RSA

RSA არის გასაღების გენერირების და დაშიფვრის ერთერთი ცნობილი ალგორითმი. სადაც როგორც Diffie-hellman-ის გასაღების მომოცვლის და დაშიფვრის ალგორთმშიც პრინციპულად მსგავსი მარტივ რიცხვებზე და დისკრეტულ ლოგარითმული განტოლების გამოყენებაზე დაიყვანება.

**ალგორითმის მახასითებლები და შეზღუდვები :**   
მხოლოდ ერთ მხარეს შეუძლია ინფორმაციის გაშიფვრა, რომელიც ახდენს გასაღების გენერირებას. მეორე მხარე მხოლოდ ინფორმაციას შიფრავს მიწოდებული გასაღებით და უგზავნის გასაღების შემქმნელს.

ინფორმაციის ზომა შემოსაზღვრულია მარტივი რიცხვის ზომით.(მაგალითში მივანიშნებ შეზღუდვის ადგილს.)

გასაღები გადაიცემა წყვილის სახით.

ამ ასიმეტრიულ მეთოდში ხდება გასაღების გამოთვლა, რის შედეგადაც ვიღებთ ორ Public და Private გასაღებებს რომლებიც არიან წყვილი რიცხვები.

ღია გასაღები იგზავნება, პირადი გასაღები მუდმივად შემქმნელთან რჩევა რითის შემდეგში მოახდნს დეშიფრაციას.

**ალგორითმის დასაწყისი:**

პირველ რიგში, პირველი პიროვნება ახდენს ორი მარტივი რიცხვის p და q-ს გენერირებას.

p\*q = N ამის შემდეგ მოხდება ამ ორი რიცხვის გადამრავლება.

აღსანიშნავია სწორედ ეს ფაქტი, რომ იარსებებს მხოლოდ ერთადერთი მარტივ რიცხვთა წყვილი რომელიც ნამრავლში მოგვცემს N-ს. და რაც უფრო დიდია p და q მით უფრო დიდია N და დღევანდელი კომპიუტერების მონაცემებით არარეალურად დიდი დრო სჭირდება რომ N რიცხვი დაიშალოს ორი მარტივ მამრავლად. ანუ N დან აღდგეს p და q.

ამის შემდეგ ვითვლის

ꝕ(N) ფუნქციას. რომელიც იდეურად წარმოადგენს სიმრავლის ზომას p და q რიცხვების მიმართ ურთერთ მარტივი რიცხვებისა.

შესაბამისად ჩვენ უნდა დავადგინოთ p-ს და q-ს მიმართ ურთერთმარტივ რიცხვთა რაოდენობა და გადავამრავლოთ.

იქიდან გამომდინარე რომ ჩავიფიქრეთ მარტივი რიცხვები(განმარტების თანახმად იყოფიან მხოლოდ თავის თავზე და 1 ზე) შესაბამისად მათზე ნაკლები ყველა რიცხვი მათ მიმართ ურთიერთმარტივია.

შესაბამისად სიმრავლის ზომაა p-1 და q – 1

შესაბამისად ꝕ(N) = (p-1)\*(q-1)

ეს რიცხვები დაგვეხმარება ღია გასაღების შერჩევაში.

კერძოდ არსებობს ორი პირობა.

ღია გასაღები უნდა იყოს ꝕ(N) ზე ნაკლები

Kღია < ꝕ(N) და უსგ(kღია, ꝕ(N)) = 1

ანუ ისინი არიან ურთიერთ მარტივი რიცხვები(გამნარტება ზემოთ მიწერია)

ღია გასაღების აჩევის შემდეგ ვაგენერირებთ დახურულ გასაღებს რომელიც უნდა იყოს შემდეგი განტოლების ამონახსნი

kღია \* kდახურული = 1 mod(ꝕ(N))

მას შემდეგ რაც მოვახდინეთ ზემოთ ხსენებული ალგორითმის რეალიზება.

ჩვენი გასაღებები მიიღებენ შემდეგი წყვილების სახეს.

(kღია, N) და (kდახურული, N)

დახურულის წყვილი რჩება პირველთან ხოლო მეორე წყვილი იგზავნება ღია სივრცეში რასაც ყველა დაინტერესებული ადამიანი დაშვებისამებრ ნახავს.

