

Cryptographie par la pratique avec Python et OpenSSL

Cours et applications concrètes

Rémi Boule

Enseignant à l'IUT de Blagnac

Préface de Stéphane Bortzmeyer

DUNOD

J'adresse mes sincères remerciements à Stéphane Bortzmeyer qui m'a fait l'honneur de préfacer cet ouvrage.

Je remercie aussi vivement Gilles Peskine. Sa relecture rigoureuse et constructive a contribué significativement à la qualité de ce livre.

Gilles Peskine est développeur de systèmes informatiques sécurisés. Normalien et docteur en informatique, il cherche à répondre à la question : comment peut-on être sûr qu'un ordinateur fait bien ce qu'il est censé faire ? Il est mainteneur de la bibliothèque de cryptographie Mbed TLS, utilisée principalement dans les systèmes embarqués, et dont il a notamment redéfini l'interface d'appel des primitives cryptographiques.

Direction et conception graphique de la couverture
Nicolas Wiel – Pierre André Gualino (graphiste)

Sommaire

Préface	1
1 Enjeux et définitions	3
1.1 Pourquoi la cryptographie ?	3
1.2 Historique	4
1.2.1 PGP	4
1.2.2 Crypto Wars	5
1.2.3 Affaire Snowden	6
PRISM	6
XkeyScore	6
Bullrun	7
Conséquences	7
1.3 Vocabulaire de la cryptographie	8
2 Outils informatiques pour la cryptographie	9
2.1 Python pour la cryptographie	9
2.1.1 Listes	9
Accéder et modifier des éléments	10
<i>Slicing</i> de listes	11
2.1.2 Dictionnaires	11
Création d'un dictionnaire	11
2.1.3 Tuple	12
2.1.4 Itérateurs et générateurs	12
2.1.5 Travailler en binaire	13
Opérations bit à bit	14
Rotation de bits	14
<i>Endianness</i>	15
2.1.6 Unicode	16
2.1.7 Congruences	16
2.1.8 Fonctions anonymes	17
2.1.9 <code>filter</code> , <code>map</code> et <code>reduce</code>	17
2.2 Représentations graphiques	18

2.2.1	matplotlib en 2D	18
2.2.2	matplotlib en 3D avec meshgrid	19
2.3	Autres bibliothèques	20
2.3.1	pack() et unpack()	20
2.3.2	requests	21
2.3.3	JSON	21
2.3.4	cryptography	22
2.4	SSL/TLS	23
2.4.1	OpenSSL	23
3	Cryptographie classique	25
3.1	Congruences dans $\mathbb{Z}$	25
3.1.1	Congruences	25
3.1.2	Opérations dans $\mathbb{Z}/n\mathbb{Z}$	26
3.2	César	27
3.3	Vigenère	28
3.4	Quel bilan ?	29
3.5	Exercices	31
3.5.1	Chiffrement de César	31
3.5.2	Attaques sur le chiffrement de César	32
3.5.3	Chiffrement de Vigenère	33
3.5.4	Attaque sur le chiffrement de Vigenère	34
4	Stéganographie	37
4.1	Contexte et historique	37
4.2	En ligne de commande	38
4.3	Stéganographie sur les fichiers	38
4.3.1	<i>Whitespace Steganography</i>	38
4.3.2	Stéganographie dans le binaire	39
4.4	Stéganographie sur les images	39
4.4.1	Stéganographie sur LSB	40
4.5	Exercices	42
4.5.1	Stéganographie sur les fichiers	42
4.5.2	Cacher une image dans une autre	43
4.5.3	Cacher du texte dans une image	45
5	Hachage	47
5.1	Fonction de hachage cryptographique	47
5.2	MD5	49
5.2.1	<i>Padding</i> et extension	49
5.2.2	Vue générale	50
5.2.3	Fonction de compression	52
	Fonctions auxiliaires	53
	Table des constantes	53
	Décalages	54

