SIGURNOST RAČUNALNIH SUSTAVA

Osnove kriptografije
Sigurnosni protokoli
Potpora sigurnosti u operacijskim sustavima
Infrastruktura javnih ključeva

http://sigurnost.zemris.fer.hr 2002-2006-danas

1

Informacijski sustavi po važnosti

- vojni informacijski sustavi,
- bankovni informacijski sustavi,
- zdravstveni i bolnički informacijski sustavi,
- informacijski sustavi državnih institucija,
- informacijski sustavi osiguravajućih društava,
- poslovni informacijski sustavi gospodarskih subjekata.

Podjela sigurnosnih mehanizama

- zaštita od vanjskih utjecaja,
- zaštita ostvarena sučeljem prema korisniku,
- unutarnji zaštitni mehanizmi,
- komunikacijski zaštitni mehanizmi.

2

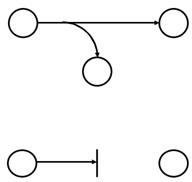
Osnovni pojmovi

identifikacija = predstavljanje
autentifikacija = identifikacija + verifikacija
autorizacija = autentifikacija + provjera ovlasti tj. provjera prava pristupa

3

Prijetnje i napadi na sigurnost

1. prisluškivanje
2. prekidanje
3. lažno predstavljanje
4. ponovno odašiljanje snimljenih starih paketa
5. modifikacija paketa
6. poricanje



4

Sigurnosni zahtjevi

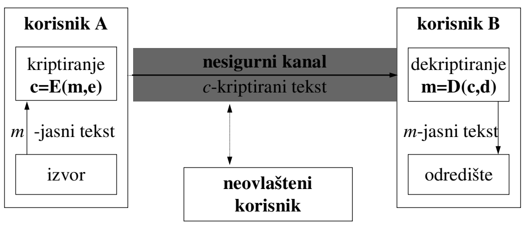
1. autentičnost
2. integritet
   (podaci su potpuni i nepromijenjeni)
3. tajnost
4. neporecivost

5. kontrola pristupa
6. raspoloživost

Napadi:
1. prisluškivanje
2. prekidanje
3. lažno predstavljanje
4. ponovno odašiljanje snimljenih starih paketa
5. modifikacija paketa
6. poricanje

5

Osnove kriptografije



6

Tipovi kriptografskih algoritama

- simetrični

e = d = K (simetrični, sjednički ili tajni ključ) (DES, 3-DES, IDEA, AES, Blowfish, RC6, GOST, Mars, Serpent, Loki, CAST...)

- asimetrični *(algoritmi s javnim ključem)*

e ≠ d (***P*** - javni i ***S*** - privatni ključ) (RSA, Diffie-Hellman, RPK, ECES, ElGamal, LUC, Blum Goldwaser...)

7

Simetrični kriptosustavi

- Kerckhoffov princip:
  Kriptosustav mora biti siguran i onda kada su sve informacije o kriptosustavu javno poznate, osim tajnog ključa.
- temelje se na jednostavnoj logičkoj operaciji *isključivo ILI (XOR)*:

$$C = P \oplus K \quad P = C \oplus K$$

$$\boldsymbol{P = (P \oplus K) \oplus K}$$

8

Klasična kriptografija

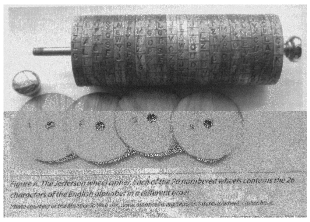
(time se na ovom predmetu nećemo baviti!)

