Chiffrement Symétrique
XOR Application

Éric BERTHOMIER
berthomiereric70@gmail.com

4 janvier 2026

XOR ENCRYPTION

Version 3.0 - Version Stagiaire

Question de logique

Un touriste cherche l'horloger dans une ville où une seule personne ment (le menteur) et où les autres disent la vérité (véridiques). Il interroge trois personnes (A, B, C).

- Il demande à A où est l'horloger, mais la réponse est incompréhensible.

Question de logique

Un touriste cherche l'horloger dans une ville où une seule personne ment (le menteur) et où les autres disent la vérité (véridiques). Il interroge trois personnes (A, B, C).

- Il demande à A où est l'horloger, mais la réponse est incompréhensible.
- Il demande alors à B ce qu'a dit A. B répond : *"Il dit que c'est lui l'horloger."*

Question de logique

Un touriste cherche l'horloger dans une ville où une seule personne ment (le menteur) et où les autres disent la vérité (véridiques). Il interroge trois personnes (A, B, C).

- Il demande à A où est l'horloger, mais la réponse est incompréhensible.
- Il demande alors à B ce qu'a dit A. B répond : *"Il dit que c'est lui l'horloger."*
- C intervient et dit à B : *" Tu es un menteur."*

XOR ENCRYPTION

Question de logique

Un touriste cherche l'horloger dans une ville où une seule personne ment (le menteur) et où les autres disent la vérité (véridiques). Il interroge trois personnes (A, B, C).

- Il demande à A où est l'horloger, mais la réponse est incompréhensible.
- Il demande alors à B ce qu'a dit A. B répond : *"Il dit que c'est lui l'horloger."*
- C intervient et dit à B : *" Tu es un menteur."*

XOR ENCRYPTION

Question de logique

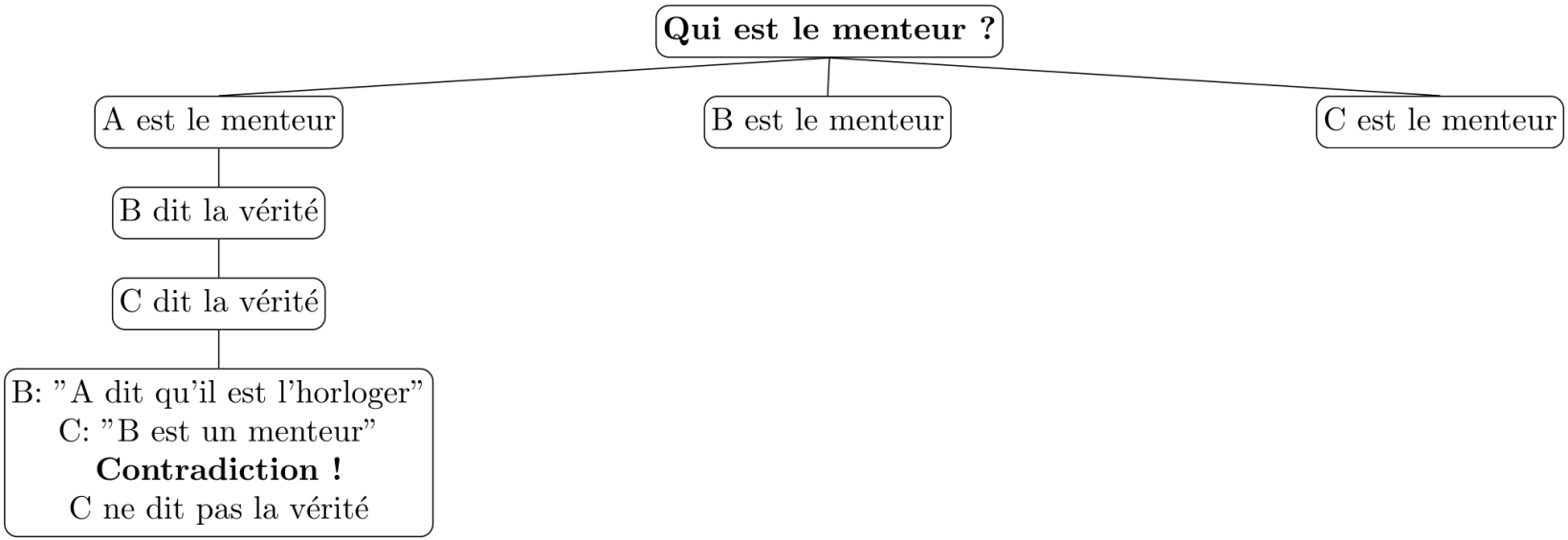
Un touriste cherche l'horloger dans une ville où une seule personne ment (le menteur) et où les autres disent la vérité (véridiques). Il interroge trois personnes (A, B, C).

