Cryptography

Lecture 10: Hash Functions

December 17, 2024

Overview

Contents

1. Classical cryptography
   (Shift & Vigenère cipher, one-time pad, perfect secrecy)
2. Security definitions & threat models
   (Computational security, CPA & CCA)
3. Private-key cryptography
   (Message authentication, hash functions, primitives, relevant ciphers)
4. Public-key cryptography
   (Assumptions, key management, digital signatures, relevant ciphers)

Proposals for Pseudorandom Functions — AES

AES

- Block cipher "Advanced Encryption Standard" (successor of DES)
- Extremely popular due to its speed & simplicity (hardware support in most processors)
- Standard implementation of "pseudorandom permutation"
- Utilizes substitution-permutation networks

Characteristics of AES

- Block size 128 bits and key size 128, 192, or 256 bits
- 10-, 12-, or 14-round substitution-permutation network
- NIST (FIPS 197) & ISO (IEC 18033-3:2010) standard

Proposals for Pseudorandom Functions — AES

Vincent Rijmen (* 1970)

- Belgian cryptographer
- Professor at K.U. Leuven & University of Bergen
- IACR Fellow & MIT Top-100 innovator under 35

© Vincent Rijmen

Joan Daemen (* 1965)

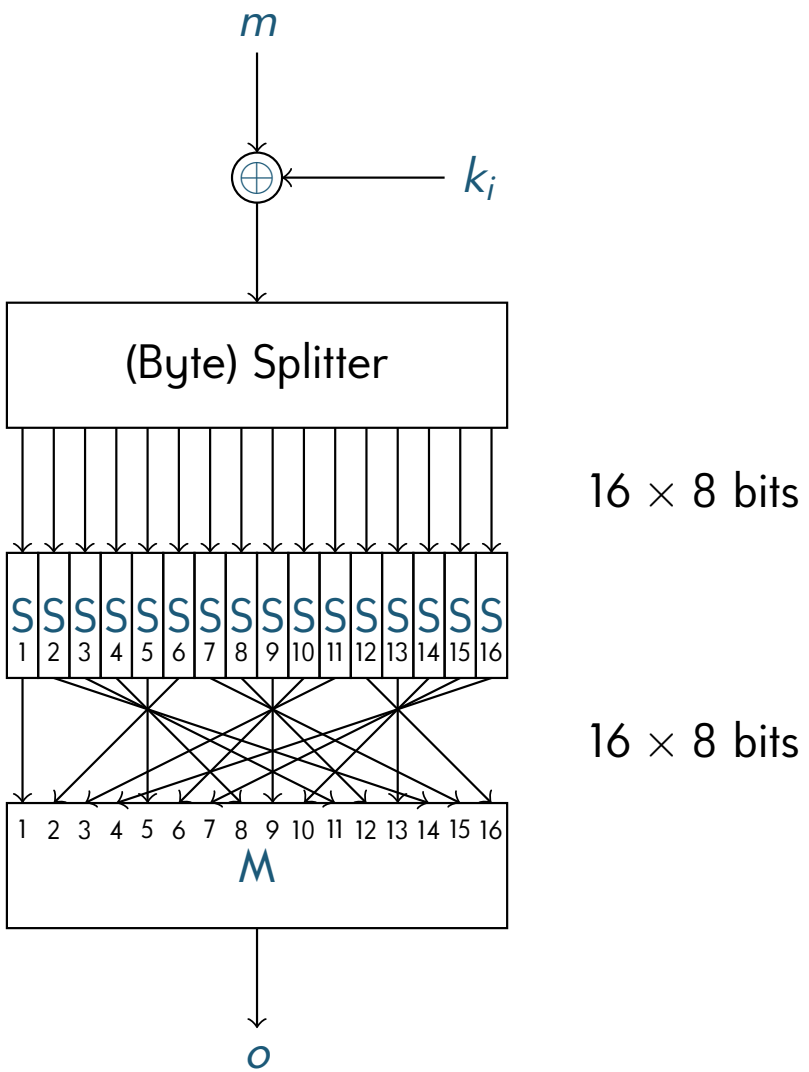
- Belgian cryptographer
- Professor at Radbound University Nijmegen
- Co-designed SHA-3 & Levchin prize in 2017