5.2.4	Zoom sur une opération	54
5.2.5	Rondes suivantes	54
5.3	SHA-1 et SHA-2	56
5.4	SHA-3	57
5.4.1	Construction en éponge	57
5.5	Collisions	58
5.5.1	Paradoxe des anniversaires	60
5.6	Salage et stockage de mot de passes	62
5.7	HMAC et KMAC	63
5.8	Blockchain	63
5.8.1	Qu'est ce qu'un bloc Bitcoin ?	64
	En-tête de bloc Bitcoin	66
	Arbre de Merkle	68
5.8.2	Minage	69
	Preuve de travail	70
5.9	Exercices	72
5.9.1	Inversion de MD5	72
5.9.2	Implémenter MD5	72
5.9.3	Dérivation de clé avec PBKDF2	77
5.9.4	Nombre maximum de bitcoins	79
5.9.5	Blockchain	79
5.9.6	Arbre de Merkle	81
5.9.7	Minage de bitcoins	83
6	Chiffrement symétrique	87
6.1	Chiffrement par flot	87
6.1.1	ChaCha20	88
6.1.2	Poly1305	88
6.1.3	ChaCha20-Poly1305	89
6.2	Modes de chiffrement	89
6.2.1	<i>Padding</i>	89
6.2.2	Mode ECB	90
6.2.3	Mode CBC	91
6.2.4	Mode CTR	92
6.2.5	Mode GCM	93
	Chiffrement	94
	Données authentifiées additionnelles	94
	Tag d'authenticité	94
	Déchiffrement et authentification	95
	Performances	96
6.3	Chiffrement de bloc : AES	96
6.3.1	Pré-requis mathématiques pour AES	97
6.3.2	Opérations usuelles dans $GF(2^8)$	98
6.3.3	S-box AES	100
6.3.4	Chiffrement	101

6.3.5	Déchiffrement	102
6.3.6	En Python et OpenSSL	102
6.4	AES en pratique : LoRaWAN	103
6.4.1	LoRaWAN	104
6.4.2	Clés de session	104
6.4.3	Analyse d'un paquet de données	104
6.4.4	Accès à la donnée	106
6.4.5	Chiffrement en LoRaWAN	107
	Un simple XOR ?	107
	Suite S	108
6.5	Cryptographie symétrique légère	109
6.6	Exercices	110
6.6.1	Division de polynômes	110
6.6.2	Mathématiques pour AES	110
6.6.3	Analyse d'un paquet LoRaWAN en Python	112
6.6.4	Cryptographie en LoRaWAN	114
6.6.5	OTP	117
6.6.6	AES en mode GCM avec données associées	118
7	Cryptographie asymétrique	119
7.1	Limitations de la cryptographie symétrique	119
7.2	Diffie-Hellman	120
7.2.1	Problème de la distribution des clés	120
7.2.2	Naissance d'Alice et Bob	121
7.2.3	Fonction à sens unique	121
7.2.4	Logarithme discret	122
7.2.5	Échange de Diffie-Hellman (DH)	123
7.2.6	Clés asymétriques	124
7.3	Signature numérique	125
8	RSA	127
8.1	Rappels d'arithmétique	127
8.1.1	PGCD	127
8.1.2	Algorithme d'Euclide	128
8.1.3	Égalité de Bézout	129
8.1.4	Algorithme de Blankinship	130
8.1.5	Inversion modulo p	132
8.2	Nombres premiers	132
8.2.1	Répartition des nombres premiers	132
8.2.2	Tests de primalité	133
	Test de Fermat	134
	Test de Miller-Rabin	135
	Avec <code>sympy</code>	135
	Fonction Python <code>isprime()</code>	136
8.3	RSA	138

8.3.1	Création des clés	139
8.3.2	Choix de l'exposant de chiffrement RSA	139
8.3.3	Choix de p et q	140
8.3.4	Chiffrement et déchiffrement	141
8.3.5	Exponentiation modulaire rapide	141
8.4	Preuve de RSA	142
8.5	PKCS #1	143
8.5.1	Primitives du standard	143
	I2OSP et OS2IP	143
	Schéma de chiffrement RSAES-OAEP	144
8.6	Signature RSA	145
8.7	RSA en Python et OpenSSL	145
8.8	Exercices	147
8.8.1	Arithmétique pour RSA	147
8.8.2	Arithmétique avec Python	151
8.8.3	RSA avec Python	151
9	Cryptographie sur courbes elliptiques	153
9.1	Contexte	153
9.2	Courbe elliptique sur $\mathbb{R}$	154
9.2.1	Addition	155
9.2.2	Multiplication scalaire	158
9.3	Courbe elliptique modulo p	158
9.3.1	Représentation graphique	159
9.3.2	Opérations	159
9.4	Cryptographie sur courbes elliptiques	160
9.4.1	ECDLP	160
9.4.2	Choix des paramètres de la courbe	160
9.4.3	Exemples de courbes elliptiques	161
9.4.4	Échange de clés : ECDH	163
9.4.5	Signature : ECDSA	164
9.5	Courbes elliptiques en pratique	165
9.5.1	Avec OpenSSL	165
9.5.2	Bitcoin	165
	Adresses Bitcoin	166
	Courbe secp256k1	167
9.6	Exercices	169
9.6.1	Opérations sur courbes elliptiques	169
9.6.2	Courbe de l'ANSSI	171
9.6.3	Chiffrer, déchiffrer	172
9.6.4	Secp256k1	173
10	Nombres aléatoires	175
10.1	Introduction	175
10.1.1	PRNG et TRNG	176