- supstitucijske šifre: najpoznatija Cezarova šifra (pomak za 3 slova) – monoalfabetska šifra
- Vigenereova šifra: polialfabetska šifra, svako slovo se preslikava u jedno od *m* mogućih slova u ovisnosti o svom položaju
- Playfairova šifra: šifriraju se parovi slova
- Hillova šifra: grupira tekst po *m* slova
- ***ONE TIME PAD – jednokratna bilježnica***
- transpozicijske šifre (zamjena položaja slova)

9

Klasična kriptografija

Naprave za šifriranje



10

DES (*Data Encryption Standard*)

- 1977. razvijen u IBM-u
- najpoznatiji simetrični algoritam i još uvijek se koristi unatoč njegovoj *nesigurnosti*
- veličina ključa: 56 – bita
- smatra se nesigurnim: 1998: "DES Challenge II" ostvareno računalo za $250.000 koje za manje od 3 dana razbija DES poruku (nagrada je bila $10.000)
- mala veličina ključa je najveći nedostatak koji se otklanja višestrukim kriptiranjem (*triple* DES s ključem veličine 112 ili 168 bita)

11

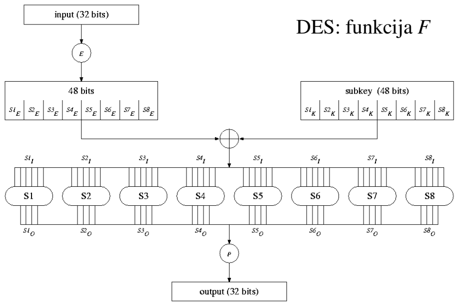
DES – postupak kriptiranja

- kriptiraju se blokove duljine 64 bita (8 bajtova)
- iz ***K*** (56 bitova) se određuje 16 podključeva ***Ki*** duljine 48 bita
- Postupak kriptiranja poruke P duljine 8 bajtova:
  - *L0 R0 = IP (P).*
- 16 koraka: *Li = Ri-1*

  Ri = Li-1 ⊕ f (Ri-1, Ki),
- *f(Ri-1, Ki)* obavlja "preslagivanje" bitova u *Ri-1* ovisno o parametru *Ki*
- na kraju se obavlja inverzna permutacija od *IP*

  $C = IP^{-1}$ (R16 L16).

12



DES: funkcija *F*

Supstitucijske tablice

- Ulaz u S tablicu je veličine 6 bita, a izlaz 4 bita.
- Supstitucijska tablica *S1*:

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

- Prvi i zadnji bit svakog dijela ulaza predstavlja adresu retka.
- Srednja četiri bita određuju adresu stupca u tablici selekcije.
- Od ulaznih 48 bita dobivamo 32 bita nakon supstitucije.

- simulator DES kriptosustava:
  http://os2.zemris.fer.hr/algoritmi/simetricni/1999_basic/desvis.exe

Pregled simetričnih algoritama kriptiranja

DES
Lucifer
DESX
3-DES
Blowfish
FEAL
ICE
IDEA
Khufu
NewDES
RC2
RC5
SHARK

AES (15.3.2000.)
finalisti, ožujak 1999:
MARS
RC6
Serpent
Twofish
15 odabranih 20.8.1998.
LOKI97
FROG
DEAL, HPC
CAST 256
E2, DFC
CRYPTON
MAGENTA
SAFER+

jednostavni
3-Way
Enigma
Solitaire
TEA

kriptiranje toka podataka
A5
Pike
RC4
SEAL
SOBER
WAKE

višenamjenski
Panama
Sapphire

ostali
CAST
GOST
Misty
Square
Turtle

Utrostručeni DES, 3DES

3DES (P, K1, K2, K3) =
DES (DES^{-1}(DES (P, K1), K2), K3)

3DES^{-1} (C, K1, K2, K3) =
DES $^{-1}$(DES (DES $^{-1}$(C, K3), K2), K1)