© Radbound University

Proposals for Pseudorandom Functions — AES

Substitution-permutation networks

- Bit-wise XOR with round key k_i
- Split into small chunks
  (AES: split 128 bits into 16 $\times$ 8 bits)
- Confusion part
  Bijective non-linear substitution
  (AES: single substitution S)
- Diffusion part
  Bijective linear transformation
  (AES: shuffle + linear transform)



Excursion: Hash Functions

Excursion: Hash Functions

§9.3 Definition (Hash function)

Let p be polynomial such that $p(n) > n$ for all $n \geq 1$ (expansive).
Let $\ell, u \colon \mathbb{N} \to \mathbb{N}$ be such that $\ell(n) \leq u(n)$ and $p(n) < u(n)$ for all $n \in \mathbb{N}$.
(Unkeyed) hash function (with hash length p and limits ℓ and u) is deterministic polynomial-time function $h \colon \{1\}^* \times \{0,1\}^* \to \{0,1\}^*$ such that $h(1^n, m) \in \{0,1\}^{p(n)}$ for all $n \in \mathbb{N}$ and $m \in \{0,1\}^*$ with $\ell(n) \leq |m| \leq u(n)$.
Hash functions with $\ell = u$ are called compression functions.

Notes

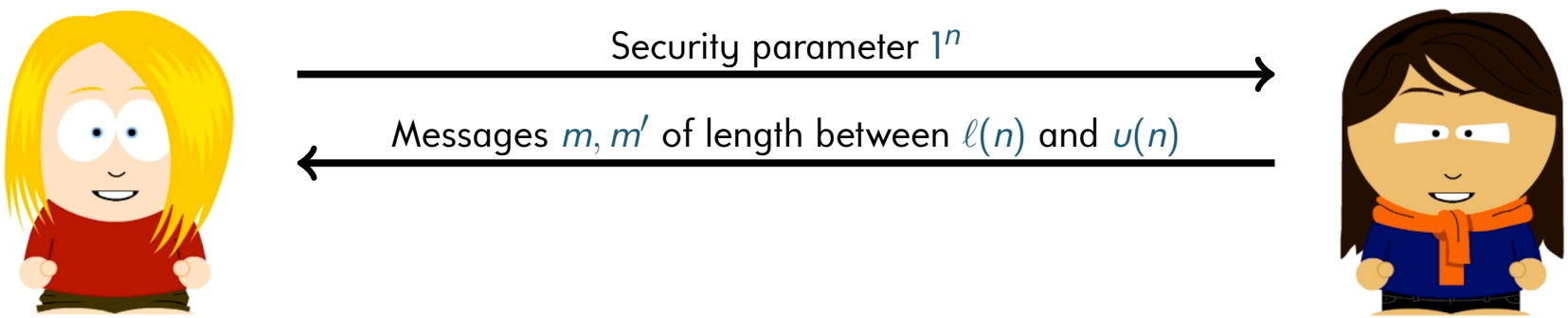
- As usual we write first (security) parameter as subscript (i.e. $h_n(m)$ instead of $h(n, m)$)
- h_n summarizes message m of length between $\ell(n)$ and $u(n)$ in $h_n(m)$ of constant length $p(n)$

Excursion: Hash Functions

§9.4 Definition (Collision-finding game)

Let $n \in \mathbb{N}$, h hash function for hash length p and limits ℓ and u, and $\mathcal{A}$ stateful algorithm. Collision-finding game $\mathrm{Hash}^{\mathrm{coll}}_{h,\mathcal{A}}(n)$ is

1. Security parameter 1^n sent to adversary $\mathcal{A}$
2. $\mathcal{A}$ selects messages $m, m' \in \{0,1\}^*$ with $|m|, |m'| \in [\ell(n), u(n)]$
3. Return $(m \neq m') \wedge (h_n(m) = h_n(m'))$



Return $(m \neq m') \wedge (h_n(m) = h_n(m'))$

Excursion: Hash Functions

§9.5 Definition (Collision-resistant)

Hash function h is collision-resistant if for every PPT algorithm $\mathcal{A}$

$$\mathrm{P}\left[\mathrm{Hash}^{\mathrm{coll}}_{h,\mathcal{A}}(n)\right] \simeq 0$$