- Il demande à A où est l'horloger, mais la réponse est incompréhensible.
- Il demande alors à B ce qu'a dit A. B répond : *"Il dit que c'est lui l'horloger."*
- C intervient et dit à B : *" Tu es un menteur."*

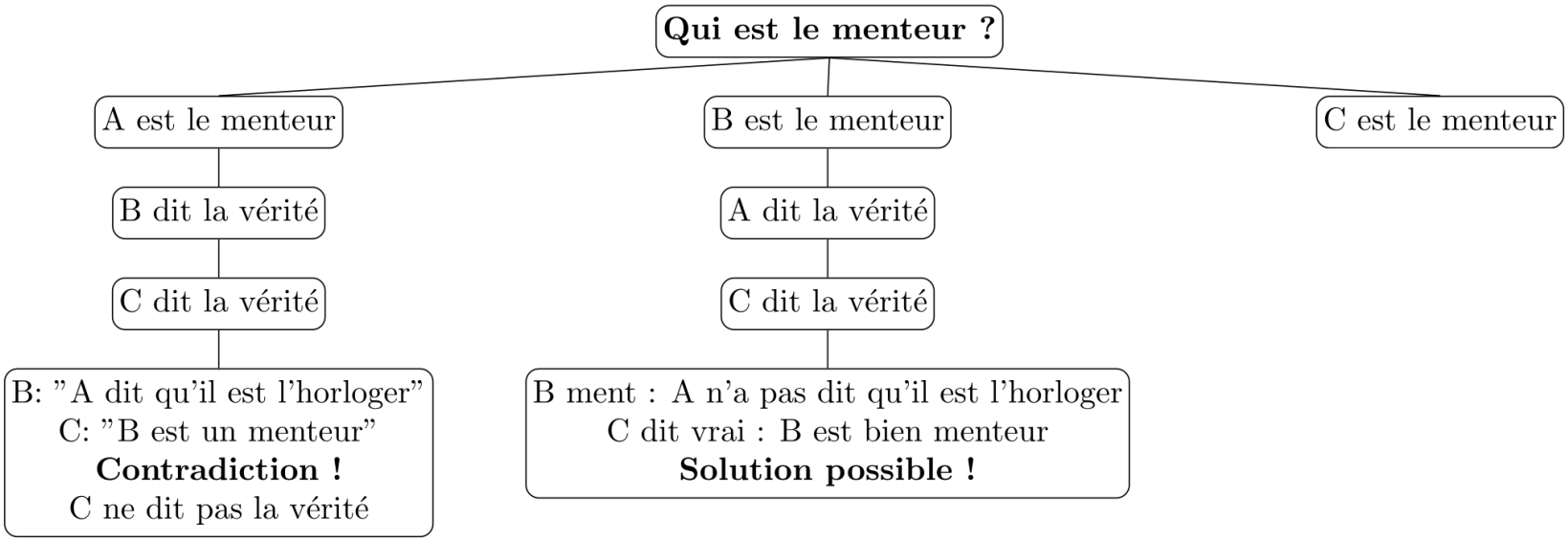
Qui est le menteur et qui est l'horloger ?

