Rappels
CRYPTA

Ludovic Perret

Sorbonne Universités, UPMC Univ Paris 06, INRIA Paris
LIP6, `PolSyS` Project, Paris, France

2017 – 2018

Plan du cours

Plan du cours

RC4 [R. Rivest, 1987]

- Génération d'une suite chiffrante à partir d'un tableau S de 256 **octets** et d'une clef K.

Phase d'initialisation (RC4-KSA)

- **Pour** $i, 0 \leq i \leq 255$ **faire** $S[i] := i$ **FinPour**
- $j := 0$
- **Pour** $i, 0 \leq i \leq 255$ **faire**

  $j := (j + S[i] + K[i \mod \mathrm{len}(\mathrm{K})]) \mod 2^8$

  Échanger$(S[i], S[j])$
- **FinPour**

RC4 [R. Rivest, 1987]

Génération de la suite chiffrante (RC4-PRGA)

- $i := 0 \quad j := 0$
- **Pour** $k, 0 \leq k \leq \mathrm{no} - 1$ **faire**

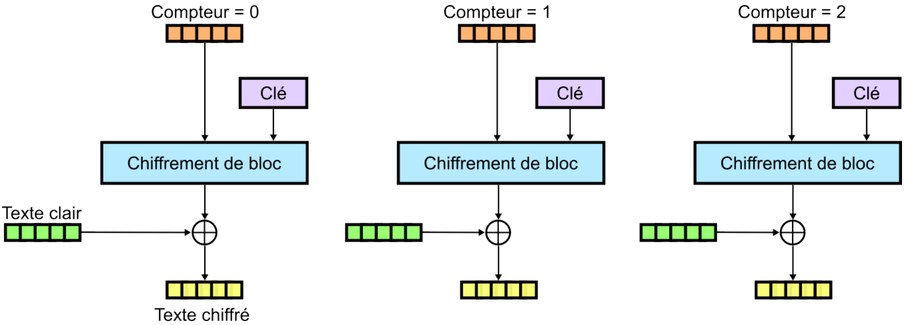
  $i := (i + 1)$ mod $2^8 \quad j := (j + S[i])$ mod 2^8

  Échanger$\big(S[i], S[j]\big)$

  $t := (S[i] + S[j])$ mod 2^8

  $\mathrm{SuiteChiffrante}[k] := S[t]$
- **FinPour**
- **Return** $\mathrm{SuiteChiffrante}$

Mode CTR



Source : `https://fr.wikipedia.org/`

Plan du cours

Plan du cours

Paradoxe des anniversaires

Proposition

Soient $H : X \to Y, x_1, \ldots, x_k$ des éléments distincts de X tirés aléatoirement, et $y_i = H(x_i)$, pour tout $i, 1 \leq i \leq k$.

$$\Pr(\exists \text{ collision}) \approx 1 - e^{-\frac{k(k-1)}{2N}}, \text{ avec } N = |Y|.$$

Paradoxe des anniversaires

Démonstration.

- On suppose que les y_i sont des éléments aléatoires de Y.

Paradoxe des anniversaires

Démonstration.

- On suppose que les y_i sont des éléments aléatoires de Y.
- Nous avons $N = |Y|$. La probabilité que $y_{i+1} \notin \{y_1, \ldots, y_i\}$ est $p_{i+1} = (1 - i/N)$.

Paradoxe des anniversaires

Démonstration.

- On suppose que les y_i sont des éléments aléatoires de Y.
- Nous avons $N = |Y|$. La probabilité que $y_{i+1} \notin \{y_1, \ldots, y_i\}$ est $p_{i+1} = (1 - i/N)$.
- La probabilité que les $y_1, \ldots, y_k$ – tirés dans cet ordre – soient distincts est
$$P = \prod_{i=0}^{k-1} p_{i+1} = \prod_{i=0}^{k-1} (1 - i/N).$$

Paradoxe des anniversaires

Démonstration.

- On suppose que les y_i sont des éléments aléatoires de Y.
- Nous avons $N = |Y|$. La probabilité que $y_{i+1} \notin \{y_1, \ldots, y_i\}$ est $p_{i+1} = (1 - i/N)$.
- La probabilité que les $y_1, \ldots, y_k$ – tirés dans cet ordre – soient distincts est
$$P = \prod_{i=0}^{k-1} p_{i+1} = \prod_{i=0}^{k-1} (1 - i/N).$$
- La probabilité de non-collision est donc P.

Paradoxe des anniversaires

Démonstration.

