SHIFRLASH ALGORITMILARI, RSA ALGORITMI

Rasulov Islom Jovli o‘g‘li
Termiz Davlat Universiteti 2-kurs magistranti

ANNOTATSIYA

Maqola axborot xavfsizligi tahdidlarining kompyuter tizimlariga ta’siriga bag‘ishlangan. Axborotni muhofaza qilishning asosiy xarakteristikalari, tamoyillari, usullari va mexanizmlarini, shuningdek, Eyler algoritmi orqali axborotni muhofaza qilishning keng doiradagi algoritmlari, mexanizmlarini belgilovchi tizimningning tashkiliy-texnologik va inson-mashina xususiyati kabi masalalar ko‘rib chiqiladi.

Kalit so‘zlar: algoritm, Eyler funksiyasi, axborotga taxdidlar,tashkiliy va texnologik echimlar, xavfsizlik quyi tizimi.

Xozirgi kunda butun dunyo hayotining barcha jabhalari ishonchli tarzda raqamli formatga o‘tmoqda, shu jumladan bu tizimga nisbatan qarashlar har doim ham yoqimli ko‘rinishga ega emasdir. Axborotga tahdidlar jinoyatlar axborot maydoniga kirib bormoqda va har birimiz o‘ziga xos raqamli tasvir (avatar) sifatida shaxsiy manfaatlar uchun bizning ma’lumotlarimizga kirishga harakat qilinmoqda. Turli raqamli sohalarda turli tarmoqlarda tahdidlariga duchor bo‘lmoqdamiz. Kasperskiy laboratoriyasining ma’lumotlariga ko‘ra, o‘tgan 2020-yilda to‘lov dasturiga to‘lov va biznesning to‘xtab qolishi bilan bog‘liq global zarar 40 milliard dollardan ko‘proqni tashkil etgan, to‘lov dasturi hujumlarining 49 foizi to‘lov dasturi yordamida amalga oshirilgan. Shunday qilib, ma’lumotlarning xavfsizligi, shubhasiz, zamonaviy jamiyatning o‘ta muhim vazifasidir.

Yuqorida aytib o‘tilganidek, xakerlik hujumlarining 50% ga yaqini "ransomware" deb ataladigan dastur yordamida amalga oshiriladi. Ransomware - bu to‘lovni olish uchun foydalanuvchining kompyuteridagi ma’lumotlarni shifrlaydigan zararli kompyuter dasturi, lekin uni yashirish va saqlash uchun raqamli ma’lumotlarni shifrlaydigan kompyuter dasturi deb ham atash mumkin, bu holda to‘lov dasturi himoyachi hisoblanadi. Kompyuter dasturi salbiy g‘oyalar tufayli kiberqurolga aylanishi mumkin. Biroq, turli holatda ham ishlash printsipi taxminan bir xil va bir xil algoritmlarga asoslangan. Ochiq ma’lumotlarni shifrlash dasturini yaratish uchun eng mashhur, klassik algoritmni ko‘rib chiqing.

RSA – algoritmi ma’lumotlarni kodlash va dekodlash uchun turli xil bir tomonlama funktsiyalarga asoslangan ma’lumotlarni shifrlash algoritmi. Bu shifrlash uchun ochiq kalitni va shifrni ochish uchun shaxsiy kalitni yaratadi, shuning uchun

ma’lumotlarni shifrlash jarayoni juda oddiy va tez ish hisoblanadi. Shifrlashning bunday yondashuvi simmetrik shifrlashga javob sifatida assimetrik shifrlash deb ham ataladi, bunda shifrlash va shifrni ochish kalitlari bir xil bo‘ladi. Asosan, assimetrik algoritmning ishlashini quyidagicha ifodalash mumkin: Umumiy (qulf) va shaxsiy (qulflash kaliti) kalitlarni yaratgan va xavfsiz bo‘lmagan aloqa orqali faqat ochiq kalitni (qulf) uzatgan ma’lum bir foydalanuvchi Foydalanuvchi1 bor. kanalni boshqa Foydalanuvchi2 foydalanuvchisiga yuboradi, u o‘z navbatida ushbu kalit (qulf) yordamida xabarni shifrlaydi va xuddi shu kanal orqali o‘z orginalligida shaxsiy kaliti (qulflash kaliti) bo‘lgan foydalanuvchi Foydalanuvchi1 uzatadi. xabarning shifrini ochishi mumkin (aslida qulfni ochish rasmda keltirilgan).