XOR ENCRYPTION

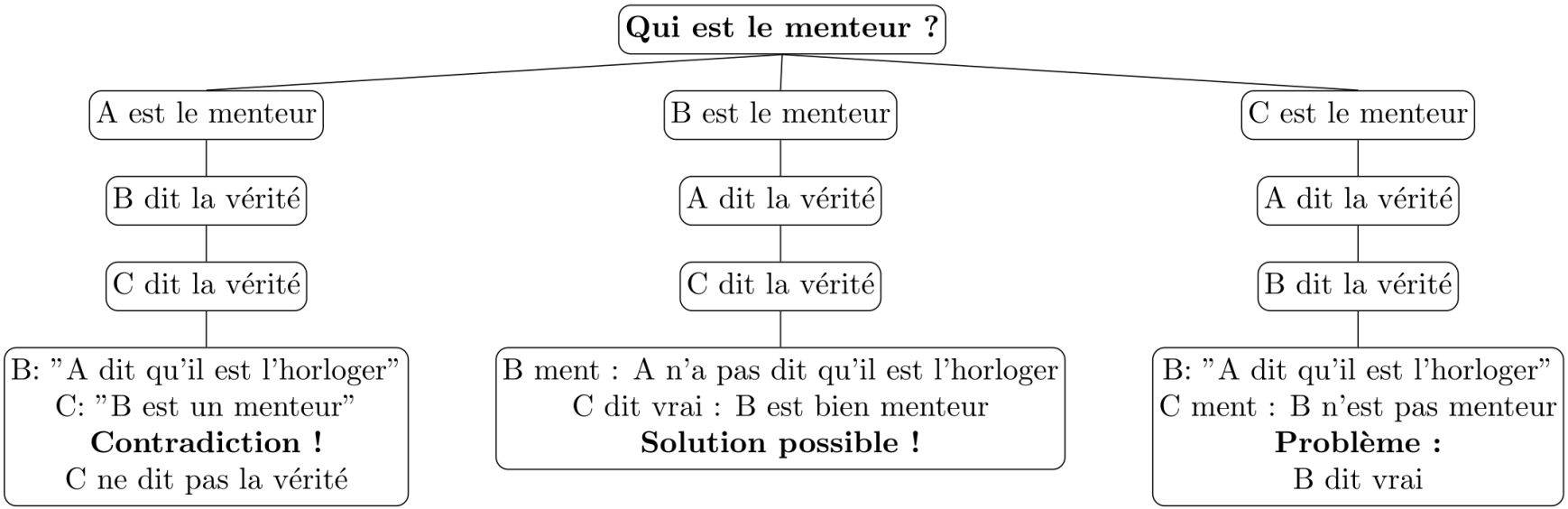
Arbre de décision



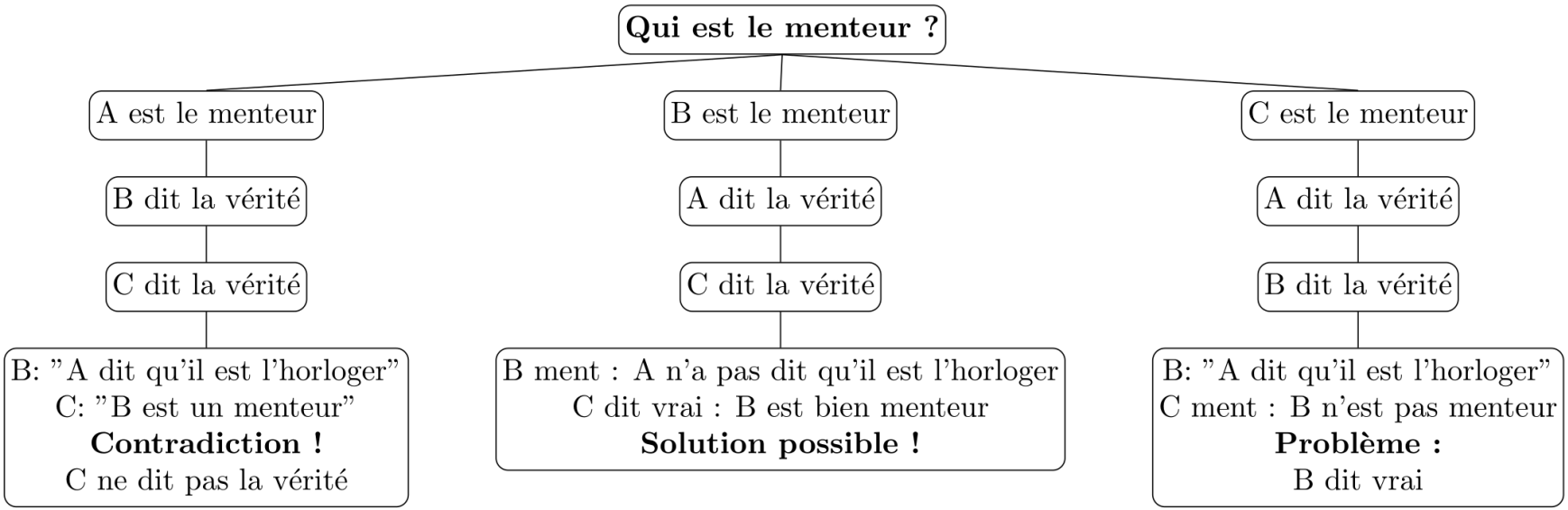
Arbre de décision



Arbre de décision

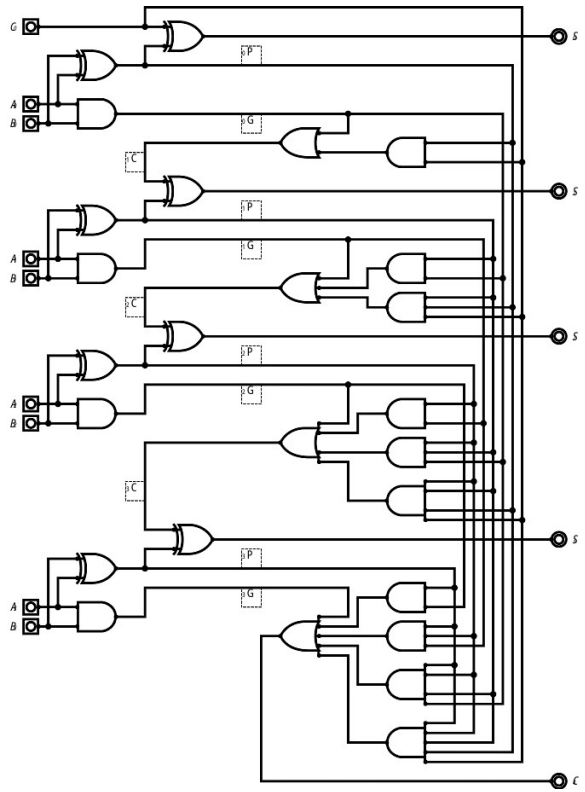


Arbre de décision



Source : Bibm@th.net

Logique dans la vie courante.



La logique en électronique : les portes logiques

Informatique : valeurs logiques

En informatique, on retrouve nos 2 valeurs logiques :

- VRAI - TRUE associée à la valeur numérique 1

Informatique : valeurs logiques

En informatique, on retrouve nos 2 valeurs logiques :

- VRAI - TRUE associée à la valeur numérique 1
- FAUX - FALSE associée à la valeur numérique 0

Informatique : opérateurs logiques

- ET - AND - Que l'on associe à une multiplication

Informatique : opérateurs logiques

- ET - AND - Que l'on associe à une multiplication
- OU - OR - Que l'on associe à une addition

Informatique : opérateurs logiques

- ET - AND - Que l'on associe à une multiplication
- OU - OR - Que l'on associe à une addition
- Mais comme en logique, on ne peut avoir que 2 valeurs 0 ou 1, on dit que 1+1=2 soit 1 (VRAI).