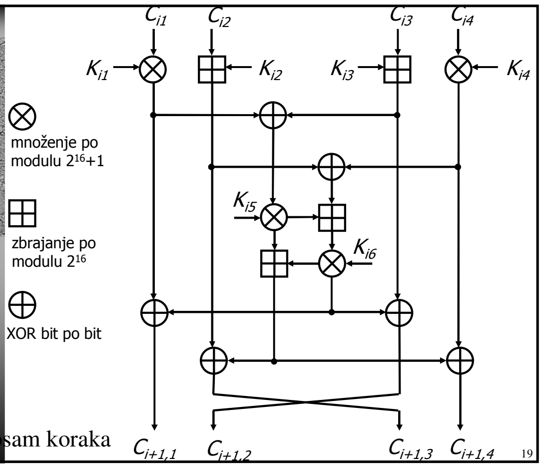
Izbijeljeni DES, DESX

DESX (P, K1, K2, K3) =
DES (P ⊕K2, K1) ⊕K3

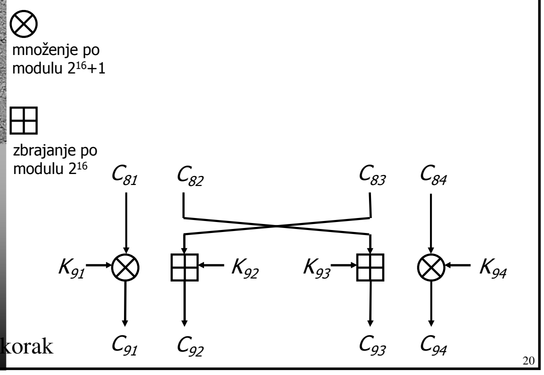
DESX^{-1}(C, K1, K2, K3) =
DES $^{-1}$ (C ⊕K3, K1) ⊕K2

IDEA (*Internatonal Data Encription Algorithm*)

- dovršen 1992., siguran
- ključ je duljine 128 bita
- blokovi duljine 64 bita dijele se na 4 podbloka (16 b)
- postupak kriptiranja se provodi u 9 koraka
- u svakom od prvih 8 koraka sudjeluju:
  - 4 podbloka i 6 podključeva duljine 16 bita
- u devetom koraku se koriste 4 podključa
- dakle, iz ključa ***K*** je potrebno generirati 8x6+4=52 podključeva



IDEA
prvih osam koraka



IDEA
deveti korak

Napredni kriptosustav (AES)

- natječaj za napredni kriptosustav (AES – *Advanced Encryption Standard*) je raspisao NIST (*National Institute of Standards and Technology*) 12.9.1997. godine
- 3DES je proglašen kao privremeni standard
- na natječaj su se mogli prijaviti samo algoritmi sa sljedećim svojstvima:
  - simetrični blok algoritmi s javnim izvornim tekstom programa
  - blok podataka koji se kriptira je minimalne veličine 128 bita
  - veličina ključa od 128, 192 i 256 bita

- pobjednik: Rijndael (izgovara se *"Rain Doll"*)
  - Rijndael - 86 glasova
  - Serpent - 59
  - Twofish - 31
  - RC6 - 23
  - MARS - 13

Konačno polje *GF(2⁸)*

- elementi polja su polinomi oblika:
  $a_7x^7 + a_6x^6 + ... + a_1x + a_0$, a_i {0, 1}
- svaki bajt $a_7a_6a_5a_4a_3a_2a_1a_0$ (niz od 8 bitova) je predstavljen odgovarajućim polinomom

- *zbrajanje* - isključivo ILI
- *množenje* - binarno množenje polinoma modulo fiksni ireducibilni polinom
  $g(x) = x^8 + x^4 + x^3 + x + 1 \equiv 11B_H$

Primjer množenja u GF(2⁸)

$$57_H \bullet 83_H = C1_H$$

$$57_H = 01010111_2 \equiv x^6 + x^4 + x^2 + x + 1$$
$$83_H = 10000011_2 \equiv x^7 + x + 1$$

$$\begin{array}{l} \underline{(x^6 + x^4 + x^2 + x + 1) \bullet (x^7 + x + 1) =} \\ x^{13} + x^{11} + x^9 + x^8 + x^7 \\ \qquad\qquad\qquad x^7 + x^5 + x^3 + x^2 + x \\ \oplus \qquad\qquad\qquad \underline{x^6 + x^4 + x^2 + x + 1} \\ x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \end{array}$$

ostatak pri dijeljenju s fiksnim ireducibilnim polinomom *g(x)*:

$$\begin{array}{l} (x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1) : (x^8+x^4+x^3+x+1) = x^5+x^3 \\ \underline{x^{13} + x^9+x^8+x^6+x^5 \quad -} \\ \quad x^{11} + x^4+x^3+1 \\ \quad \underline{x^{11} + x^7+x^6 + x^4+x^3 \quad -} \\ \qquad x^7+x^6 + 1 \equiv 11000001_2 \equiv C1_H \end{array}$$

