \{e(x,k) | x \in X, k \in K\}$

1.1.2 Vernam system

The Vernam cryptosystem of length l is defined as $(\{0,1\}^l, \{0,1\}^l, \{0,1\}^l, e, d)$ where

$e(x,k) = x \oplus k$ and $d(y,k) = y \oplus k$.

A vernam system of length $l > 0$ provides perfect secrecy for every uniform P_K. It is the perfect system for Scenario 1.

1.1.3 Perfect Secrecy

A cryptosystem with key distribution $\mathcal{V} = \mathcal{S}[P_k]$ provides perfect secrecy if for all plaintext distributions P_X, the probability of every plaintext remains the same, i.e.:

$$P(x) = P(x|y) \quad \forall x \in X, y \in Y, P(y) > 0$$

Example Proof:

We need to show the criteria above for all plaintext distributions P_X. Therefore we use variable probabilities for the plaintexts $P_X(a) = p, P_X(b) = 1 - p$ (for 2 plaintexts, else $p_1, ..., p_n$).

K \ X		a	b
$\frac{1}{2}$	k_0	A	B
$\frac{1}{2}$	k_1	B	A

$$P(a|A) = \frac{P(a,A)}{P(A)} = \frac{\frac{1}{2} * p}{\frac{1}{2} * p + \frac{1}{2} * (1-p)} = p = P(a)$$

$$P(a|B) = \frac{P(a,B)}{P(B)} = \frac{\frac{1}{2} * p}{\frac{1}{2} * p + \frac{1}{2} * (1-p)} = p = P(a)$$

$$P(b|A) = \frac{P(b,A)}{P(A)} = \frac{\frac{1}{2} * (1-p)}{\frac{1}{2} * (1-p) + \frac{1}{2} * p} = 1 - p = P(b)$$

$$P(b|B) = \frac{P(b,B)}{P(B)} = \frac{\frac{1}{2} * (1-p)}{\frac{1}{2} * (1-p) + \frac{1}{2} * p} = 1 - p = P(b)$$

Theorem:

Let $\mathcal{S} = (X, K, Y, e, d)$ be a cryptosystem providing perfect secrecy, then it holds $|K| \geq |Y| \geq |X|$.

Shannons Theorem:

Let $\mathcal{V} = \mathcal{S}[P_k]$ be a cryptosystem with key distribution P_K and $|K| = |Y| = |X|$. The system provides perfect secrecy if and only if

1. P_K is a uniform distribution
2. $\forall x \in X, y \in Y \exists k \in K$ with $e(x,k) = y$ (There must be a key for every plaintext/ciphertext pair)

1.2 Scenario 2

Multiple messages with constant length, no repetition

1.2.1 Vernam in Scenario 2

Vernam is not a secure cryptosystem anymore, since from 2 ciphertexts, Eve can learn non-trivial information about the plaintexts:

$$y_0 \oplus y_1 = x_0 \oplus k \oplus x_1 \oplus k = x_0 \oplus x_1$$

Also with 1 plaintext-ciphertext pair (CPA), the key can be calculated as $k = x \oplus y$.

1.2.2 Substitution Cryptosystem

Let X be a non-empty finite set. A substitution cryptosystem over X is a tuple (X, P_X, X, e, d) where P_X is the set of all permutations of X.

$$e(x, \pi) = \pi(x) \quad d(y, \pi) = \pi^{-1}(y) \quad \forall x, y \in X, \pi \in P_X$$

Substitution cryptosystems provide "perfect security" in scenario 2, BUT they are impractical because the substitution table (π) has a size of $2^l * l$.

Therefore, we need a weaker security definition that takes into account, that attackers are resource bound.