Table de vérité AND / OR

$$\wedge = \text{AND}$$
$$\vee = \text{OR}$$

a	b	$a \wedge b$	$a \vee b$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

a	b	$a \wedge b$	$a \vee b$
False	False	False	False
False	True	False	True
True	False	False	True
True	True	True	True

Ou exclusif : XOR

L'opérateur XOR ($\oplus$) retourne 1 si les deux valeurs sont différentes ou 0 si elles sont identiques.

a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

a	b	$a \oplus b$
False	False	False
False	True	True
True	False	True
True	True	False

Exemple simple de chiffrement avec XOR

Supposons le message : "A" (ASCII 65, binaire 01000001) et la clé de chiffrement : "b" (ASCII 98, binaire 01100010).

Exemple simple de chiffrement avec XOR

Supposons le message : "A" (ASCII 65, binaire 01000001) et la clé de chiffrement : "b" (ASCII 98, binaire 01100010).

Bit	Opérande 1	Opérande 2	Résultat XOR
0	0	0	0
1	1	1	0
2	0	1	1
3	0	0	0
4	0	0	0
5	0	0	0
6	1	1	0
7	1	0	1

Exemple simple de chiffrement avec XOR

Valeur 1	Opérateur	Valeur 2	Résultat
01000001	$\oplus$	01100010	00100001
65	$\oplus$	98	33
A	$\oplus$	b	#

Le chiffrement de A par la clé b en XOR donne le résultat : #

Exemple simple de chiffrement avec XOR

chiffrement_xor.py

```
#!/usr/bin/python3
from operator import xor
def affiche_ascii(valeur):
    print (f"Valeur texte : {valeur}")
    ordval=ord(valeur)
    print (f"Valeur Ascii : ", ordval)
    print (f"Valeur Binaire :", bin(ordval))
    print ("---")
    return (ordval)

def affiche_caractere(valeur):
    print (f"Valeur ascii : {valeur}")
    chrval=chr(valeur)
    print (f"Valeur Ascii : ", chrval)
    print (f"Valeur Binaire :", bin(valeur))
    print ("---")

valeur = "A"
cle = 'b'
chiffre = ''

ordvaleur = affiche_ascii (valeur)
ordcle = affiche_ascii (cle)
print (f"{ordvaleur} XOR {ordcle}")
chiffre = xor(ordvaleur, ordcle)
affiche_caractere (chiffre)
print (f"{chiffre} XOR {ordcle}")
doublexor = xor(chiffre, ordcle)
affiche_caractere (doublexor)
```

Problématique du chiffrement avec XOR

Cependant le chiffrement d'un texte XOR un autre texte ne donne pas forcément un texte comme résultat.

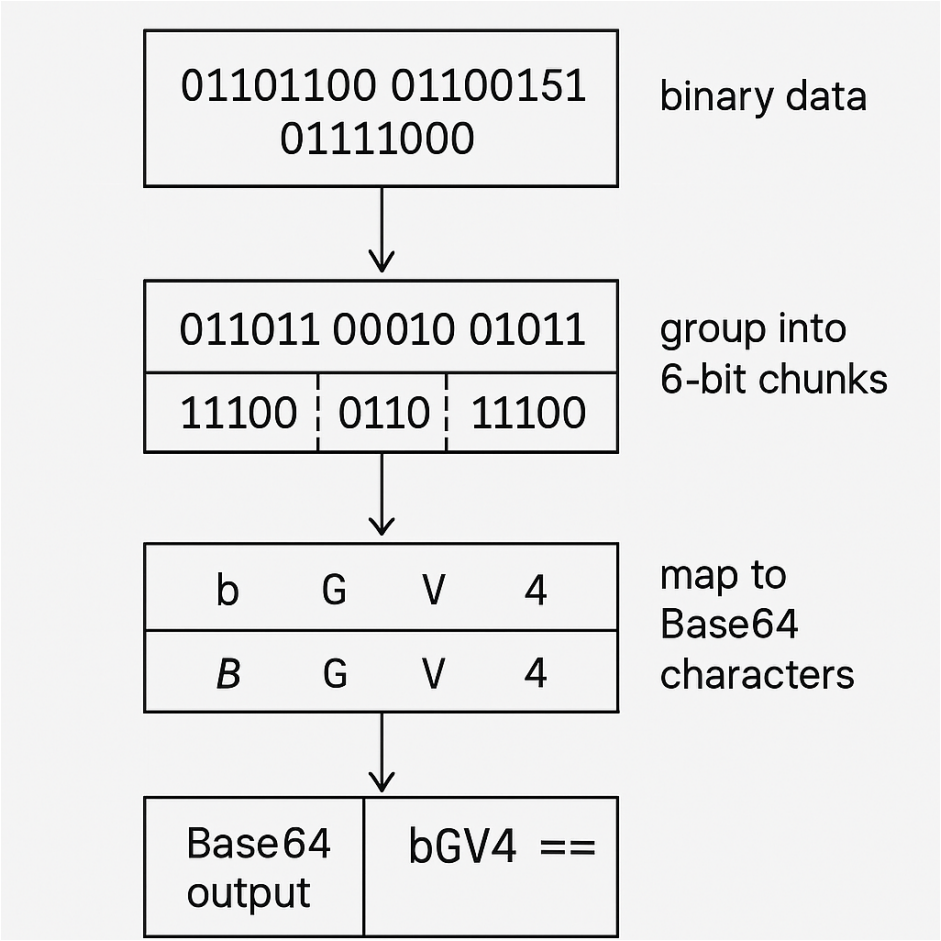
chiffrement_xor_pb.py

```python
#!/usr/bin/python3

chaine1 = "Bonjour"
chaine2 = "Clef123"

def xor_chiffrement(s1, s2):
    longueur = min(len(s1), len(s2))
    result = bytearray()
    for i in range(longueur):
        result.append(ord(s1[i]) ^ ord(s2[i]))
    return result

resultat = xor_chiffrement(chaine1, chaine2)
chaine_resultat = ''.join(chr(b) for b in resultat)
hex_resultat = resultat.hex()
hex_prefixed = ' '.join(['0x' + hex_resultat[i:i+2] for i in range(0, len(hex_resultat), 2)])

print("Résultat chaîne :", chaine_resultat)
print("Résultat hexadécimal :", hex_prefixed)
```

```
Résultat chaîne :
                ^GA
Résultat hexadécimal : 0x01 0x03 0x0b 0x0c 0x5e 0x47 0x41
```