10.2	PRNG congruentiels linéaires	177
10.2.1	Définition	177
10.2.2	Choix des paramètres	178
10.3	Mersenne Twister	179
10.3.1	Description	180
	<i>Twisting</i>	180
	<i>Tempering</i>	182
10.3.2	MT19937	182
10.3.3	Qualité et sécurité de MT	184
10.4	Qualité	184
10.4.1	Test du khi-deux	185
10.4.2	Test spectral	187
10.4.3	Test du <i>birthday spacing</i>	189
10.5	Générateurs cryptographiques	190
10.6	Exercices	191
10.6.1	GCL	191
10.6.2	Test spectral	193
10.6.3	Mersenne Twister	194
11	HTTPS et SSH	197
11.1	HTTPS	197
11.1.1	Contexte et historique	198
	EFF	198
	Google et Wikipédia pour HTTPS	198
	<i>Let's encrypt</i>	199
11.1.2	TLS 1.3	199
	<i>Client Hello</i>	200
	<i>Server Hello</i>	201
	Calcul de la clé du <i>Handshake</i>	202
	<i>Encrypted Extensions</i>	202
	Certificat	202
	Fin du <i>Handshake</i>	203
11.2	SSH	204
11.2.1	Origines	204
11.2.2	Protocole SSH	205
11.2.3	Clés SSH	206
11.2.4	Connexion SSH aux rayons X	206
	Bibliographie	210
	Index	214

Préface

Par Stéphane Bortzmeyer

En 1995, alors que la cryptographie en France était quasiment interdite aux civils, Stéphane Bortzmeyer a publié une tribune dans le journal *Le Monde* appelant à sa libéralisation. C'est un des premiers textes publics sur la question ¹. Ingénieur en réseaux informatiques, il est auteur de plusieurs RFC (documents qui codifient divers aspects de l'Internet) et a publié *Cyberstructure, l'Internet, un espace politique* ², prix du livre Cyber 2019 catégorie cybersécurité dans le cadre du Forum international de la cybersécurité. Il tient un blog sur l'Internet : <https://www.bortzmeyer.org/>.

QUEL est le point commun entre une féministe iranienne qui communique avec des camarades en Allemagne, un lycéen brésilien qui regarde TikTok au lieu de réviser, une banquière suisse qui transfère des fonds et un courtier ivoirien qui regarde les cours du cacao ? Sans en avoir conscience, tous et toutes, dans leur utilisation de l'Internet, utilisent la cryptographie pour se protéger.

Aujourd'hui, alors que beaucoup d'activités humaines ont migré, de gré ou de force (pensons aux services publics qui ne sont souvent accessibles qu'en ligne !), vers l'Internet, les enjeux de la cryptographie sont encore plus importants. Contrairement à ce qu'essaie de faire croire une certaine propagande, le but de la cryptographie n'est pas d'obscurcir ce qui était en clair avant mais au contraire de retrouver la relative vie privée qui existait avant la numérisation. Alors que vous pouviez autrefois prendre le train, acheter un livre, discuter avec des amis, sans donner votre nom, et sans laisser de traces, la numérisation a rendu tout enregistrable et espionnable. La cryptographie est donc le principal outil technique pour limiter ces possibilités d'enregistrement et d'espionnage. À ce titre, elle est une composante indispensable de l'Internet.

Il ne suffit pas qu'un vendeur affirme « ne vous inquiétez pas, c'est crypté » pour être protégé. Les détails comptent. L'utilisation fiable de la cryptographie n'est pas toujours simple. C'est pour cela que le livre de Rémi Boulle est extrêmement utile. Il est nécessaire que des compétences en cryptographie soient plus largement répandues, pour que davantage de personnes puissent participer aux débats politiques impliquant de la cryptographie. Ce livre est

1. <https://www.bortzmeyer.org/crypto.html>

2. <https://cfeditions.com/cyberstructure/>

clair, pédagogique et explique progressivement les concepts, avec des exemples concrets.