დაშიფვრა:

მეორე პირს ააქვს უკვე კოდირებული გამზადებული ინფორმაცია რომელზეც შეგვიძლია ოპერაციები ჩავატაროთ.

დავარქვათ მას m;(ასევე მას აქვს მიღებული წყვილი (kღია, N))

ის ადგენს ლოგარითმულ ფუნქციას:

mkღია mod(N) = m’

იქიდან გამომდინარე რომ ნაში N ზე გვაძლევს რიცხვებს სიმრავლიდან

[0, N) მას მხოლოდ შეუძლია დაშიფროს N ზე პატარა რიცხვები რადგან დეშიფრაციისას შებრუნებული პროცესისას N ზე დიდ რიცხვს ვერ მივიღებთ.

M’ < N

ამის შემდეგ m’ გადაეგზავნება ისევ უკან და ხდება შეპრუნებული მეწყვილის გამოყენებით დეშიფრაცია

(m’)kდახურული mod(N) P.S პირველ მხარეშ შენახული აქვს შემდეგი წყვილი (kდახურული, N)

Mkღია \* kდახურული  mod(N) = M

მოხდა დეშიფრაცია

საიმედოობა:

დამოკიდებულია მარტივი რიცხვების ზომაზე

RSA გამოავეყნა ჩემპიონატი თ ვინ შეძლებდა ორი მამრავლის პოვნას მიცემუი დიდი 120 თანრიგზე მაღლა თ ზუსტად მახსოვს ვერცერთი რიცხვი ვერაა დაშლილი.

შედარებით მოუხერხებელი ალგორითმია რადგან მხოლოდ ერთ მხარეს შეუძლია ინფორმაციის განშიფვრა.

მაგონებს პიროვნების და სერვერის ურთიერთობას.

მაგალითი იხილეთ EXEL ის ფაილში.

ფერთა კოდები

ფერთა კოდები ერთგვარი ვიზუალიზაციაა diffie-hellman- ის გასაღების მიმოცვლის ალგორითმის

იდეა არის მარტივი

გვაქვს სამი ფერი, ერთი საჯარო ორი პირადი

დავარქვათ a(პირადი), b(საჯარო) c(პირადი)

თუ შევურევთ

ab მივიღებთ რაღაც p ფერს

თუ შევურევთ

bc მივიღებთ რაღაც q ფერს

თუ ეს ორი მხარე გაცვლიან ამ ფერებს და კიდევ შიგნით დაამატებენ თავიანთ ფერს

მიიღებენ

abc & bca ფერებს

შესაბამისად ჩვენი ბუნებრივი ინსტინქტი და ფერთა კოდები გვკარნახობს რომ შედეგი უდნა იყოს ერთი და იგივე იმის და მიუხედავად რომ რეალურად მომხდარა არც a და არც c ფერის გამჟღავნება

ხოლო მიღებულ ფერს ვერავინ ვერ გაიგებს იმ დაშვებით რომ ორი ფერის განცალკევება რეალობაში რთულია.

(კომპიუტერით რეალურად თ მართლა კოდს ავიღებთ და შეკრების ოპერაციას გავაკეთებთ გამოკლევა მარტივია, თუ დაგაინტერესებთ ეს ნაწილი ლექციაზე გეტყვით რატომ ჩემ აზრს.)

ეს ერთგვარი diffie hell მანის ვიზუალიზაციაა რადგან გასაღების მიმოცვლის პრინცილპი იგივეა ხოლო ფერებს რიცხვები და ოპერაციას ლოგარითმული განტოლება ცვლის

ფერთა კოდების ვიზუალიზაცია იხილეთ EXEl ში.

რაღათქმაუნდა რეალურად ეს მეთოდი არ გამოიყენება. იმ მიზეზის გამო რომ რეალურად ერთ ფერს ერთი კოდი შეესაბამება (ხშირად ჩაწერილი ან RGB ან HEX კოდად)

შესაბამისად თ ერთი HEX კოდი ვიცით და ვიცით ჯამი. მარტივია ფარული HEX კოდის გამოცნობა გამოკლების ოპერაციით. რადგან ისინი მხოლოდ რიცხვებია და შეკრების იპერაცია ლოგარითმულ ახარისხებას ვერ ანაცვლებს.