Accompagnement avec Base64

Pour permettre de tout laisser au format texte et ainsi avoir plus de chance de traverser les défenses cyber, nous allons convertir le binaire en Base64.



Un peu plus loin avec le Base64 (1/2)

Chaque groupe de 6 bits (valeur comprise entre 0 et 63) est converti en un caractère Base64 grâce à une table de correspondance (alphabet Base64).

Alphabet Base64

A encoder	Encodage
Lettres majuscules AZ	Valeurs 025
Lettres minuscules az	Valeurs 2651
Chiffres 09	Valeurs 5261
Symboles + et /	Valeurs 62 et 63

Exemple de mapping

Valeur binaire (6 bits)	Décimal	Caractère Base64
010000	16	Q
011010	26	a
001101	13	N
111100	60	8

Encore plus loin avec le Base64 (2/2)

Gestion du remplissage

- Si la longueur des données nest pas un multiple de 3 octets, des bits à zéro sont ajoutés pour compléter le dernier groupe.
- Des caractères = sont ajoutés à la fin pour obtenir un nombre de caractères multiple de 4.

Exemple simple de chiffrement avec XOR puis Base64

chiffrement_xor_base64.py

```python
#!/usr/bin/python3
import base64

def xor_encrypt(data: bytes, key: bytes) -> bytes:
    return bytes([b ^ key[i % len(key)] for i, b in enumerate(data)])

def main():
    # Saisie utilisateur
    phrase = input("Entrez une phrase à chiffrer : ")
    cle = input("Entrez une clé de chiffrement : ")

    # Conversion en bytes
    data_bytes = phrase.encode('utf-8')
    key_bytes = cle.encode('utf-8')

    # Chiffrement XOR
    xor_result = xor_encrypt(data_bytes, key_bytes)

    # Encodage en base64
    base64_result = base64.b64encode(xor_result)

    print("Chiffrement base64 :")
    print(base64_result.decode('utf-8'))

if __name__ == "__main__":
    main()
```

Exemple d'utilisation

```
./chiffrement_xor_base64.py
Entrez une phrase à chiffrer : Je suis en cours de cyber
Entrez une clé de chiffrement : cyber
Chiffrement base64 :
KRxCFgcKCkIAHEMaDRAAEFkGAFIAAAAAAA==
```

Exemple simple de déchiffrement Base64 / XOR

dechiffrement_xor_base64.py

```python
#!/usr/bin/python3
import base64

def xor_encrypt(data: bytes, key: bytes) -> bytes:
    return bytes([b ^ key[i % len(key)] for i, b in enumerate(data)])

def main():
    # Saisie utilisateur
    message_b64 = input("Entrez le message chiffré (base64) : ")
    cle = input("Entrez la clé de chiffrement : ")

    # Décodage base64
    try:
        encrypted_bytes = base64.b64decode(message_b64)
    except Exception as e:
        print("Erreur lors du décodage base64 :", e)
        return
    # Conversion de la clé en bytes
    key_bytes = cle.encode('utf-8')

    # Déchiffrement XOR
    decrypted_bytes = xor_encrypt(encrypted_bytes, key_bytes)

    # Conversion en texte lisible
    decrypted_text = decrypted_bytes.decode('utf-8')
    print("Message déchiffré :")
    print(decrypted_text)

if __name__ == "__main__":
    main()
```