1.2.3 l-Block Cipher

Let $l : \mathbb{N} \to \mathbb{N}$ be a polynomial. An l-block cipher B is a cryptosystem of the form

$\left(\{0,1\}^{l(\eta)}_{\eta\in\mathbb{N}},\ Gen(1^\eta),\ \{0,1\}^{l(\eta)}_{\eta\in\mathbb{N}},\ E,\ D\right)$ or simplified: $\left(\{0,1\}^{l},\ Gen(1^\eta),\ \{0,1\}^{l},\ E,\ D\right)$

1.2.4 Substitution-Permutation Cryptosystem (SPCS)

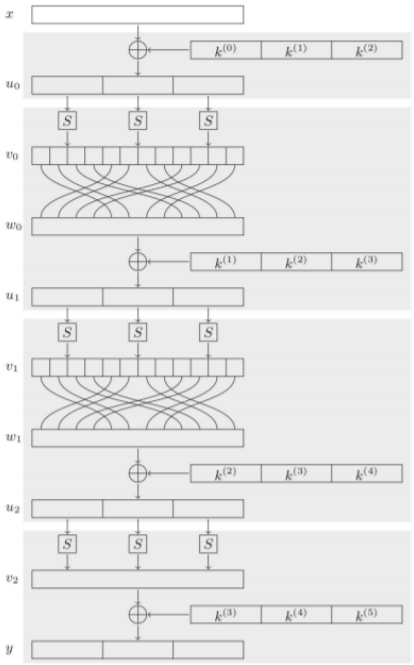
Notation:

- plaintexts are split into m words with length n with $l = m * n$, $x^{(i)}$ denotes the i'th word
- $[r] = \{0, 1, ..., r-1\}$
- $\beta \in \mathcal{P}_{[l]}$, then $x^\beta(i) = x(\beta(i))$

General Principle: Over r rounds, (round) key additions, word substitutions and bit permutations are applied, including an initial step that just applies key addition and shortened last round without bit permutation.

$E(x : \{0,1\}^{mn}, k : \{0,1\}^s) : \{0,1\}^{mn}$

1. *initial white step (round key addition)*
   $u = x \oplus K(k, 0)$
2. $r-1$ *regular rounds*
   for $i = 1$ to $r-1$ do
   a. *word substitutions*
      for $j = 0$ to $m-1$ do
      $v^{(j)} = S(u^{(j)})$
   b. *bit permutation*
      $w = v^\beta$
   c. *round key addition*
      $u = w \oplus K(k, i)$
3. *shortened last round (without bit permutation)*
   for $j = 0$ to $m-1$ do
   $v^{(j)} = S(u^{(j)})$
   $y = v \oplus K(k, r)$; return y



Known Attacks:

- Brute Force Attack
- Linear Cryptanalysis
- Differential Cryptanalysis

Linear Cryptanalysis:

- Relies on a set T of plaintext-ciphertext pairs
- Instead of brute forcing the whole key, get small parts of the key at a time

- TODO

AES (Advanced encryption standard): basically SPCS with modifications

1.2.5 Algorithmic Security of Block Ciphers

We consider a block cipher secure if it is almost as good as a substitution cryptosystem w.r.t. resource-bound adversaries. Therefore an adversary U has to be able to distinguish BCS and SCS. Formally, we use the BCS for $b = 1$ (real world) and the SCS for $b = 0$ (random world) in the security game.
The winning probability is $Pr[\mathbb{E}(1^n) = 1]$. Since a random guesser already has a probability of 0.5, the advantage is introduced to normalize.

$\mathbb{S}(1^\eta) : \{0,1\}$
1. *Choose real world or random world.*
   $b \xleftarrow{\$} \{0,1\}$
   if $b = 1$ then
   $k \xleftarrow{\$} \mathsf{Gen}(1^\eta)$ and $F = E(\cdot, k)$
   else
   $F \xleftarrow{\$} \mathcal{P}_{\{0,1\}^{l(\eta)}}$
2. *Guess phase.*
   $b' \xleftarrow{\$} U(1^\eta, F)$
3. *Output.*
   return b'.

$$Adv_{U,B}(\eta) = 2 * \left(Pr[\mathbb{E}_U^B(1^\eta) = 1] - \frac{1}{2} \right) \in [-1, 1] \qquad suc_{U,B}(\eta) = Pr[\mathbb{S}_U^B\langle b = 1\rangle(1^\eta) = 1]$$

$$Adv_{U,B}(\eta) = suc_{U,B}(\eta) - fail_{U,B}(\eta) \qquad fail_{U,B}(\eta) = Pr[\mathbb{S}_U^B\langle b = 0\rangle(1^\eta) = 1]$$

