

UNIVERSITETI I PRISHTINËS “HASAN PRISHTINA”
FAKULTETI I SHKENCAVE MATEMATIKO-NATYRORE
DEPARTAMENTI I MATEMATIKËS
PROGRAMI: SHKENCA KOMPJUTERIKE



Lënda: Siguria e të dhënave
Algoritmi Affine Cipher

Punuar nga:

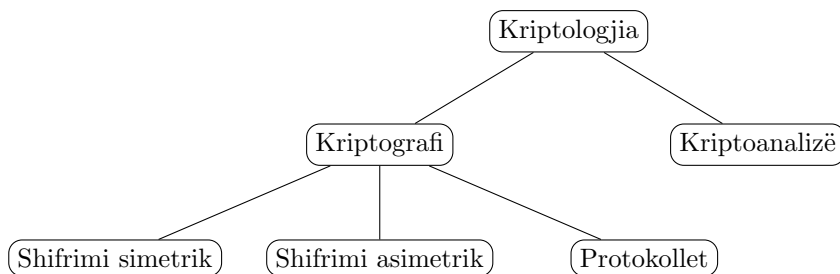
1. Rinesë Morina
2. Rona Latifaj
3. Eljesa Kqiku

Mars 2023

Hyrje në kriptologji

Kriptografia është studimi i teknikave matematikore për përdorim në të gjitha aspektet e sigurisë së të dhënave. Kriptoanaliza është shkenca e cila merret me studimin e metodave për “thyerjen” apo keqpërdorimin e këtyre metodave. Kriptologjia është studimi i kriptografisë dhe kriptoanalizës. Siguria e të dhënave mbulon këto aspekte:

- konfidencialitetin apo privatësinë
- integritetin e të dhënave
- autentikimin
- validimin e saktë.



Në mesin e algoritmeve kriptografik shifrimi Afin bën pjesë në shifrimet simetrike, që do të thotë se përdor të njëtin çelës për shifrim dhe deshifrim.

Përshkrimi

Shifra Affine është një lloj shifrimi me zëvendësim ku çdo shkronjë në tekstin e thjeshtë zhvendoset për disa pozicione. Është një shifër monoalfabetike, që do të thotë se çdo shkronjë zëvendësohet me të njëjtën shkronjë gjatë gjithë mesazhit. Megjithatë, ndryshe nga Shifra e Cezarit, Shifra Affine përdor një formulë matematikore pak më komplekse për të kriptuar mesazhin.

Për të enkriptuar një mesazh duke përdorur shifrën Affine, zgjidhen dy numra, a dhe b . Shkronjat e tekstit të thjeshtë përfaqësohen më pas me numra sipas pozicionit të tyre në alfabet

$$A = 0, B = 1, C = 2, \dots$$

dhe më pas transformohen duke përdorur formulën e mëposhtme:

$$C = (a * P + b) \bmod 26$$

ku C është shkronja e tekstit të shifruar, P është shkronja e tekstit të thjeshtë (e përfaqësuar nga një numër) dhe mod 26 do të thotë që rezultati është marrë moduli 26 (d.m.th., pjesa e mbetur kur ndahet me 26).

Për të deshifruar tekstin e shifruar, përdoret formula e mëposhtme:

$$P = a^{-1} * (C - b) \bmod 26$$

ku a^{-1} është inversi multiplikativ modular i a (d.m.th., numri që, kur shumëzohet me a , jep rezultatin 1 modulo 26).

Historia

Shifra Affine është një nga llojet më të vjetra të metodave të kriptimit, që daton në kohët e lashta, kur grekët dhe romakët e përdornin atë për të mbrojtur sekretet e tyre. Në fakt, vetë Julius Caesar thuhet se ka përdorur një version të thjeshtë të Affine Cipher për t'u dërguar mesazhe gjeneralëve të tij gjatë fushatave. Sidoqoftë, matematika e përdorur në versionin modern të Affine Cipher nuk u zhvillua deri në shekullin e 15-të, kur matematikani italian Leon Battista Alberti krijoi një numër polialfabetik të bazuar në një formulë të ngjashme me Affine Cipher. Ky shifër u përdor për qëllime diplomatike dhe ushtarake dhe u bë i njohur si Shifra Alberti.

Zhvillimi i vërtetë i Affine Cipher erdhi në shekullin e 18-të me punën e matematikanit francez Charles-Jean de la Vallée-Poussin. Ai botoi një punim në 1892 mbi përgjithësimin e Affine Cipher, i cili lejoi një gamë më të gjerë të çelësve të mundshëm dhe e bëri shifrën më të sigurt. Shifra Affine është një lloj shifrimi zëvendësues që kombinon përdorimin e aritmetikës modulare me një funksion të thjeshtë linear. Funksioni linear është i formës $ax + b$, ku a dhe b janë numra të plotë të zgjedhur nga përdoruesi dhe x është shkronja e tekstit të thjeshtë që kodohet. Aritmetika modulare përdoret për të siguruar që shkronja e tekstit të shifruar që rezulton është një shkronjë e vlefshme në alfabetin e zgjedhur.