Exemple d'utilisation

```
./dechiffrement_xor_base64.py
Entrez le message chiffré (base64) : KRxCFgcKCkIAHEMaDRAAEFkGAFIAAAAAAA==
Entrez la clé de chiffrement : cyber
Message déchiffré :
Je suis en cours de cyber
```

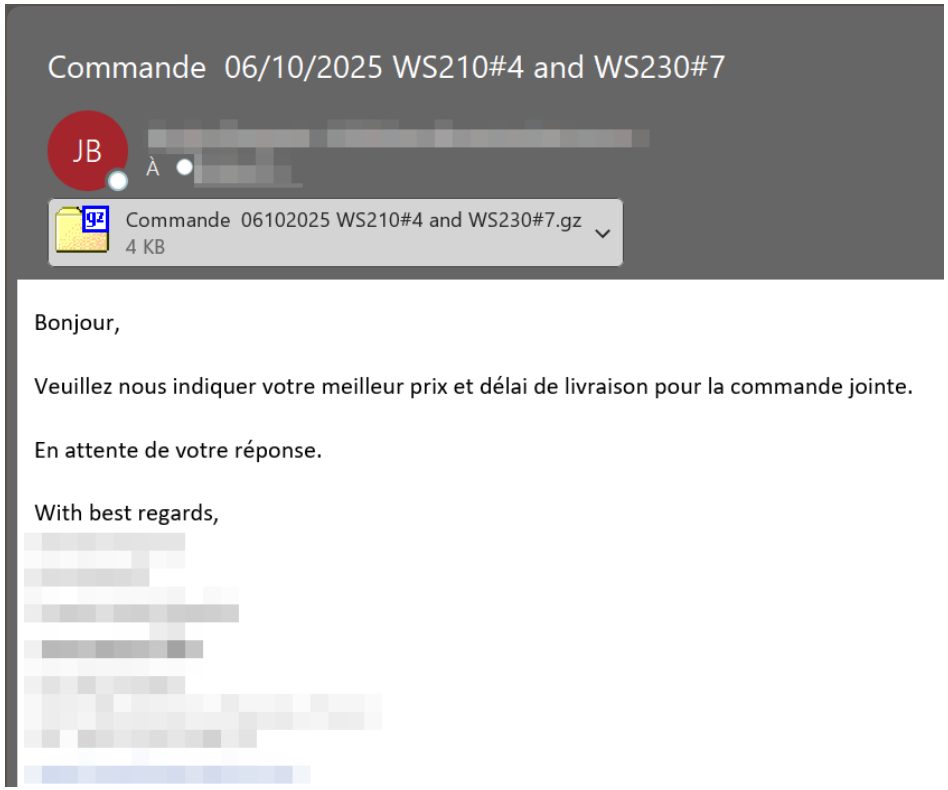
Le malware Remcos GuLoader utilise ce chiffrement.

Analyse - Joe Sand Box (12/06/2025)

XOR ENCRYPTION

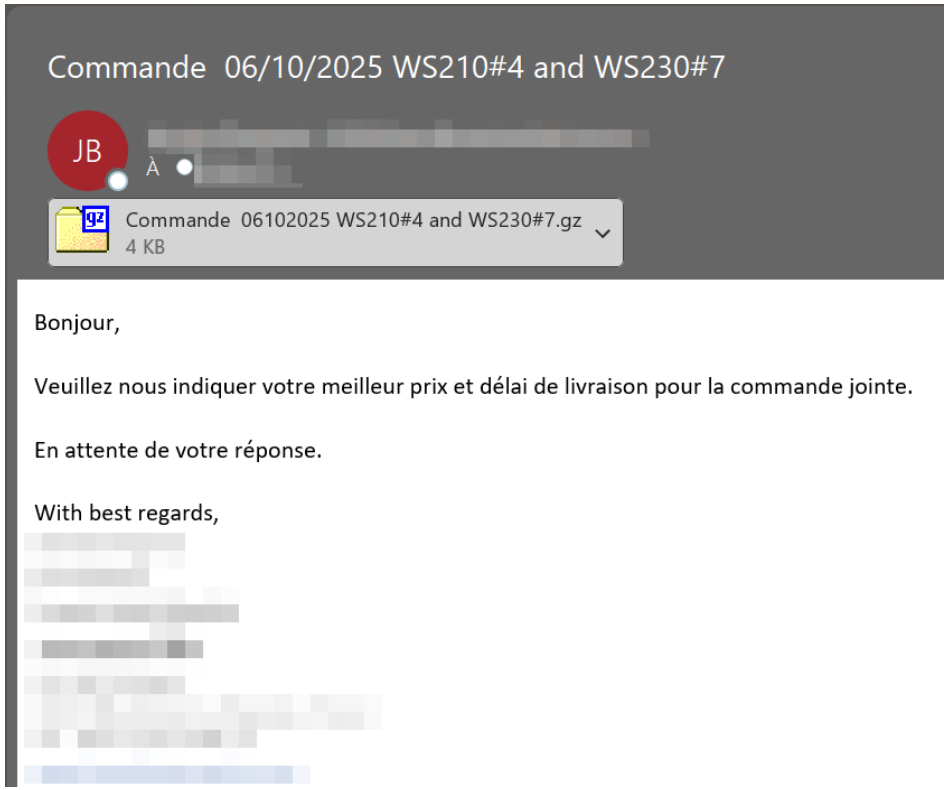
Vecteur d'infection - Courriel

Message envoyé au travers de la messagerie d'une ancienne employée d'une entreprise.