1.2.6 PRP/PRF Switching Lemma

Since substitution cryptosystems cannot be distinguished from (secure) l-Block cryptosystems, we can see l-Block cryptosystems as pseudo-random permutations (PRP). Anyway, for proving purposes, it can be easier to see them as pseudo-random functions. The PRP/PRF Switching Lemma says, that we can use them interchangeably, since the difference of advantages is negligible:

Let B be an l-block cipher and U be an l-distinguisher with runtime bound $q(\eta)$ where q is a positive polynomial and $\eta \in \mathbb{N}$. Then the following holds true:

$$|Adv_{U,B}^{PRP}(\eta) - Adv_{U,B}^{PRF}(\eta)| \leq \frac{q(\eta)^2}{2^{l(\eta)+1}}$$

1.3 Scenario 3

Arbitrary messages with any length (possibly with repetition)

1.3.1 Symmetric Encryption Scheme

A symmetric encryption scheme is a tuple $S = (Gen(^\eta), E, D)$ with

- security parameter η
- ppt key generation algorithm $Gen(1^\eta)$
- ppt encryption algorithm $E(x : \{0,1\}^*, k : K) : \{0,1\}^*$
- dpt decryption algorithm $D(y : \{0,1\}^*, k : K) : \{0,1\}^*$

- and $D(E(x,k),k) = x$

E cannot be deterministic, because else we wouldn't be able to send the same message multiple times, i.e. the same plaintext encrypted under the same key should result in a different ciphertext (with a high probability).

1.3.2 Encryption Schemes from Stream Ciphers

Idea: Vernam is safe if we use every key just once. So using the key as seed of a random number generator, that generates a stream of random numbers, enables the usage of the vernam system for arbitrarily long messages.

1.3.2.1 Number generator

A number generator (NG) is a dpt algorithm of the Form $G : (s : \{0,1\}^\eta) : \{0,1\}^{p(\eta)}$ where p is the expansion factor.

1.3.2.2 PRNG-Distinguisher

TODO

1.3.3 Encryption Schemes from Block Ciphers

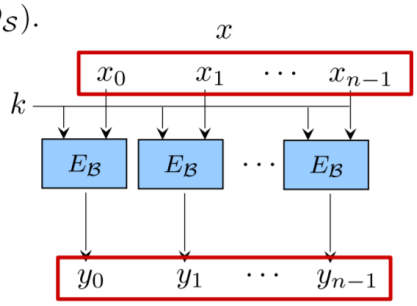
1.3.3.1 ECB Mode

Idea: Split the message in blocks of constant length and encrypt each block under the given key using the underlying block cipher.

$$\mathcal{S} = \mathsf{ECB}\text{-}\mathcal{B} = (\mathsf{Gen}_\mathcal{B}(1^\eta), E_\mathcal{S}, D_\mathcal{S}).$$

$E_\mathcal{S}(x : \{0,1\}^{l(\eta)+}, k : K_\mathcal{B}) : \{0,1\}^*$:

1. Split x into several blocks of length $l(\eta)$:
   $x =: x_0 || \cdots || x_{n-1}, n \in \mathbb{N}, x_i \in \{0,1\}^{l(\eta)}$
2. $y_i = E_\mathcal{B}(x_i, k) \quad \forall i \in \{0, \ldots, n-1\}$
3. **return** $y := y_0 || \ldots || y_{n-1}$

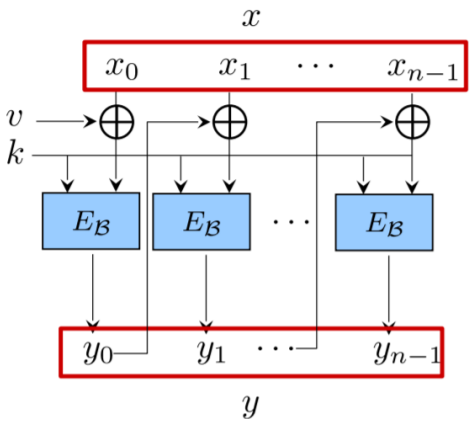


Security: It's not secure, since the ciphertext carries non-trivial information about the plaintext: for $y = y_0 || y_1$, then $y_0 = y_1$ if $x_0 = x_1$.

