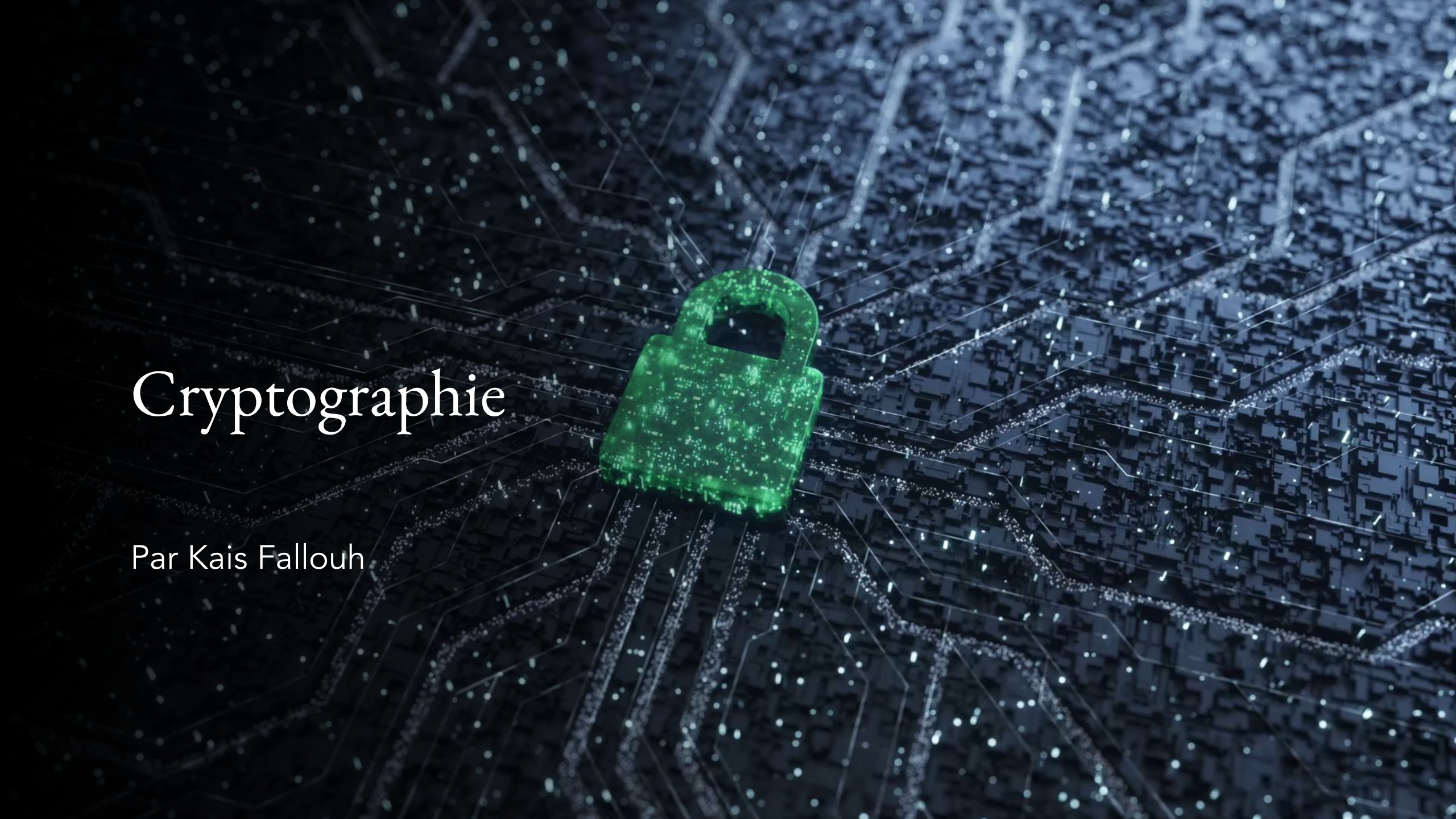