Vecteur d'infection - Courriel

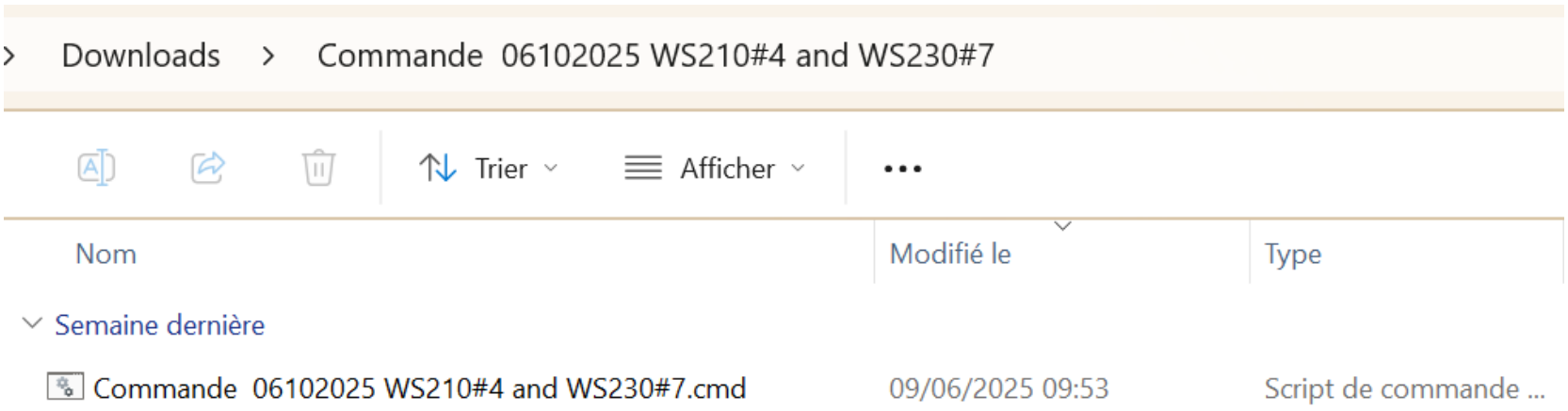
Message envoyé au travers de la messagerie d'une ancienne employée d'une entreprise.



Procédure de départ d'un salarié ?

XOR ENCRYPTION

Commande - Zip



Quelle est la problématique ?

Mon cher ami l'antivirus

1. 12/06/2025 - Jour de la découverte par nos services - Mise en place des contre-mesures.

Mon cher ami l'antivirus

1. 12/06/2025 - Jour de la découverte par nos services - Mise en place des contre-mesures.
2. 12/06/2025 - Virus Total : 1/62 - Détection heuristique

XOR ENCRYPTION

Mon cher ami l'antivirus

1. 12/06/2025 - Jour de la découverte par nos services - Mise en place des contre-mesures.
2. 12/06/2025 - Virus Total : 1/62 - Détection heuristique
3. 18/06/2025 - Virus Total : 24/62 - Détection heuristique

