Chiffrement affine

INNOCENTS NJIEMOUN JUDES FRANCK 20V2436
TONYE NWALAL JULES DIMITRI 20R2016
NGOUBENE EWANE JACK JUNIOR 20V2259
MBONO TEKAM ULRICH VANEL 18I2275

October 2023

Table des matières

INTRODUCTION GÉNÉRALE

Le chiffrement affine est une méthode fondamentale de cryptographie qui a joué un rôle crucial dans l'histoire de la sécurité des communications. Cette technique repose sur une transformation mathématique, où chaque caractère d'un message est substitué par un autre caractère en utilisant une fonction affine. L'opération modulo est également employée pour maintenir les résultats dans les limites de l'alphabet ou du groupe de caractères en question. Le chiffrement affine a des racines historiques profondes, ayant été utilisé dans diverses cultures pour sécuriser des messages confidentiels depuis l'Antiquité. Dans cet exposé, nous explorons en détail :

— le fonctionnement du chiffrement affine, en mettant en lumière la manière dont les paramètres de la fonction influencent la sécurité du chiffrement.

— Nous aborderons également les processus de chiffrement et de déchiffrement, tout en soulignant les vulnérabilités potentielles et les applications contemporaines de cette technique.

— En fin de compte, nous découvrirons pourquoi le chiffrement affine demeure pertinent dans le paysage de la cryptographie moderne malgré ses limites apparentes

Chapitre 1

Fonctionnement Chiffrement Affine

1.1 Fonctionnement Chiffrement

Le chiffrement affine est une technique de cryptographie qui repose sur une transformation mathématique, où chaque caractère d'un message est substitué par un autre caractère en utilisant une fonction affine. Cette fonction de chiffrement est généralement représentée sous la forme mathématique suivante :

$$E(x) = (ax + b) \mod m$$

où : - $E(x)$ est le caractère chiffré
- x est le caractère d'origine à chiffrer.
- a et b sont des paramètres de la transformation, avec a étant un entier positif et b un entier quelconque.
- m est la taille de l'alphabet ou du groupe de caractères.

1.1.1 Choix des paramètres a et b

Le choix des paramètres a et b est crucial pour la sécurité du chiffrement. a doit être soigneusement choisi pour être premier avec m, c'est-à-dire qu'il doit être co-premier avec m pour éviter des vulnérabilités. b peut prendre n'importe quelle valeur, mais il détermine le décalage de substitution, ce qui signifie que différentes valeurs de b produiront des chiffrements différents pour un même message.

1.1.2 Alphabet et modularité

L'alphabet ou le groupe de caractères est défini en fonction du contexte. Par exemple, dans le cas de l'alphabet anglais, m serait généralement égal à 26 (les 26 lettres de l'alphabet). L'opération modulo ($\mod m$) est utilisée pour s'assurer que le chiffre résultant reste dans les limites de l'alphabet. Cette opération garantit que le chiffrement affine est réversible, c'est-à-dire que vous pouvez déchiffrer un message chiffré en utilisant la fonction de déchiffrement affine.

1.1.3 Résumé du fonctionnement

En résumé, le chiffrement affine prend un caractère d'origine, l'applique à une fonction affine pour le substituer par un autre caractère, et utilise l'opération modulo pour garantir que le résultat reste dans les limites de l'alphabet. Les paramètres a et b de la fonction sont essentiels pour la sécurité, et l'alphabet ou le groupe de caractères utilisé définit la taille de l'ensemble des résultats possibles. Cette technique forme la base de nombreuses méthodes de chiffrement et est un exemple classique de cryptographie.

1.2 Fonctionment du déchiffrement

1.2.1 Processus de déchiffrement

Le processus de déchiffrement dans le chiffrement affine est l'opération inverse du chiffrement. Il permet de récupérer le message d'origine à partir du message chiffré en utilisant une fonction de déchiffrement affine. La fonction de déchiffrement affine est généralement de la forme suivante :

$$D(x) = a^{-1} \cdot (x - b) \mod m$$

où : - $D(x)$ est le caractère déchiffré, correspondant au caractère d'origine x.
- a^{-1} est l'inverse multiplicatif de a modulo m. Trouver a^{-1} est essentiel pour le déchiffrement, et il doit être calculé en utilisant les mathématiques modulaires.
- x est le caractère chiffré que l'on souhaite déchiffrer.
- b est le même paramètre de décalage utilisé lors du chiffrement.
- m est la taille de l'alphabet ou du groupe de caractères.