Pavarësisht nga rëndësia e tij historike, Shifra Affine konsiderohet të jetë një formë relativisht e dobët e enkriptimit sot, dhe ajo thyhet lehtësisht nga teknikat moderne të kriptanalizës. Megjithatë, ai ende mbetet një shifër e rëndësishme historike dhe shërben si një hyrje e dobishme në studimin e kriptografisë. Affine Cipher ka shërbyer gjithashtu si bazë për zhvillimin e shifrave më të avancuara, të tilla si Hill Cipher dhe RSA Cipher. Përdorimi i tij është dokumentuar në ngjarje të shumta historike, si burgimi i Mary Queen of Scots në shekullin e 16-të, ku ajo përdori shifrën për të komunikuar fshehurazi me mbështetësit e saj. Megjithatë, shifra u thye përfundimisht nga kapësit e saj, duke çuar në ekzekutimin e saj. Pavarësisht nga dobësitë e tij, Affine Cipher ka pasur një histori të gjatë dhe të rëndësishme në zhvillimin e kriptografisë dhe vazhdon të studiohet dhe përdoret sot.

Në matematikë (specifikisht në algjebërën lineare) transformimet affine janë kombinime të transformimeve lineare dhe translacioneve. Pra, një pasqyrim i formës:

$$x \rightarrow ax + b$$

ku a dhe b janë konstante të pavarura nga x . Kjo është pikërisht forma se si është i definuar operatori i enkriptimit tek shifrimi affin, dhe pikërisht nga këtu mendohet të jetë derivuar emri i këtij shifrimi. Para analizës së frekuencave të shkronjave në kriptografi, shifrimet me zëvendësim monoalfabetik ishin të pathyeshëm dhe të pranueshëm për përdorimet e nevojshme. Me zhvillimet e mëvonshme në krypto-analizë mesazhet e enkriptuara me shifime monoalfabetike u bënë lehtë të thyeshëm. Shifrimi affin i takon kësaj familjeje si rast i vecantë i shifrit me zëvendësim monoalfabetik.

Dobësitë e shifrit affin

Shifrimi affin i takon klasës së shifrimeve me zëvendësim mono-alfabetik, dhe si shkak i saj trashëgon edhe dobësitë e kësaj klase në të cilën bejnë pjesë edhe shifrimi i Cezarit dhe shifrimi Atbash me vlerat e parametrave $a = 1$ dhe $a = -1$ përkatësisht. Duke konsideruar se ekzistojnë 286 versione jo-triviale të shifrit affin, duke mos i numruar 26 shifrimet affine triviale. Ky numër rrjedhë nga fakti se ekzistojnë 12 numra më të vogël se 26 të cilët janë ko-prim me të - pra, këto janë mundësitë e përzgjedhjes së numrit a . Tani, secila vlerë e numrit a mund të ketë 26 vlera për parametrin b . Rrjedhimisht, $12 \times 26 = 312$ çelësa të mundshëm. E duke e krahasuar në Parimin(Ligjin) e Kerckhoffs-it ky shifrim e kategorizon sistemin Affin si shumë të pasigurtë.

Dobësia primare e këtij shifrimi është kryptoanaliza, e cila ndihmën e metodave si: analiza e frekuencës apo brute force mund të gjejë plaintext-et e dy cyphertext-eve, nga të cilat çelësi mund

të gjindet duke zgjedhur ekuacione multi-vlerëshe. Fakti se parametrat a dhe m janë relativisht prim mund të përdoret për zvogëlimin e bashkësisë së mundësive të celësit në një sistem të automatizuar.

Raste të përdorimit

Affine Cipher është një metodë relativisht e thjeshtë enkriptimi që mund të përdoret në një sërë situatash ku kërkohet një nivel bazë sigurie. Disa raste të përdorimit të zakonshëm përfshijnë:

- Enkriptimi i mesazheve ndërmjet dy palëve që kanë rënë dakord për vlerat a dhe b
- Kriptimi i fjalëkalimeve ose të dhënave të tjera të ndjeshme që nuk kërkojnë siguri të nivelit të lartë
- Demonstrimi i parimeve bazë kriptografike në mjediset arsimore

Algoritmi formal

Enkriptimi

HYRJA: Numrat a dhe b që formojnë çelësin (a, b) , gjatësia e mesazhit n , vargu i karakterëve të mesazhit origjinal $P=p_i$, ku $1 \leq i \leq n$