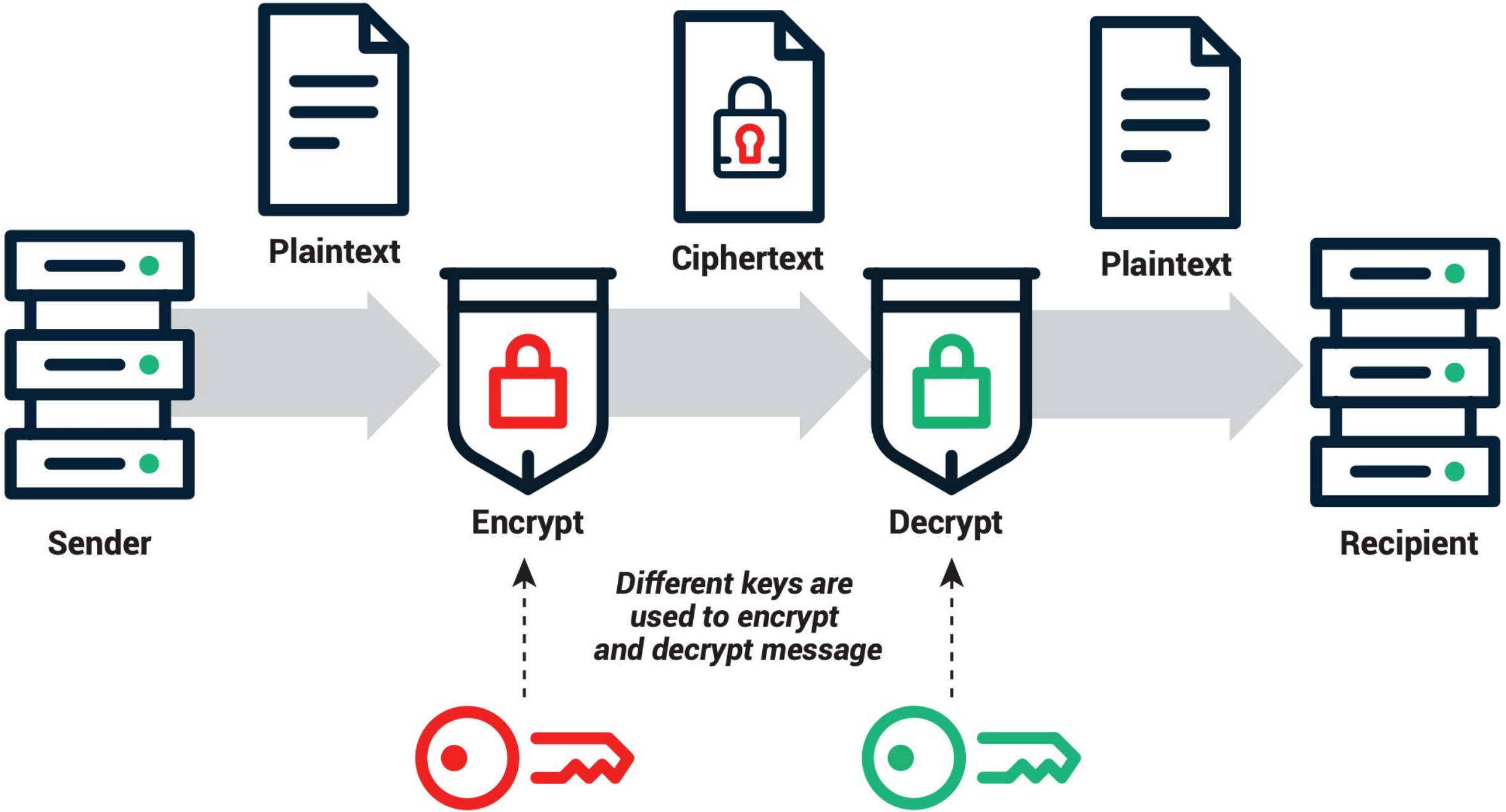