1.2.2 Voici comment fonctionne le processus de déchiffrement :

— 1. Tout d'abord, on identifie le caractère chiffré x que l'on souhaite déchiffrer.
— 2. Ensuite, on applique l'opération $(x - b)$ pour annuler le décalage de substitution initial.
— 3. Ensuite, on multiplie le résultat par l'inverse multiplicatif a^{-1} modulo m. Cela annule l'effet de la multiplication lors du chiffrement.
— 4. Enfin, on applique l'opération modulo m pour s'assurer que le résultat reste dans les limites de l'alphabet ou du groupe de caractères.
—

Après avoir suivi ces étapes, on obtient le caractère d'origine x, ce qui permet de reconstituer le message d'origine en appliquant ce processus à chaque caractère du message chiffré.

NB : Il est important de noter que pour que le déchiffrement fonctionne correctement, il est essentiel de connaître les paramètres a, b, et la taille de l'alphabet m, ainsi que d'avoir la connaissance de l'inverse multiplicatif de a modulo m. Ces informations sont nécessaires pour inverser la transformation et récupérer le message d'origine à partir du message chiffré.

Chapitre 2

Processus de Chiffrement et déchiffrement

2.1 Processus de Chiffrement et déchiffrement

2.1.1 Processus de chiffrement avec le chiffrement affine

Le processus de chiffrement avec le chiffrement affine consiste à transformer chaque caractère du message d'origine en un caractère chiffré en utilisant la fonction de chiffrement affine $E(x) = (ax + b) \mod m$, où a et b sont ls paramètres de la transformation, x est le caractère d'origine, et m est la taille de l'alphabet ou du groupe de caractères.

Supposons que nous voulions chiffrer le message "HELLO" en utilisant $a = 3$ et $b = 7$ avec l'alphabet anglais (26 lettres). Voici comment cela se déroulerait :

2.1.2 Message d'origine : "HELLO"

Chiffrement caractère par caractère

. - "H" devient $E(7) = (3 * 7 + 7) \mod 26 = 28 \mod 26 = 2$, donc "H" est chiffré en "C".

- "E" devient $E(4) = (3 * 4 + 7) \mod 26 = 19 \mod 26 = 19$, donc "E" est chiffré en "T".

- "L" devient $E(11) = (3 * 11 + 7) \mod 26 = 40 \mod 26 = 14$, donc "L" est chiffré en "O".

- "L" devient $E(11) = (3 * 11 + 7) \mod 26 = 40 \mod 26 = 14$, donc "L" est chiffré en "O".

- "O" devient $E(14) = (3 * 14 + 7) \mod 26 = 49 \mod 26 = 23$, donc "O" est chiffré en "X".

Message chiffré : "CTOOX"

Le message "HELLO" est donc chiffré en "CTOOX" en utilisant le chiffrement affine avec $a = 3$ et $b = 7$ dans cet exemple.

2.1.3 Processus de déchiffrement avec le chiffrement affine

Le processus de déchiffrement consiste à inverser la transformation pour retrouver le message d'origine à partir du message chiffré. Pour le chiffrement affine, la fonction de déchiffrement est généralement de la forme $D(x) = a^{-1} \cdot (x - b) \mod m$, où a^{-1} est l'inverse multiplicatif de a, x est le caractère chiffré, b est le paramètre de décalage, et m est la taille de l'alphabet.