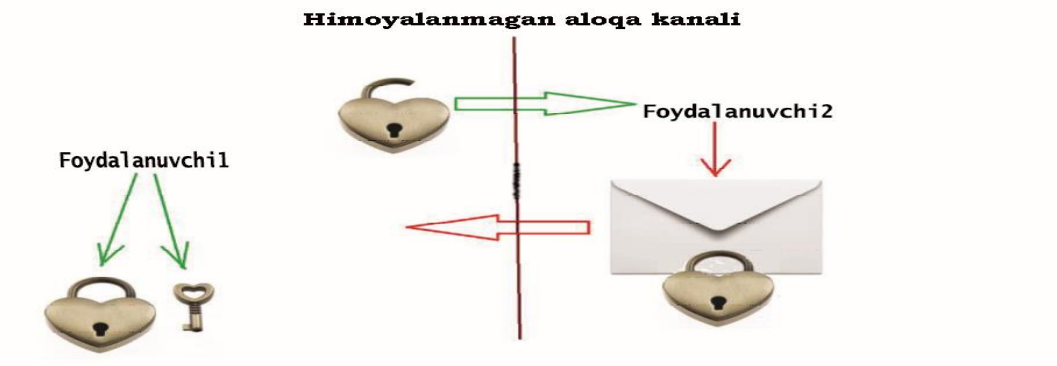
Shu bilan birga, aloqa kanaliga kirish huquqiga ega bo‘lgan ma’lum bir "buzg‘unchi" ochiq kalitga kirish huquqiga ega ekanligini va hatto undan o‘zi foydalanishi mumkinligini tushunish kerak, ammo bu algoritmni shifrni ochish (qulfni buzish) vazifasi juda murakkab. Buni hal qilish uchun uzoq yillar kerak bo‘lishi mumkin, chunki bu muammo matematik nuqtai nazardan katta butun sonlarni koeffitsientlarga ajratish, ya’ni sonni barcha mumkin bo‘lgan omillarga ajratish muammosiga tushiriladi.

Asimmetrik shifrlash algoritmining ishlash sxemasi

RSA algoritmining ishlash printsipini batafsil ko‘rsatish uchun biz ushbu jarayonning asosiy parametrlarini - shifrlash kalitlarini nazorat qilishda ma’lumotlarni shifrlash va shifrlash imkonini beruvchi kompyuter modelini yaratamiz. Modellashtirish uchun biz Microsoft Visual Studio integratsiyalashgan ishlab chiqish muhiti, uning Windows Forms kengaytmasi va C# dasturlash tilidan foydalanamiz.

Shifrlash kalitlarini yaratish algoritmi 5 bosqichdan iborat bo‘lib, biz ularni quyidagicha tavsiflaymiz:

```
// 1) p va q ikkita tub sonini hosil qiling (interval istalgan bo‘lishi mumkin)
int p = GetSimpleNumber(100, 150);
```



```
int q = GetSimpleNumber(100, 150);
  // 2) Modulni hisoblang - p va q ko‘paytmasi:  n = p×q.
  int n = p*q;
  // 3) Eyler funksiyasini hisoblang: ph = (p - 1)×(q - 1).
int f = (p - 1) * (q - 1);
// 4) e sonini aniqlang - ochiq ko‘rsatkich
int e = GetOpenExponent(f);
// 5) d sonini hisoblang - maxfiy ko‘rsatkich (maxfiy kalit)
int d = GetSecretExponent(e,f,999);
// Raqamlar juftligi [e,n] - ochiq kalit
// Xabarni shifrlash, bu erda m - matn
double Ec = Shifrlash (m, e, n);
// Raqamlar juftligi [d,n] - shaxsiy kalit
// Bizning xabarimiz shifrini ochish
double Dc = Shifrni ochish (Ec, d, n);
```

Agar tub sonlarni yaratishda hamma narsa juda oddiy bo‘lsa, ochiq ko‘rsatkichni hisoblash uchun quyidagi shartlarni hisobga olish kerak:

1) e soni tub bo‘lishi va Eyler funksiyasining f qiymatiga mos kelishi kerak.