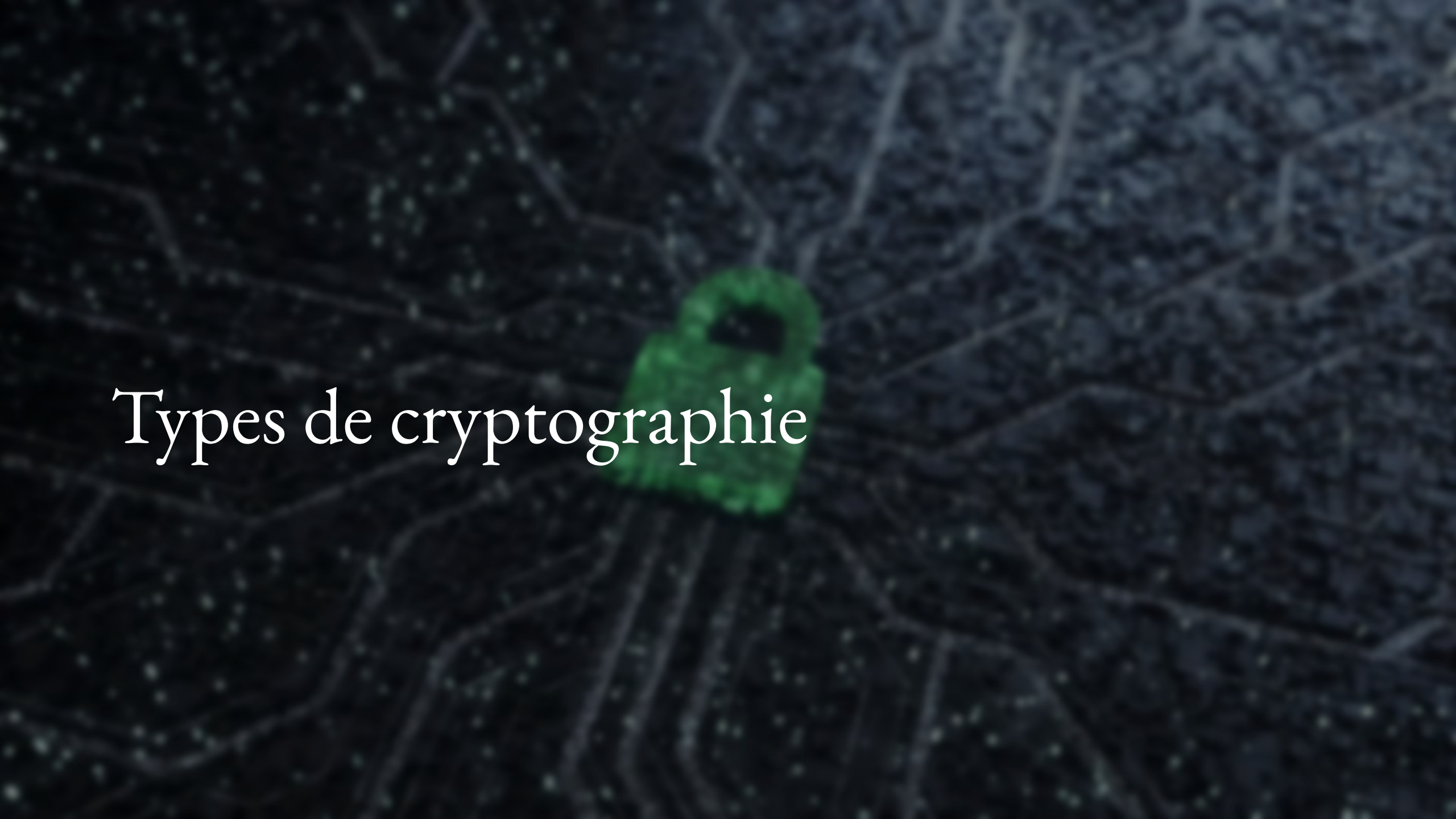
Cryptographie

Par Kais Fallouh

Cryptographie, c’est quoi ?

La cryptographie est le processus de dissimulation ou de codage d’informations afin que seule la personne à qui un message était destiné puisse le lire


Plaintext
Ciphertext
Plaintext
Sender
Encrypt
Decrypt
Recipient
Different keys are used to encrypt and decrypt message



Types de cryptographie

Chiffrement symétrique

- est un type de chiffrement dans lequel la même clé est utilisée pour chiffrer et déchiffrer les données. Cette méthode est simple, rapide et efficace, ce qui la rend adaptée au chiffrement de grands volumes de données
- Exemples d'algorithmes de chiffrement symétrique :

  - AES (Advanced Encryption Standard)
  - DES (Data Encryption Standard)
  - Blowfish
  - Twofish
  - RC4 (Rivest Cipher 4)

Cryptographie asymétrique

(Cryptographie à clé publique)

- Utilise une paire de clés : publique et privée
- Clé publique : peut être partagée avec n'importe qui
- Clé privée : gardée secrète par le propriétaire
- Fonctionnement : L'expéditeur utilise la clé publique du destinataire pour chiffrer les données, et le destinataire utilise sa clé privée pour les déchiffrer
- Exemples d'algorithmes de chiffrement asymétrique :

  - RSA (Rivest–Shamir–Adleman)
  - Diffie-Hellman
  - Elliptic Curve Cryptography (ECC)

Menaces des ordinateurs quantiques

- Ordinateurs classiques :

  - bits (0 ou 1)

  - calculs séquentiels utilisant des ports logiques (AND, OR, NOT)

  - capacité dépends du nombre des transistors (linéaire)

- Ordinateurs quantiques :

  - qubits, superposition (0 et 1 simultanément)