- On suppose que les y_i sont des éléments aléatoires de Y.
- Nous avons $N = |Y|$. La probabilité que $y_{i+1} \notin \{y_1, \ldots, y_i\}$ est $p_{i+1} = (1 - i/N)$.
- La probabilité que les $y_1, \ldots, y_k$ – tirés dans cet ordre – soient distincts est

$$P = \prod_{i=0}^{k-1} p_{i+1} = \prod_{i=0}^{k-1} (1 - i/N).$$

- La probabilité de non-collision est donc P.
- En approchant $1 - x$ par e^{-x} pour x proche de 0, on obtient : $P \simeq \prod_{i=0}^{k-1} e^{-\frac{i}{N}} = e^{-\frac{k(k-1)}{2N}}$.

$\square$

Collision

Proposition

Soit $H : X \rightarrow Y$ une fonction de hachage, avec $|X| \geq |Y|$ et $|Y| = N$.
Pour trouver une collision avec probabilité $\geq 1/2$, il “suffit" de hacher :

$$\mathcal{O}(\sqrt{N}) \text{ éléments de } X.$$

Autrement dit . . .

Pour avoir une probabilité $\geq 1/2$ de trouver une collision, il suffit de hacher un peu plus de $\sqrt{N}$ éléments de X.

Preuve

Démonstration.

Notons $\epsilon = 1 - P$, la probabilité d'avoir au moins une collision.
Exprimons k en fonction de ϵ et N :

$$\epsilon \simeq 1 - e^{-\frac{k(k-1)}{2N}} \Rightarrow -\frac{k(k-1)}{2N} \simeq \ln(1-\epsilon).$$

Ainsi, $k^2 - k \simeq 2N\ln(\frac{1}{1-\epsilon})$. En ignorant le terme $-k$, on obtient :

$$k \simeq \sqrt{2N\ln\left(\frac{1}{1-\epsilon}\right)}.$$

Pour $\epsilon = 1/2$, on trouve $k \simeq 1.18 \cdot \sqrt{N}$. $\square$

Illustration

- Supposons que X est un ensemble d'individus
- Y l'ensemble des 365 jours d'une année non bissextile
- $H(x)$, le jour de l'anniversaire d'une personne de X (on suppose que X comporte plus de 365 personnes)
- On obtient $k \simeq 1.18 \cdot \sqrt{365} \simeq 1.18 \cdot 19.10 \simeq 22.5$

Plan du cours

Fonction de compression

Problème

Comment gérer une donnée de taille variable ?

Définition

fonction de compression : fonction qui transforme toute chaîne d'une taille fixée $r + n$ en une chaîne de taille n.

$$f : \{0,1\}^{r+n} \quad \mapsto \quad \{0,1\}^{n}.$$

Construction de Merle-Damgård – (I)

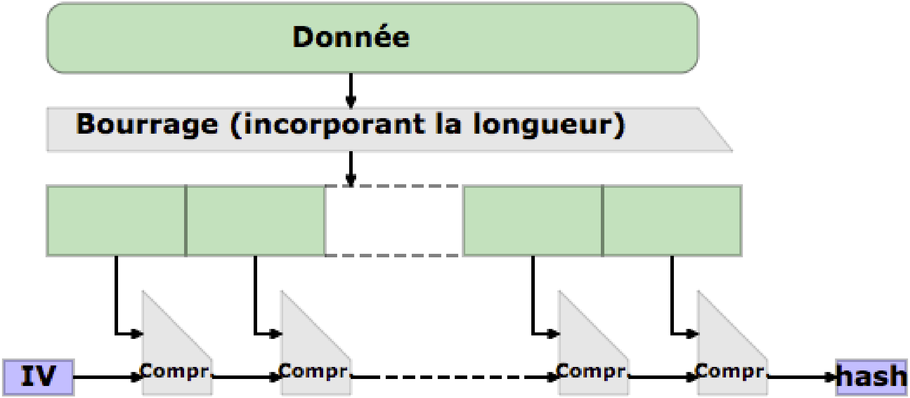
La chaîne x (de longueur arbitraire) à hacher subit un prétraitement (padding) qui la transforme en t blocs de r bits $x_1, \ldots, x_t$.

- $\mathrm{IV} \in \{0,1\}^n$ une valeur initiale (ou vecteur d'initialisation),
- $f : \{0,1\}^r \times \{0,1\}^n \mapsto \{0,1\}^n$ une fonction de compression.

On calcule :

$$H_0 = \mathrm{IV},\ H_i = f(H_{i-1}, x_i),\ 1 \leq i \leq t.$$

Merkle-Damgård



Source : `https://fr.wikipedia.org/`

Remarque

Fonction de hachage $\Longleftrightarrow$ Fonction de compression

Plan du cours

Davies-Meyer – (I)

Soit $\mathrm{E} : \mathbb{F}_2^k \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ un chiffrement par bloc.