Le sujet est riche, et connaître les algorithmes de cryptographie n'est qu'une étape. Il faudra aussi apprendre les limites de la cryptographie qui, comme toute technique, résout des problèmes mais ne fait pas de miracles. Ainsi, la cryptographie ne protège pas contre l'entité avec qui vous communiquez. En termes religieux, « la cryptographie permet de s'assurer qu'on communique bien avec le diable, et que les anges ne peuvent pas écouter, mais elle ne peut pas garantir que Satan est digne de confiance ». Ce rappel est d'autant plus important à une époque où la communication est malheureusement médiée par un petit nombre de grosses entreprises. Faire du HTTPS avec Gmail ne va pas empêcher Google de lire les messages.

La cryptographie, je l'ai dit, doit être utilisée correctement. Ainsi, comme dans l'exemple ci-dessus, il est préférable qu'elle fonctionne de bout en bout, c'est-à-dire que ce qui est chiffré sur votre machine ne soit déchiffré que sur la machine de votre correspondant ou correspondante et pas par un intermédiaire. Le chiffrement de bout en bout n'est pas un détail technique, c'est un facteur très important dans la sécurité.

J'en profite d'ailleurs pour donner un conseil de lecture pour un livre (en anglais) qui parle de cryptographie sous un tout autre angle : « *Concealing for freedom* », de Ksenia Ermoshina et Francesca Musiani, qui traite des aspects sociaux et humains de la sécurité informatique et notamment de la cryptographie (avec des questions du genre « pourquoi les féministes iraniennes utilisent WhatsApp et pas Signal », parce que, comme le note Rémi Boule, il y a des pays où « le simple fait de vouloir chiffrer une image ou un document fait de vous un suspect »).

Revenons justement au livre de Rémi Boule. Je recommande aux étudiant·es de le lire, de faire les exercices, de comprendre ce qui se passe. Vous aurez ainsi acquis une meilleure compréhension de ces algorithmes (même si, évidemment, il est déconseillé de les programmer soi-même pour une utilisation réelle : il y a beaucoup de pièges, qu'il n'est pas possible de couvrir dans un livre d'introduction). Mais vous aurez aussi acquis une meilleure idée de ce que la cryptographie permet, et de ce qu'elle ne permet pas.

À l'heure où j'écris, la Commission Européenne essaie toujours de faire passer le règlement CSAM (*Child Sexual Abuse Material*, dit « Chat Control »), qui forcerait les systèmes de messagerie (comme Signal, WhatsApp, Proton et autres) à examiner les messages envoyés. Si la messagerie en question chiffre, cela obligerait à casser le chiffrement pour cet examen. Certains partisans de cette directive prétendent que non, en jouant sur les mots. Mieux connaître la cryptographie aide à déjouer des arguments trompeurs !

Stéphane Bortzmeyer

1

Enjeux et définitions

Le mot « cryptographie » vient du grec ancien kruptos qui signifie « caché » et de graphein qui signifie « écrire ». L'objectif premier est de permettre la communication entre deux entités sans qu'une tierce partie, qui peut voir passer tous les messages et les lire, ne puisse les comprendre.

1.1 Pourquoi la cryptographie ?

La cryptographie a pour buts principaux d'assurer les points suivants :

- Confidentialité : qui peut lire le contenu des messages ? Il faut s'assurer que les informations ne soient lisibles que par les personnes autorisées. Cela est souvent réalisé grâce au chiffrement, qui transforme les données pour les rendre incompréhensibles sauf à disposer d'une information secrète telle qu'une clé de déchiffrement.
- Intégrité : est-ce que les informations ont été modifiées intentionnellement (le plus souvent à des fins frauduleuses) ou accidentellement ? Il doit être rendu très difficile de modifier le contenu d'un message sans que cela ne soit détecté. Les fonctions de hachage en sont le composant principal.
- Authenticité : est-ce bien la bonne personne qui m'envoie ces données ? On ne doit pas pouvoir émettre des messages en se faisant passer pour quelqu'un d'autre. Ceci peut être réalisé par des signatures numériques (avec, par exemple, des certificats délivrés par des autorités de confiance), des *tokens*, ou des codes d'authentification de message (MAC).
- Non-répudiation : est-ce que l'auteur d'un message ou d'une transaction peut ensuite nier en être à l'origine ? À la manière de la signature d'un chèque, on ne peut pas répudier sa signature. Les signatures numériques sont un outil clé.