  - plusieurs processus possible grâce à la superposition

Challenge 1 : RSA

- c:
  9527279598647518950551898025113700350929262114016638388785485386372069242020414244842407483465714932685355309762648637120661751376993027758082311643797548714895610750924756496565241745055068018169186943206789202836898500722963394314909168441983413621479347691041735953769663287404527232666503671732462399288 5
- p:
  1138748058490985498512533584824038422665392994275775638448938124220615719798655524399533515832878197031060306067148668885626377645265426804393603655621524 3
- q:
  1297222287521808654742581896147725791510551570598228372685183350807960046054247926797205021683860464974287051520046235900731543184878416379031242446243962 9
- dp:
  8191957726161111880866028229950166742224147653136894248088678244548815086744810656765529876284622829884409590596114090872889522887052772791407131880103961
- dq:
  3570695757580148093370242608506191464756425954703930236924583065811730548932270595568088372441809535917032142349986828862994856575730078580414026791444659

Challenge 1 : RSA

- Hint : https://en.wikipedia.org/wiki/RSA_(cryptosystem)#Example

Challenge 1 : RSA

- e = 65537
- d = mod_inverse(p, e)
- qi = mod_inverse(q, p)

- m1 = pow(c, dp, p)
- m2 = pow(c, dq, q)

- h = (qi * (m1 - m2)) % p
- m = m2 + h * q
- decoded_m = bytes.fromhex(hex(m)[2:]).decode('utf-8')

- print(decoded_m)

Challenge 2 : RSA

- from Crypto.PublicKey import RSA
- key = RSA.generate(4096, e=5)
- msg = "welcome to uiuctf!\nyour super secret flag is: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
- m = int(msg.encode("hex"), 16)
- c = pow(m, key.e, key.n)

Challenge 2 : RSA

- n = 826280450476795403105390383916395625701073920777162153138597185953056944510888027904354828464602421249363674719063026424044747076553321187265165775178889032794638105579799203345357910166892700405175658568627675449699540840288382597105404255643311670752496397923267416409538484199324051251779098290351314013642933189000153869540797043267546151497242578717464980825955180662199508957183411268811625401646070827084944007483568527240194185553478349118552388947992831458170444924129523129671104469299148320613669401657183290772893794967935207930444530128455715930912396159031673581402512689887196340755500324027444712984725593749637947968318889725735972238835022070258644127271944675313059568048692821272117818934238685689249214608044529062871338311672093407988563237143335520310739909530999468602604401205507447372648318950975692813406759796513551693936063874856010242831791410751241160796801836410406380053401474903123702910202828454172637852004817991431486529025890690643064948035321242348503628008926468239093472083469567412208772246267654444230814321868717928257721393692548308253770155315183138383827178677363405092269401171610136046375762902332065892101184362733264374446472420477100886644068100898422212270643634477091054493275 7
- e = 5
- c = 199037898049081148054548566008626493558290050160287889209057083223407180156125399899465196611255722303390874101982934954388936179424024104549780651688160499201410108321518752502957346260593418668796624999582838387982430520095732090601546001755571395014548912727418182188910950322763678024849076083148030838828924108260083080562081253547377722180347372948445614953503124471116393560745613311509380885545728947236076476736881439654048388176520444109172092029548244462475513941506675855751026925250160078913809995564374674278235553349778352067191820570404315381746499936539482369231372882062307188454140330786512148310245052484671692280269741146507675933518321695623680547732771867757371698350343979932499637752314262246864787150534170586075473209768119198889190503283212208200005176410488476529948013610803040328568552414972234514746292014601094331465138374210925373263573292609023829742634966280579621843784216908520325876171463017051928049668240295956697023793952538148945070686999838223927548227156965116574566365108818752174755077045394837234760506722554542515056441166987424547451245495248956829984641868331576895415337336145024631773347254905002735757

Challenge 2 : RSA

- *e* est trop petit comparativement à *n* (4096 bits)
- *m* n'est très long (96 caractères)
- Ce qui veut dire *m^e* est plus petit de *n*
- *C = m^e (mod n) = m^e*

Challenge 2 : RSA

- Sol :

- import gmpy2

- m = gmpy2.iroot(c, 5)[0]
- hex_m = hex(int(m))[2:]

- Vous mettez hex_m dans https://www.rapidtables.com/convert/number/hex-to-ascii.html

Sources intéressantes

- https://ctfacademy.github.io/crypto/index.htm
- https://ctf101.org/cryptography/overview/
- https://github.com/RsaCtfTool/RsaCtfTool
- https://en.wikipedia.org/wiki/RSA_(cryptosystem)#Example
- https://medium.com/@hva314/some-basic-rsa-challenges-in-ctf-part-1-some-basic-math-on-rsa-5663fa337c27
- ** https://www.dcode.fr/rsa-cipher **