Mon cher ami l'antivirus

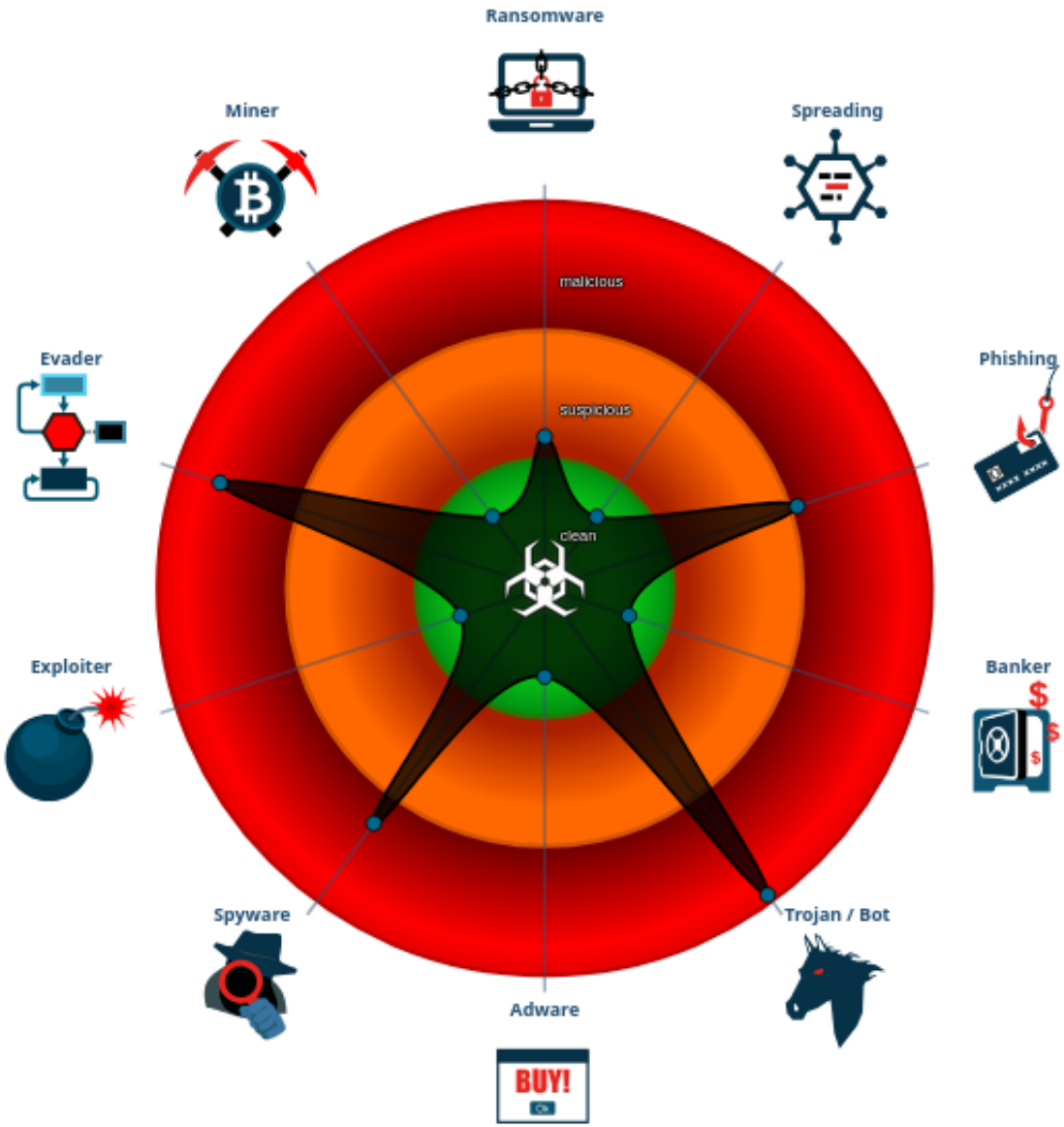
1. 12/06/2025 - Jour de la découverte par nos services - Mise en place des contre-mesures.
2. 12/06/2025 - Virus Total : 1/62 - Détection heuristique
3. 18/06/2025 - Virus Total : 24/62 - Détection heuristique
4. 21/06/2025 - Virus Total : 26/62 - Détection heuristique

XOR ENCRYPTION

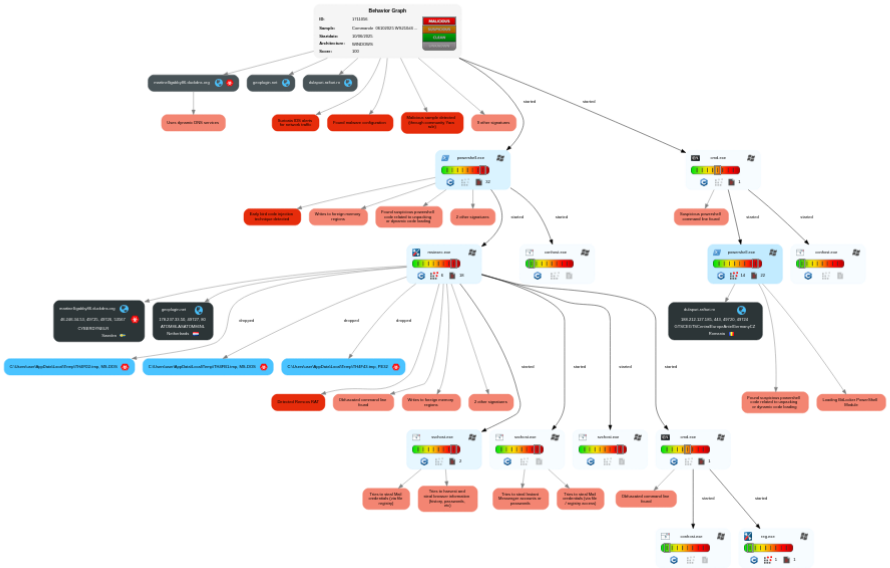
Mon cher ami l'antivirus

1. 12/06/2025 - Jour de la découverte par nos services - Mise en place des contre-mesures.
2. 12/06/2025 - Virus Total : 1/62 - Détection heuristique
3. 18/06/2025 - Virus Total : 24/62 - Détection heuristique
4. 21/06/2025 - Virus Total : 26/62 - Détection heuristique
5. À ce jour - Recherche Virus Total

XOR ENCRYPTION


Ransomware
Miner
Spreading
malicious
suspicious
clean
Evader
Phishing
Exploiter
Banker
Spyware
Trojan / Bot
Adware
BUY!

Behavior Graph



Hide Legend

Legend:

- Process
- Signature
- Created File
- DNS/IP Info
- Is Dropped
- Is Windows Process
- Number of created Registry Values
- Number of created Files
- Visual Basic
- Delphi
- Java
- .Net C# or VB.NET
- C, C++ or other language
- Is malicious
- Internet

XOR ENCRYPTION

XOR ENCRYPTION