DALJA: Vargu i karakterëve të enkriptuar $C=c_i$, ku $1 \leq i \leq n$

HAPI 1: Për çdo $i = 1, 2, \dots, n$ ekzekuto Hapin 2

HAPI 2: Nëse $p_i = \text{" "}$ atehere:
 $c_i = \text{" "}$;
Përndryshe
 $c_i = (a * p_i + b) \bmod 26$;

HAPI 3: Kthe($C=c_i \ 1 \leq i \leq n$);
Ndalo;

Dekriptimi

HYRJA: Numrat a dhe b që formojnë çelësin (a, b) , gjatësia e mesazhit n , vargu i karakterëve të mesazhit të enkriptuar $C=c_i$, ku $1 \leq i \leq n$

DALJA: Vargu i karakterëve të dekriptuar $P=p_i$, ku $1 \leq i \leq n$

HAPI 1: Për çdo $i = 1, 2, \dots, n$ ekzekuto Hapin 2

HAPI 2: Nëse $c_i = \text{" "}$ atehere:
 $p_i = \text{" "}$;
Përndryshe
 $p_i = a^{-1} * (c_i - b) \bmod 26$;

HAPI 3: Kthe($P=p_i \ 1 \leq i \leq n$);
Ndalo;

Shembull

Supozoni se duam të enkriptojmë mesazhin "HELLO" duke përdorur kodin Affine me $a = 5$ dhe $b = 8$. Së pari, ne përfaqësojmë çdo shkronjë si një numër sipas pozicionit të saj në alfabet:

<i>H</i>	<i>E</i>	<i>L</i>	<i>L</i>	<i>O</i>
↓	↓	↓	↓	↓
7	4	11	11	14

Më pas, aplikojmë formulën e enkriptimit për secilën shkronjë:

$$C(H) = (5 * 7 + 8) \bmod 26 = 11 = R$$

$$C(E) = (5 * 4 + 8) \bmod 26 = 22 = C$$

$$C(L) = (5 * 11 + 8) \bmod 26 = 9 = L$$

$$C(L) = (5 * 11 + 8) \bmod 26 = 9 = L$$

$$C(O) = (5 * 14 + 8) \bmod 26 = 12 = A$$

Prandaj, mesazhi i koduar është "RCLLA". Për të deshifruar mesazhin, përdorim formulën e deshifrimit me $a^{-1} = 21$ (pasi $5 * 21 \equiv 1 \bmod 26$):

$$P(R) = (21 * (11 - 8)) \bmod 26 = 7 = H$$

$$P(C) = (21 * (22 - 8)) \bmod 26 = 4 = E$$

$$P(L) = (21 * (9 - 8)) \bmod 26 = 11 = L$$

$$P(L) = (21 * (9 - 8)) \bmod 26 = 11 = L$$

$$P(A) = (21 * (12 - 8)) \bmod 26 = 14 = O$$

Prandaj, mesazhi i deshifruar është "HELLO"

Në këtë shembull, ne kemi përdorur $a = 5$ dhe $b = 8$ si çelësat e enkriptimit. Sidoqoftë, çdo vlerë e a dhe b mund të përdoret për sa kohë që ekziston inversi multiplikativ i a modulo 26. (pra a dhe 26 janë relativisht të thjeshtë).

Sa më të mëdha të jenë vlerat e a dhe b , aq më i sigurt do të jetë kriptimi, por edhe aq më i vështirë do të jetë deshifrimi pa i ditur çelësat. Në përgjithësi, Affine Cipher është një metodë e thjeshtë por efektive e kriptimit që është përdorur për shekuj. Ndonëse mund të mos jetë aq i sigurt sa algoritmet moderne kriptografike, mund të jetë ende i dobishëm në situata të caktuara ku kërkohet kriptimi bazë.

Implementimi në Java

```
1 public class AffineCypher {  
2     static int a = 17;  
3     static int b = 20;  
4  
5     static String encryptMessage(char[] msg)
```

```

6      {
7
8          String cipher = "";
9          for (int i = 0; i < msg.length; i++) {
10             if (msg[i] != ' ')
11                 cipher = cipher
12                     + (char) (((a * (msg[i] - 'A')) + b) % 26) + 'A');
13             else
14                 cipher += msg[i];
15         }
16         return cipher;
17     }
18
19     static String decryptCipher(String cipher)
20     {
21         String msg = "";
22         int a_inv = 0;
23         int flag = 0;
24
25         for (int i = 0; i < 26; i++) {
26             flag = (a * i) % 26;
27             if (flag == 1)
28                 a_inv = i;
29         }
30         for (int i = 0; i < cipher.length(); i++) {
31             if (cipher.charAt(i) != ' ')
32                 msg = msg + (char) (((a_inv *
33                     ((cipher.charAt(i) + 'A' - b)) % 26)) + 'A');
34             else
35                 msg += cipher.charAt(i);
36         }
37         return msg;
38     }
39 }

```