Množenje u *GF(2⁸)* uz pomoć funkcije *xputa()*

$\mathbf{57_H \bullet 83_H} = 57_H \bullet (01_H + 02_H + 80_H)$

$= 57_H + AE_H + 38_H \qquad \mathbf{= C1_H}$

- množenje s $x \equiv 02_H$ je zapravo posmak za jedan bit ulijevo:

  $\mathbf{57_H \bullet 02_H = 10101110_2 \equiv AE_H}$

- ukoliko dođe do preliva, tada treba oduzeti $g(x) \equiv 11B_H$:

$$57_H \bullet 04_H = AE_H \bullet 02_H = 101011100_2$$
$$\oplus\ \underline{100011011_2}$$
$$01000111 \equiv 47_H$$

$57_H \bullet 08_H = 47_H \bullet 02_H = 8E_H$
$57_H \bullet 10_H = 8E_H \bullet 02_H = 07_H$
$57_H \bullet 20_H = 07_H \bullet 02_H = 0E_H$
$57_H \bullet 40_H = 0E_H \bullet 02_H = 1C_H$
$\mathbf{57_H \bullet 80_H =}\ 1C_H \bullet 02_H\ \mathbf{= 38_H}$

25

Množenje s polinomima stupnja manjeg od 4

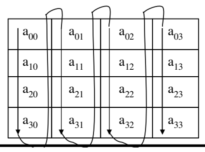
- množenje je definirano kao binarno množenje polinoma modulo $x^4 + 1$:
  - <u>u 1. koraku</u> se dobiva polinom šestog stupnja: $c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0$
  - <u>u 2. koraku</u> se dobiveni rezultat reducira po modulu polinoma x^4+1
- $d(x)=a(x) \bullet b(x)$ može se zapisati i u matričnom obliku:

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix}$$

26

AES blok

- moguće veličine ključa: 128, 192 ili 256 bita
- veličina bloka: 128 bita (AES, izvorni algoritam Rijndael dopušta veličine bloka od 128, 192 ili 256 bita nezavisno od veličine ključa)
- blok koji se kriptira smješten je u pravokutni niz bajtova u četiri retka, dok broj stupaca ovisi o njegovoj duljini: *Nb* = 4, 6, ili 8
- na isti način se tretira i ključ: *Nk* - broj stupaca bloka ključa
- 128 bitni blok izgleda ovako:



- redoslijed punjenja bloka

27

AES – broj koraka

- kriptiranje i dekriptiranje se obavlja u koracima
- broj koraka ovisi o veličini bloka podataka i veličini bloka ključa:

Rijndael

N_r	N_b=4	N_b=6	N_b=8
N_k=4	10	12	14
N_k=6	12	12	14
N_k=8	14	14	14

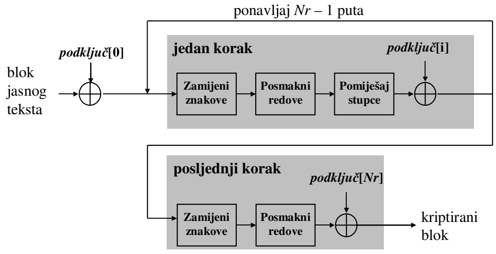
AES

N_r	N_b=4
N_k=4	10
N_k=6	12
N_k=8	14

28

AES

- podključevi su po veličini jednaki veličini bloka podataka i dobivaju se iz izvornog ključa
- svi podključevi čine jedan *prošireni ključ* koji ukupno ima (*Nr* + 1)·*Nb* bitova



29

Funkcije koje koristi AES

zamijeni znakove

znak = Sbox[*znak*]

dodaj podključ

blok = *blok* ⊕ *podključ*[*i*]

posmakni redove

- rotira (kružno posmiče) znakove ulijevo i to u drugom, trećem i četvrtom redu bloka (*C1*, *C2* i *C3*) za unaprijed poznati broj mjesta koji ovisi o *Nb*
- prvi red (C_0) se ne posmiče