- On découpe la donnée x à hacher en t blocs $x_1, \ldots, x_t$ de taille n.

$$H_0 = \mathrm{IV},\; H_i = E(m_i, H_{i-1}) \oplus H_{i-1},\; 1 \leq i \leq t.$$

Le haché est H_t.

Davies-Meyer – (I)

Soit $\mathrm{E} : \mathbb{F}_2^k \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ un chiffrement par bloc.

- On découpe la donnée x à hacher en t blocs $x_1, \ldots, x_t$ de taille n.

$$H_0 = \mathrm{IV}, \ H_i = E(m_i, H_{i-1}) \oplus H_{i-1}, \ 1 \leq i \leq t.$$

Le haché est H_t.

Point fixe

- $H = E(m, H)$.

Matyas-Meyer-Oseas

Soient $\mathrm{E} : \mathbb{F}_2^k \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ un chiffrement par bloc et $g : \mathbb{F}_2^n \to \mathbb{F}_2^k$.

- On découpe la donnée x à hacher en t blocs $x_1, \ldots, x_t$ de taille n.

$$H_0 = \mathrm{IV},\ H_i = E\big(g(H_{i-1}), m_i\big) \oplus m_i,\ 1 \leq i \leq t.$$

La haché est H_t.

Miyaguchi-Preneel

Soient $\mathrm{E} : \mathbb{F}_2^k \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ un chiffrement par bloc et une fonction $g : \mathbb{F}_2^n \to \mathbb{F}_2^r$.

- On découpe la donnée x à hacher en t blocs $x_1, \ldots, x_t$ de taille n.

$$H_0 = \mathrm{IV},\ H_i = E\big(g(H_{i-1}), x_i\big) \oplus x_i \oplus H_{i-1},\ 1 \leq i \leq t.$$

La valeur hachée est H_t.

Plan du cours

SHA2 (SHA256, SHA384 et SHA512)

SHA = Secure Hash Algorithm

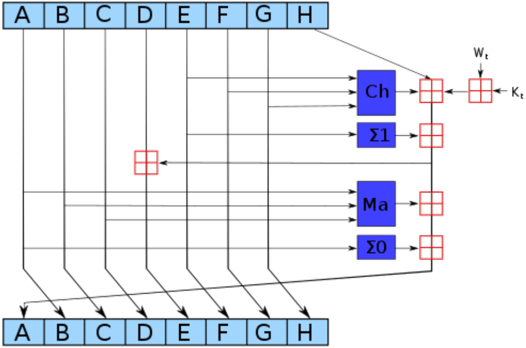
- Merkle-Damgård
- Fonction de compression ; taille des blocs $\in \{512, 1024\}$ bit
- Emprunte $\in \{224, 256, 384, 512\}$ bit

Fonction de Compression – SHA256

- Variables de chaînage de 256 bits (A, B, C, D, E, F, G, H)
- 64 étapes élémentaires (tours)
- Expansion du bloc de message

  16 mots (32 bits) vers 64 mots

SHA2 – Tour



Source : `https://fr.wikipedia.org/`

$$Ch(x, y, z) = (x \wedge y) \oplus (\neg x \wedge z)$$
$$Ma(x, y, z) = (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z)$$
$$\Sigma_0(x) = \mathrm{ROT}^2(x) \oplus \mathrm{ROT}^{13}(x) \oplus \mathrm{ROT}^{22}(x)$$
$$\Sigma_1(x) = \mathrm{ROT}^6(x) \oplus \mathrm{ROT}^{11}(x) \oplus \mathrm{ROT}^{25}(x)$$

SHA12 – (II)

Expansion de message : $W_i = m_i, \forall i, 0 \leq i \leq 15$, et

$$W_t = \sigma_0(W_{t-2}) + W_{t-7} + \sigma_0(W_{t-15}) + W_{t-16},$$

pour $t, 16 \leq i \leq 63$.

$$\sigma_0(x) = \mathrm{ROT}^7(x) \oplus \mathrm{ROT}^{18}(x) \oplus \mathrm{SHR}^3(x)$$
$$\sigma_1(x) = \mathrm{ROT}^{17}(x) \oplus \mathrm{ROT}^{19}(x) \oplus \mathrm{SHR}^{10}(x)$$

Plan du cours

Compétition SHA3

Xiaoyun Wang, Hongbo Yu.
How to Break MD5 and Other Hash Functions.
EUROCRYPT 2005.