Prenons l'exemple chiffré précédent "CTOOX" et déchiffrons-le en utilisant $a = 3$ et $b = 7$:

2.1.4 Message chiffré : "CTOOX"

Déchiffrement caractère par caractère :

- "C" devient $D(2) = (3^{-1} \cdot (2 - 7)) \mod 26 = 9 \cdot (-5) \mod 26 = -45 \mod 26 = 5$, donc "C" est déchiffré en "H".
- "T" devient $D(19) = (3^{-1} \cdot (19 - 7)) \mod 26 = 9 \cdot 12 \mod 26 = 108 \mod 26 = 4$, donc "T" est déchiffré en "E".
- "O" devient $D(14) = (3^{-1} \cdot (14 - 7)) \mod 26 = 9 \cdot 7 \mod 26 = 63 \mod 26 = 11$, donc "O" est déchiffré en "L".
- "O" devient $D(14) = (3^{-1} \cdot (14 - 7)) \mod 26 = 9 \cdot 7 \mod 26 = 63 \mod 26 = 11$, donc "O" est déchiffré en "L".
- "X" devient $D(23) = (3^{-1} \cdot (23 - 7)) \mod 26 = 9 \cdot 16 \mod 26 = 144 \mod 26 = 16$, donc "X" est déchiffré en "O".

Message déchiffré : "HELLO"

Ainsi, le message "CTOOX" est déchiffré avec succès en "HELLO" en utilisant le chiffrement affine avec $a = 3$ et $b = 7$, nous retrouvons ainsi le message d'origine.

Chapitre 3

Les Avantages et Limites du Chiffrement Affine

Le chiffrement affine possède à la fois des avantages et des limites qui déterminent son utilisation en cryptographie moderne.

3.1 Avantages et Limites du Chiffrement Affine

Le chiffrement affine possède à la fois des avantages et des limites qui déterminent son utilisation en cryptographie moderne.

3.1.1 Avantages :

Simplicité et rapidité :

Le chiffrement affine est facile à comprendre et à mettre en œuvre. Il ne nécessite que quelques opérations mathématiques simples, ce qui le rend rapide pour le chiffrement et le déchiffrement.

Structure modulaire :

L'utilisation de l'opération modulo assure que le chiffrement affine reste réversible, ce qui est essentiel pour le déchiffrement. Il garantit également que le chiffrement affine fonctionne pour des alphabets ou groupes de caractères de différentes tailles.

Polyvalence :

Le chiffrement affine peut être utilisé pour chiffrer à la fois des messages textuels et des données numériques. Il peut être adapté à diverses applications.

3.1.2 Limites :

Vulnérabilité aux attaques de force brute :

Le chiffrement affine peut être vulnérable à des attaques de force brute, où un attaquant essaie toutes les combinaisons possibles de valeurs de a et b pour déchiffrer un message. Cela le rend moins sûr pour la protection de données sensibles.

Faible résistance à l'analyse statistique :

Le chiffrement affine ne masque pas bien les statistiques du texte chiffré. Par exemple, la fréquence des lettres dans le texte chiffré peut donner des indices à un attaquant.

Dépendance des paramètres :

La sécurité du chiffrement affine dépend fortement du choix des paramètres a et b. Si ces paramètres ne sont pas choisis correctement, le chiffrement peut être compromis.

3.1.3 Utilisation du chiffrement affine en Cryptographie Moderne

Le chiffrement affine, en raison de ses vulnérabilités potentielles, n'est généralement pas utilisé pour protéger des données hautement sensibles en cryptographie moderne. Cependant, il trouve encore des applications dans certains domaines, notamment :

Éducation en cryptographie :

Le chiffrement affine est souvent utilisé dans les cours d'introduction à la cryptographie pour enseigner les concepts fondamentaux de la substitution et des opérations mathématiques modulaires.

Exemples pédagogiques :

Il est également utilisé comme exemple pédagogique pour expliquer la réversibilité du chiffrement, la multiplication modulaire et les concepts de clés de chiffrement.

Cryptographie légère :

Dans certaines applications où la sécurité n'est pas la principale préoccupation, le chiffrement affine peut être utilisé en raison de sa simplicité et de sa légèreté.

Dans l'ensemble, le chiffrement affine est un outil pédagogique important pour comprendre les principes de base de la cryptographie, mais il est rarement utilisé pour la sécurité des données sensibles en raison de ses vulnérabilités aux attaques. La cryptographie moderne repose sur des méthodes plus complexes et sécurisées pour protéger les informations confidentielles.