N_b	C_1	C_2	C_3
4	**1**	**2**	**3**
6	**1**	**2**	**3**
8	**1**	**3**	**4**

30

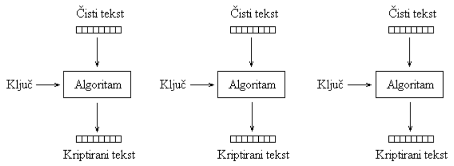
Funkcije koje koristi AES

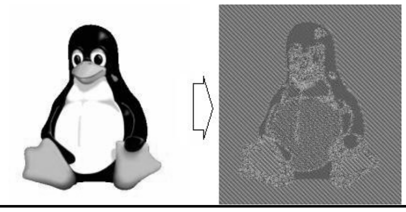
pomiješaj stupce

- množi se stupac po stupac bloka (tako da se svaki stupac promatra kao četveročlani polinom) s fiksnim polinomom $a(x) = 03_H x^3 + 01_H x^2 + 01_H x + 02_H$ modulo x^4+1
- odnosno, za svaki stupac bloka računa se stupac novog stanja:

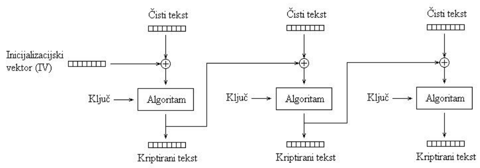
$$\begin{bmatrix} s_{0c}' \\ s_{1c}' \\ s_{2c}' \\ s_{3c}' \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0c} \\ s_{1c} \\ s_{2c} \\ s_{3c} \end{bmatrix}$$

Načini kriptiranja: *ECB - Electronic Codebook*

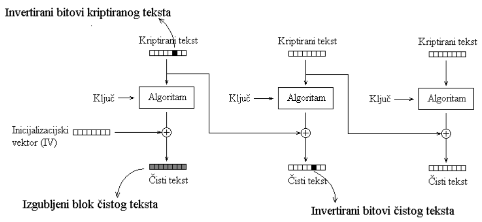




Načini kriptiranja: *Cipher Block Chaining (CBC)*

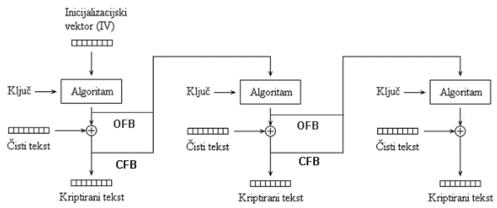


Načini kriptiranja: *Cipher Block Chaining (CBC)*



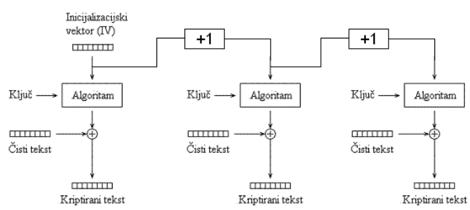
Cipher Feedback (CFB) i Output Feedback (OFB)

- koriste se za kriptiranje toka podataka



Counter (CTR)

- *IV* je rastuća vrijednost ili neka druga funkcija



Asimetrični kriptosustavi

ili sustavi s javnim ključem

Neke činjenice i algoritmi iz teorije brojeva

Djeljivost

Broj a djeljiv je s brojem d kada je a višekratnik od d.

$d \mid a$ $\quad$ d dijeli a, d je djelitelj od a;
$a = k \times d$ $\quad$ a je višekratnik od d.

Najmanji djelitelj od a je $d = 1$, a najveći djelitelj je $d = a$.

To su trivijalni djelitelji. Netrivijalni djelitelji zovu se faktori.

Primjer 11.1.

Broj $a = 24$ ima sljedeće djelitelje: $1, 2, 3, 4, 6, 8, 12, 24$.

Trivijalni djelitelji su 1 i 24, a faktori $2, 3, 4, 6, 8, 12$. □

Prosti ili prim brojevi

$a > 1$ koji nema faktora (ima samo djelitelje 1 i a)

Teorem dijeljenja

$\forall$ cijeli broj a i bilo koji cijeli broj $n > 0$ $\exists$ q i r:

- kvocijent, količnik q i
- reziduum, ostatak r (uz $0 \le r < n$),

tako da vrijedi: $\quad a = q \times n + r$.