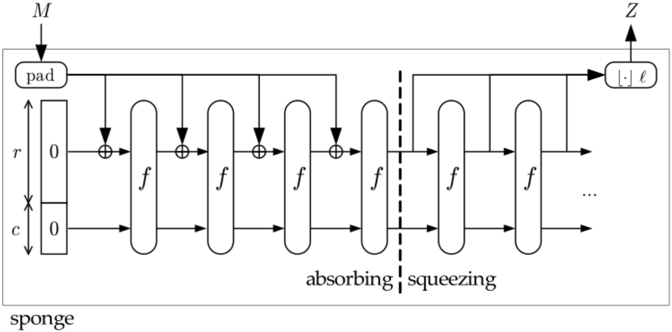
Nouveau standard

- 64 soumissions (2008)
- 14 candidats en phase 2
- 5 candidats en phase 3 (2010)

SHA3, 2012

- Keccak (G. Bertoni, J. Daemen, M. Peeters, G. Van Assche)

Construction sponge



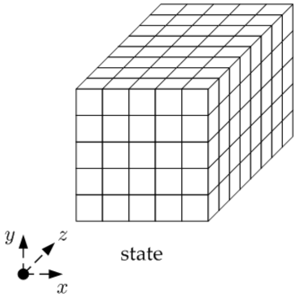
Source : `https://keccak.team/sponge_duplex.html`

- r, le taux (taille d'un bloc)
- c, la capacité
- f, permutation sur $b = r + c$ bit

Niveau de sécurité

$$2^{c/2}$$

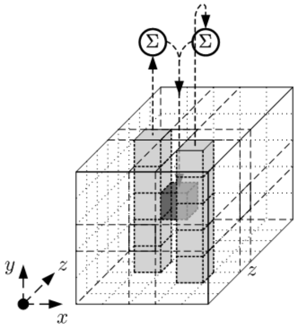
Keccak-f – Structure de donnée



Source : `https://keccak.team`

- 2^ℓ tableaux de 5×5 bit, avec $\ell \in \{1, 2, 5, 8, 16, 32, 64\}$
- $b = 25 \times 2^\ell$

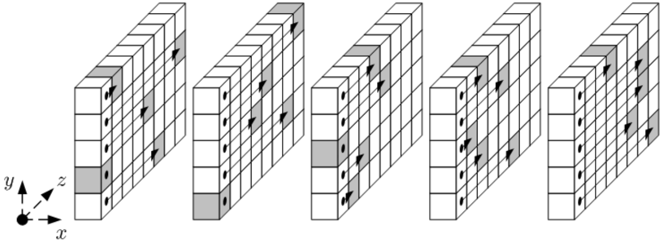
Keccak-f – Fonction θ



Source : `https://keccak.team`

$$a[i][j][k] := a[i][j][k] \oplus \sum_{j'=0}^{4} a[i-1][j'][k] \oplus \sum_{j'=0}^{4} a[i+1][j'][k-1]$$

Keccak-f – Fonction ρ



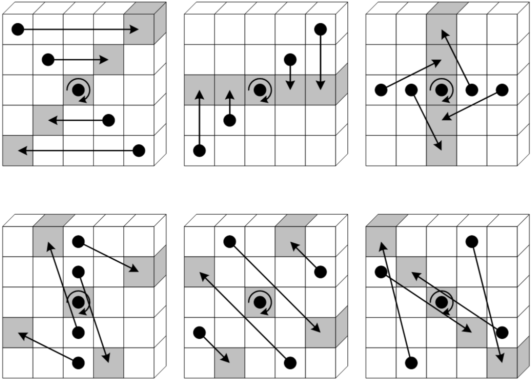
Source : `https://keccak.team`

$$a[i][j][k] := a[i][j][k - (t+1)(t+2)/2],$$

avec $t = -1$ si $i = j = 0$; sinon $t, 0 \leq t \leq 24$ et :

$$\begin{pmatrix} i \\ j \end{pmatrix} \equiv \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix}^t \begin{pmatrix} 0 \\ 1 \end{pmatrix} \bmod 5.$$

Keccak-f – Fonction π

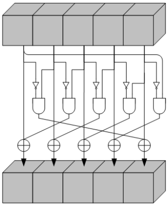


Source : `https://keccak.team`

$$a[i][j] := a[i'][j'], \text{avec}$$

$$\begin{pmatrix} i \\ j \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} i' \\ j' \end{pmatrix}$$

Keccak-f – Fonction χ



Source : `https://keccak.team`

$$a[i] := a[i] + (a[i+1] + 1)a[i+2].$$

Keccak-f

On répète $n_r = 12 + 2\ell$ fois :

$$R = \iota \circ \chi \circ \pi \circ \rho \circ \theta.$$

avec ι :

$$a := a + RC[i_r].$$

Keccak-f

On répète $n_r = 12 + 2\ell$ fois :

$$R = \iota \circ \chi \circ \pi \circ \rho \circ \theta.$$

avec ι :

$$a := a + RC[i_r].$$

SHA3

	sortie	r	c	Collision
SHA3-224	224	1152	448	112
SHA3-256	256	1088	512	128
SHA3-384	384	832	768	192