2) e sonining qiymati f dan kichik bo‘lishi kerak.

To‘g‘ridan-to‘g‘ri shifrlashga o‘tishdan oldin, biz ushbu jarayonni alohida Shifrlash (m, e, n) usulida tasvirlaymiz, biz d maxfiy ko‘rsatkichini hisoblashimiz kerak. d soni e moduli f ga o‘zaro, ya’ni d*e ko‘paytmaning f modulining qolgan qismi 1 (1) ga teng bo‘lishi kerak.

f mod (d*e) = 1 (1)

Dasturda GetSecretExponent(e,f,ot) usuli bu amal uchun javob beradi.

```
public int GetSecretExponent(int e, int f, int ot = 999)
      {   // (d*e) mod f = 1
for (int d = ot; ; d++)
 if ((d * e) % f == 1)                    return d;
 }
```

Shifrlash algoritmi bir tomonlama funktsiyani c(m,e,n) (2) hisoblash uchun qisqartiriladi.

$C(m,e,n) = m^e \bmod n$ (2)

Shifrni ochish algoritmi [4] bir tomonlama funktsiyani m(c,d,n) (3) hisoblashga qisqartiriladi.

$m(c,d,n) = e^d \bmod n$ (3)

Quyida Encryption(m, e, n) shifrlash usuli qanday ishlashiga misol keltirilgan:

```
public string Encription(string m, double e, int n)
```

```
        { // [e,n] - ochiq kalit        C = (m^e) mod n;
string c = "";
for (int i = 0; i < m.Length; i++)
c += (char)(int)BigInteger.ModPow((int)m[i], (int)e,n);
return c;
        }
```

Algoritmni tushunish qulayligi uchun manba matnining har bir belgisining raqamli ko‘rinishi (Unicode jadvalidan) ishlatilishiga e’tibor qaratishingiz kerak. Shunday qilib, har bir belgi o‘ziga xos tarzda kodlanadi. Albatta, bunday algoritm kriptoga chidamli bo‘lmaydi, bu shunchaki o‘quv modeli, ammo qo‘shimcha shifrlash usullari bilan birgalikda bu algoritm hali ham eng ishonchli va keng tarqalganlardan biri hisoblanadi. Shifrni hal qilish usuli shifrlash usuliga o‘xshaydi, shuning uchun uning tavsifini tashlab qo‘yish mumkin, ammo, agar u vizual interfeysga ega bo‘lmasa, yuqoridagi model to‘liq bo‘lmaydi.

FOYDALANILGAN ADABIYOTLAR RO‘YXATI: (REFERENCES)

1. Goldwasser S., Bellare M. Kriptografiya bo‘yicha ma’ruza yozuvlari // IEEE Trans. xabar bering. nazariya. , 289-bet.
2. Stinson D.R. Kriptografiya: nazariya va amaliyot. Boka Raton: CRC Press, 1995. 434 p. 40
3. El-Gamal kodeksi [Elektron resurs]. URL: https://studme.org/239583/informatika/shifr_gamalya (kirish 14/03/2020).
4. Petrushenko A.A. El-Gamal kriptotizimi [Elektron resurs]. URL: https://moluch.ru/conf/tech/archive/164/9306/ (kirish sanasi: 04/12/2020).
5. Elgamal T. Ochiq kalit kriptotizimi va diskretga asoslangan imzo sxemasi
6. Logarifmlar // MA’LUMOT NAZARIYASI BO‘YICHA IEEE TRANSAKTSIYALARI. 1985. No 4. S. 4.