Notes

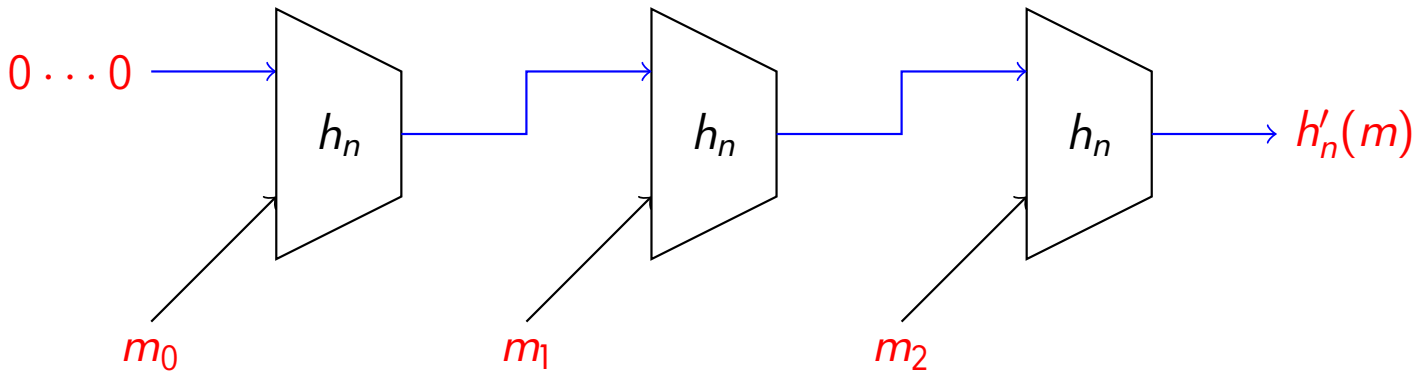
- Adversary should not be able to efficiently & reliably find collision
- Although collisions must exist since $p(n) < u(n)$

§9.6 Definition (Merkle-Damgård transform)

Let h be compression function with hash length p and limit ℓ.
Construct hash function h' with hash length p' and limits ℓ' and u' for every $n \in \mathbb{N}$ by

- $p'(n) = p(n)$ and $\ell'(n) = 0$ and $u'(n) = 2^{\ell(n)-p(n)} - 1$
- Let m be message, $a = \lceil \frac{|m|}{B} \rceil$ and $m_a = \mathrm{bin}(|m|)$ with $B = \ell(n) - p(n)$
  (number a of message blocks for message block size B)
- Let $m = m_0 \cdots m_{a-1}$ with $|m_i| = B$ for all $i < a$
  and m_{a-1} padded with 0 as necessary
- Let $b_0 = 0^{p(n)}$ and $b_{i+1} = h_n(b_i m_i)$ for all $i \leq a$
  (compress current state and block)
- Then $h'_n(m) = b_{a+1}$

Merkle-Damgård Transform



§9.7 Theorem

Hash function h' constructed by Merkle-Damgård transform (Def. §9.6) from collision-resistant compression function h is collision-resistant

Proof

By contraposition: We prove that if there exists PPT algorithm that finds collision for h', then there exists PPT algorithm that finds collision for h. Let $n \in \mathbb{N}$ and suppose that $m, m' \in \{0,1\}^*$ with $m \neq m'$ and $h'_n(m) = h'_n(m')$ is suitable collision of h'. We assume the notation of Definition §9.6 for m and primed for m'. We distinguish two cases:

- Let $|m| \neq |m'|$. By construction $m_a = \mathrm{bin}(|m|) \neq \mathrm{bin}(|m'|) = m'_{a'}$. Then $b_a m_a \neq b'_{a'} m'_{a'}$ but $h_n(b_a m_a) = h'_n(m) = h'_n(m') = h_n(b'_{a'} m'_{a'})$, so $b_a m_a$ and $b'_{a'} m'_{a'}$ form collision for compression function h_n.
- Let $|m| = |m'|$. Select maximal $i \leq a$ such that $b_i m_i \neq b'_i m'_i$. Index exists since $b_{a+1} = b'_{a'+1}$, but $m \neq m'$. By maximality, $h_n(b_i m_i) = b_{i+1} = b'_{i+1} = h_n(b'_i m'_i)$, so $b_i m_i$ and $b'_i m'_i$ form collision for compression function h_n. □

