Zavod: ZEMRIS
Kolegij: Operacijski sustavi 2
Student: **Dinko Korunić, 0036355514**
Voditelji: prof. dr. sc. Leo Budin, dr. sc. Marin Golub

Diffie-Hellman razmjena ključeva

(teorija i praktična implementacija)

1. Općenito

Diffie-Hellman protokol dogovora ključeva (engl. key agreement protocol ili exponential key agreement) je razvijen od strane dvojice kriptografa Diffie i Hellman 1976. godine i objavljen u znanstvenom radu "New Directions in Cryptography". Dotični protokol omogućava dvojici korisnika da izmijene tajni ključ preko nesigurnog medija bez ikakvih prethodno razmijenjenih tajnih ključeva. Sam protokol se može naći detaljno specificiran i u RFC 2631.

2. Algoritamski opis

Sam protokol ima dva inicijalna parametra: p i g, koji se obično zadaju unaprijed ili dohvaćaju sa kakvog poslužitelja. Oba su javna i mogu biti korištena od strane svih korisnika u sistemu. Parametar p je prim. broj (obično se naziva modulus), dok je parametar g (baza ili generator) cijeli broj manji od p sa slijedećim svojstvom: za svaki broj n izmedju 1 i p-1 uključno, postoji barem jedan broj n takav da n = g ^ k mod p. Ako pretpostavimo da su Alice i Bob žele razmijeniti određeni tajni ključ koristeći DH protokol, oni će činiti slijedeće:

1. Alice stvara slučajni (naravno, pseudoslučajni) vlastiti broj a, dok Bob stvara vlastiti slučajni vlastiti broj b. Oba broja (a i b) moraju biti cijeli brojevi između 1 i p - 2 uključno.
2. Slijedi izračunavanje pojedinačnih javnih ključeva koristeći parametre p i g te vlastite brojeve: Alice ima javni ključ g ^ a mod p, dok je Bobov javni ključ g ^ b mod p.
3. Razmijene dobivene javne ključeve
4. Alice izračunava gab = gb ^ a mod p, dok Bob izračunava gba = ga ^ b mod p. Budući da gab = gba = k. Alice i Bob sada imaju zajednički tajni ključ k.

3. Karakteristike, nedostatci, itd.

Nažalost, Diffie-Hellman razmjena ključeva je očito ranjiva na tzv. man-in-the-middle napad. U našem slučaju, napadač Carol može presresti javni ključ od Alice i poslati svoj vlastiti ključ Bobu. Naravno, kad Bob pošalje svoj javni ključ, Carol ga zamijeni sa svojim i pošalje Alice. Carol i Alice se uslijed toga dogovore oko jednog dijeljenog ključa, te Carol i Bob oko drugog. Carol nakon toga jednostavno nastavi slušati promet i dekriptirati bilo koju poruku između Alice i Bob, čitati i modificirati (re-enkriptira poruke sa odgovarajućim ključem) i emitirati. Glavni problem je da DH razmjena ključeva ne autentificira sudionike - što bi se moglo riješiti upotrebom digitalnih potpisa i podvarijanti samog protokola.
Spomenimo radi zanimljivosti podvarijantu DH - tzv. STS protokol (Station-to-Station) koji je razvijen od Diffie, van Oorschot i Wiener u 1992. godini da bi se riješili upravo

navedeni problemi. Rješenje je u tome da se dvoje partnera autentificiraju jedan drugome koristeći digitalne potpise i certifikate sa javnim ključevima: prije samog izvršenja protokola, Alice i Bob zahtijevaju par ključeva (javni/tajni) i certifikat za javni ključ. Tijekom protokola, Alice izračunava potpis na porukama koji uključuje i javnu vrijednost g ^ a mod p, a to čini i Bob. Bez obzira ako Carol presreće poruke između Alice i Boba, ona ne može lažirati potpise bez znanja pojedinačnih tajnih ključeva Alice i Boba.

4. Implementacija

Opet, i ovdje je odabir bio jezik C radi jednostavnosti i portabilnosti - s time je zbog korištenja vrlo velikih prirodnih brojeva (puno većih od unsigned long) logičan odabir pao na GNU MP biblioteku i korištenje njenih velikih prirodnih brojeva tipa mpz_t i pripadnih im funkcija. Sama implementacija se striktno drži originalnog RFC 2631, te koristi hardkodirani "0x2344010023498019329fafef32324ff65f22398490001291209291" za bazu i hardkodirani prim. broj "0x0113f7fefd73f9d36ff3fefd2e34102202d01022d0c4410113f7fefc1400000014000003e3f3fefd43fbfefc12cc102062cc102063f7fefecc300d0013" za modulus. U našem programu nije poanta bila pronalaženje takvog velikog prim. broja, stoga je ovaj primjerak izvađen iz odgovarajuće literature, no provodi se bazični test dostupan iz GMP biblioteke da li je uopće riječ o prim broju.

5. Literatura

- "Ritter's Crypto Glossary and Dictionary of Technical Cryptography"
- RFC2631 (Diffie-Hellman Key Agreement Method, June 1999, E. Rescorla)
- Diffie-Hellman Key Exchange (Alan Westrope, 1998)
- izvorni programi: Diffie-Hellman key exchange, using CypherMath8 (Win32 C); Diffie-Hellman key exchange od Jeff Williams (x86 Asm + C)

6. Primjer funkcioniranja programa

```
Postavio bazu =
145074429440517333485794650037592504365688360060809918166021
29041
Postavio modulus =
336525389506639635903693828607929703897888785182821869962801
754906120967231587216282546832125696050274514396819232504456
00913004144551467268833347
Generiram Bobov javni kljuc
Izracunao privatni broj =
751270037095217092166390382159666694941500573959844230084640
585074648085608172195965364501599628809786638303794870180158
74013531309816895835685 5
Izracunao javni kljuc =
121292053024881556038131855350203629936444345516607482230 43
```

```
4027962282555051600048259272916468431690796481673224146422654909070387838989919455623
Generiram Alicin javni kljuc
Izracunao privatni broj =
2247751356492684550177151899019534929085330705579951882949579944881578589760435656490799265324360310306203872803780180102116119104895517371266666
Izracunao javni kljuc =
1114108630708350754302558585030683261490597393207948188816631594351954092002602453446765540532216370743856517698434304318731907311302652803907915
Generiram Alicin tajni kljuc
Izracunao privatni kljuc =
2287596037469506467366367789099207571814231334562299844273784187343656711297147205858738311981067585521091855728283941783316303898952473149231518
Generiram Bobov tajni kljuc
Izracunao privatni kljuc =
2287596037469506467366367789099207571814231334562299844273784187343656711297147205858738311981067585521091855728283941783316303898952473149231518
Bobov i Alicin kljuc su *isti*!
```