1.3.3.2 CBC Mode

Idea: Add and initialization vector v that is `xor`'ed with the plaintext before encrypting. That v is part of the key.

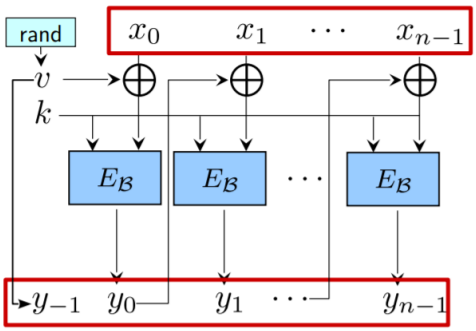
Problem: Still deterministic, so every plaintext can be sent just once.



1.3.3.3 R-CBC Mode

Idea: To solve the issues of CBC-Mode, R-CBC moves the initialization vector v out of the key and generates a random one while decryption. The vector is appended as first block of the ciphertext to enable decryption.

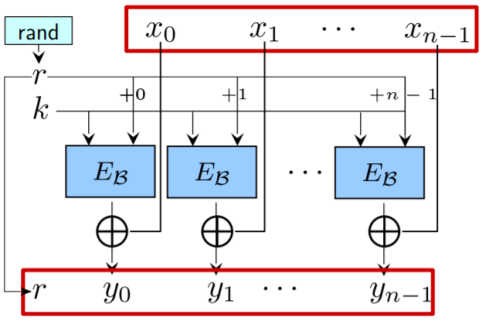
Security: Its secure if the underlying block cipher is secure.



1.3.3.4 R-CTR Mode

Idea: Alternative to R-CBC. Generate a random number r (comparable to v of R-CBC), encrypt this random number under the key and xor it with the plaintext. The counter is increased by 1 for each block. The counter r is appended as first block of y to enable decryption.

Security: Its secure if the underlying block cipher is secure.



1.3.4 CPA-Security

CPA: Chosen-Plaintext-Attack

Game: Adversary A consists of finder AF and guesser AG. The finder chooses 2 plaintexts z_0, z_1. One of them is encrypted. The guesser has to determine which of them is the corresponding plaintext.

Advantage, success and failure are defined as for block ciphers.

$\mathbb{E}(1^\eta) : \{0,1\}$

1. *Choose cipher.*
   $k \xleftarrow{\$} \mathsf{Gen}(1^\eta)$; $H = E(\cdot, k)$
2. *Find phase.*
   $(z_0, z_1) \xleftarrow{\$} AF(1^\eta, H)$
3. *Selection.*
   $b \xleftarrow{\$} \{0,1\}$; $y \xleftarrow{\$} H(z_b)$
4. *Guess phase.*
   $b' \xleftarrow{\$} AG(1^\eta, H, y)$
5. *Evaluation.*
   if $b' = b$, return 1, otherwise 0.

1.3.5 CCA-Security

CCA: Chosen-Ciphertext-Attack

Game: In addition to the encryption oracle H from the CPA-game, the adversary also gets a decryption oracle H^{-1}.

Advantage, success and failure are defined as for block ciphers.

$\mathbb{E}(1^\eta) : \{0,1\}$

1. *Choose cipher.*
   $k \xleftarrow{\$} \mathsf{Gen}(1^\eta)$; $H = E(\cdot, k)$
2. *Find phase.*
   $(z_0, z_1) \xleftarrow{\$} AF(1^\eta, H)$
3. *Selection.*
   $b \xleftarrow{\$} \{0,1\}$; $y \xleftarrow{\$} H(z_b)$
4. *Guess phase.*
   $b' \xleftarrow{\$} AG(1^\eta, H, y)$
5. *Evaluation.*
   if $b' = b$, return 1, otherwise 0.