Excursion: Hash Functions

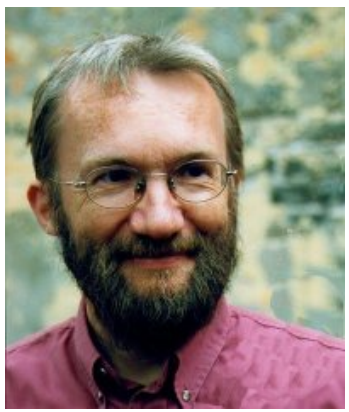
Ralph Merkle (* 1952)

- US computer scientist
- Faculty at Singularity University
- Co-inventor of public-key cryptography

© David Orban

Ivan Damgård (* 1956)

- Danish cryptographer
- Professor at Aarhus University
- Founder Cryptomathic & IACR Fellow since 2010



© Ivan Damgård

Excursion: Hash Functions

Example MD5

- 1992: Introduced by Ronald Rivest
  No security parameter, constant 128 bit hash length
  Utilizes Merkle-Damgård transform with 512 bit block size
- 1996: First collision of compression function [Dobbertin 1996]
- 2004: Broken in 1h on supercomputer [Wang, Feng, Lai, Yu 2004]
- 2013: Broken in 1s on PC (complexity 2^{18}) [Tao, Liu, Feng 2013]

- Was considered "collision-resistant" until 2004
  (No "hash function" with fixed hash length can be collision-resistant)
- Alternatives: ~~SHA-1 (1995)~~, ~~SHA-2 (2001)~~, SHA-3 (2015)
  (SHA-1: Broken at cost roughly 45,000 USD)
  (SHA-2: Susceptible to length extension attacks; limited rounds broken)

Excursion: Hash Functions

Finding collisions

- Fixed hash-length 128 bits (like MD5)
- Hashing $2^{128} + 1$ different messages certainly yields collision
  (by pigeon-hole principle)

How many messages do we need to hash
to find collision with 50% probability?

a. $\approx 2^{128}$
b. $\approx 2^{127} = \frac{2^{128}}{2}$
c. $\approx 2^{64} = \sqrt{2^{128}}$
d. $\approx 128 = \log_2(2^{128})$

§9.8 Theorem (Lower bound for birthday problem)

Given finite set S and $d \leq \sqrt{2|S|}$, probability of $|\{s_1, \ldots, s_d\}| < d$ when $s_1, \ldots, s_d$ are drawn independently and uniformly at random from S is at least

$$\frac{d(d-1)}{4|S|}$$

Proof

Nice exercise requiring some probability theory. □

Excursion: Hash Functions

Notes

- For $|S| = 2^{128}$ and $d = 2^{64} \leq \sqrt{2 \cdot 2^{128}}$ collision probability is at least

$$\frac{d(d-1)}{4|S|} = \frac{2^{64}(2^{64}-1)}{2^{130}} = \frac{2^{64}-1}{2^{66}} = \frac{1}{4} - \frac{1}{2^{66}} \approx 25\%$$

- Guideline: Birthday attacks effectively halve hash length
  (square root of hash space size sufficient to likely find collision)

Excursion: Hash Functions

Applications of collision-resistant hash-functions

- Message authentication (hash-and-MAC or HMAC)
  (standard use case of CBC-MAC; hash guarantees constant size)
- Fingerprinting (e.g. virus or malware identification)
- Deduplication (identification of duplicates)
- Password storage (store hash instead of password)
- Key derivation (generate stronger key from weak key)
- Commitment schemes

Overview

Contents

1. Classical cryptography
   (Shift & Vigenère cipher, one-time pad, perfect secrecy)
2. Security definitions & threat models
   (Computational security, CPA & CCA)
3. Private-key cryptography
   (Message authentication, hash functions, primitives, relevant ciphers)
4. Public-key cryptography
   (Assumptions, key management, digital signatures, relevant ciphers)

Recall

- Modular arithmetic $\mathbb{Z}_\ell = (Z_\ell, +_\ell, \cdot_\ell, 0, 1)$ with $Z_\ell = \{0, \ldots, \ell - 1\}$ and addition & multiplication modulo ℓ
- $\mathbb{Z}_\ell$ is commutative ring (commutative field iff ℓ prime)
  (commutative field potentially without multiplicative inverses)
- Addition, multiplication, exponentiation, & greatest common divisor (gcd) of large numbers efficient

§9.9 Lemma (Extended Euclidean algorithm)

For every $a, b \in \mathbb{N}_+$ there exist $a', b' \in \mathbb{Z}$ such that $aa' + bb' = \gcd(a, b)$. Moreover, $\gcd(a, b)$ is smallest element of

$$\{\ell \in \mathbb{N}_+ \mid a', b' \in \mathbb{Z},\ \ell = aa' + bb'\}$$

Example

- $\gcd(17, 4) = 1$ and $17 \cdot 1 + 4 \cdot (-4) = 1$
- Sample computation of extended Euclidean algorithm:

$$\gcd'(17, 4) = \left(d, y, x - y\lfloor\frac{17}{4}\rfloor\right) \text{ with } \quad (d, x, y) = \gcd'(4, 17 \bmod 4)$$

$$= (1, 0, 1)$$

$$= (1, 1, 0 - 1 \cdot 4) = (1, 1, -4)$$

- $\gcd'(a, b) = (d, x, y)$ iff $\gcd(a, b) = d$ and $ax + by = d$

§9.10 Lemma (Invertability)

Let $\ell, a \in \mathbb{N}_+$ with $\ell > 1$.
There exists $a^{-1} \in \mathbb{N}$ such that $aa^{-1} = 1 \bmod \ell$ iff $\gcd(a, \ell) = 1$

Proof

($\rightarrow$) Since $aa^{-1} = 1 \bmod \ell$ we have $aa^{-1} - 1 = \ell b$ for some $b \in \mathbb{N}$. Thus, $aa^{-1} + \ell(-b) = 1$. Hence $\gcd(a, \ell) = 1$ by Lemma §9.9.

($\leftarrow$) By Lemma §9.9 there exist $a', \ell' \in \mathbb{Z}$ such that $aa' + \ell\ell' = 1$. Hence $aa' = 1 \bmod \ell$ and $aa^{-1} = 1 \bmod \ell$ for $a^{-1} = a' \mod \ell$. □

Example

- 3 has multiplicative inverse in $\mathbb{Z}_{16}$ because $\gcd(3, 16) = 1$
  $(3 \cdot 11 = 33 = 1 \bmod 16$, so inverse is 11)
- 4 has no multiplicative inverse in $\mathbb{Z}_{16}$ because $\gcd(4, 16) = 4 \neq 1$
  $(4 \cdot i \bmod 16)_{i \in \mathbb{N}} = (0, 4, 8, 12, 0, 4, 8, 12, \dots)$

§9.11 Definition (Multiplicative group)

Let $\ell \in \mathbb{N}_+$ with $\ell > 1$. Define $\mathbb{Z}_\ell^* = (Z_\ell^*, \cdot_\ell, 1)$ with

$$Z_\ell^* = \{i \in \mathbb{N} \mid i < \ell,\ \gcd(i, \ell) = 1\}$$

and $\phi(\ell) = |Z_\ell^*|$ (Euler's totient function)

§9.12 Lemma (Multiplicative group)

$\mathbb{Z}_\ell^*$ is commutative group

Proof

Simple exercise using Lemma §9.10 □

Example

- $\mathbb{Z}_6^* = (Z_6^*, \cdot_6, 1)$ has elements $Z_6^* = \{1, 5\}$ and

$$1 \cdot_6 1 = 1 \qquad 1 \cdot_6 5 = 5 \qquad 5 \cdot_6 1 = 5 \qquad 5 \cdot_6 5 = 1$$

- Isomorphic to $(\{0,1\}, \oplus, 0)$ ($\oplus$ is XOR)
  via $h(1) = 0$ and $h(5) = 1$

§9.13 Theorem (Euler's theorem)

Let $\mathbb{G} = (G, \cdot, 1)$ be finite commutative group and $\ell = |G|$.
Then $g^\ell = 1$ for every $g \in G$.

Proof

Let $G = \{g_1, \ldots, g_\ell\}$ and observe

$$\prod_{i=1}^{\ell} g_i = \prod_{i=1}^{\ell} (g \cdot g_i) = g^\ell \cdot \prod_{i=1}^{\ell} g_i$$

because middle product contains ℓ pairwise different factors ($g \cdot g_i = g \cdot g_{i'}$ implies $g_i = g_{i'}$). Canceling $\prod_{i=1}^{\ell} g_i$ left and right, we obtain $1 = g^\ell$. □

Summary

- Relevant block cipher (AES)
  (implementations of “pseudorandom permutations”)
- Collision-resistant hash functions
- Foundations of public-key cryptography