$q = \lfloor a/n \rfloor$ $\qquad$ $r = a \bmod n$

$a = \lfloor a/n \rfloor \times n + (a \bmod n)$

$\Rightarrow a \bmod n = a - \lfloor a/n \rfloor \times n$

Ekvivalentnost po modulu, kongruentnost

$$a \equiv b \pmod n \quad \Rightarrow \quad a \bmod n = b \bmod n.$$

a i b kongruentni po modulu n, odnosno je ekvivalentan broju b po modulu n.

Relativno prosti brojevi

$$nzd(a,b) = 1$$

Brojevi a i b su relativno prosti ako je najveći zajednički djelitelj brojeva a i b jednak 1, tj. brojevi a i b nemaju zajedničkih faktora.

Eulerova phi ili totient funkcija

$Z_n = \{0, 1, 2, \ldots, n - 1\}$ - prsten u kojem su definirane operacije zbrajanja, oduzimanja i množenja po modulu n

Z_n^* podskup koji se sastoji od elemenata skupa Z_n koju su relativno prosti u odnosu na n:

$$Z_n^* = \{ a \in Z_n,\ nzd(a,n) = 1 \}$$

$\boxed{\varphi(n) = |Z_n^*| \text{ - Eulerova } \textit{phi ili totient funkcija}}$

Ako je $n = p$ prosti broj onda je $\varphi(p) = p - 1$.

Ako n nije prost, tj. $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ onda je

$$\varphi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_k}\right).$$

Ako je $n = p \times q$, gdje su p i q prosti brojevi onda je

$$\varphi(n) = n\left(1 - \frac{1}{p}\right)\left(1 - \frac{1}{q}\right) = (p-1)(q-1).$$

Dokaz za $n = p$ je trivijalan:

$Z_p = \{0, 1, 2, \ldots, p - 1\} \Rightarrow Z_p^* = \{1, 2, \ldots, p - 1\}$ □

Dokaz za $n = p \times q$

$Z_n = \{0, 1, 2, \ldots, p \times q - 1\}$ ima ukupno $p \times q$ elemenata:

$\{0\}$	s 1 elementom,
$\{p, 2\times p, \ldots, (q-1)\times p\}$	s $q-1$ elemenata,
$\{q, 2\times q, \ldots, (p-1)\times q\}$	s $p-1$ elemenata,
Z_n^*	s $\lvert Z_n^* \rvert$ elemenata.

$p \times q = 1 + (q-1) + (p-1) + |Z_n^*| \Rightarrow |Z_n^*| = (p-1)(q-1)$ □

Primjer 11.2.

Neka je $n = 15 = 3 \times 5 \quad \Rightarrow p = 3$ i $q = 5$.

$Z_{15} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}$

$Z_{15}^* = \{1, 2, 4, 7, 8, 11, 13, 14\}$

$|Z_{15}^*| = 8$, jer je $\varphi(15) = (5-1)(3-1) = 8$.

□

Primjer 11.3. *Modularno potenciranje*

Potenciranje s velikim eksponentima prikladno je obaviti uzastopnim kvadriranjem.

Želimo izračunati $d = b^a \bmod n$.

Neka je a_m , a_{m-1} , a_{m-2} ,..., a_1 , a_0 binarni prikaz od a .

```
d = 1;
i = m;
dok je (i >= 0) {
     d = (d * d) mod n;
     ako je (a[i] == 1) {
          d = (d*b) mod n;
     }
     i --;
}
```

43

Primjer 11.4.

Neka su:

$a = 635 \Rightarrow a = 1001111011_{(2)}$

$n = 734$

$b = 5$

```
d = 1;
i = m;
dok je (i >= 0) {
     d = (d * d) mod n;
     ako je (a[i] == 1) {
          d = (d*b) mod n;
     }
     i --;
}
```

Postupak izračunavanja $b^a \bmod n$:

i	9	8	7	6	5	4	3	2	1	0
a[i]	1	0	0	1	1	1	1	0	1	1
d	5	25	625	685	261	29	535	699	253	<u>21</u>

44

Eulerov teorem

$\forall$ prirodni broj $n > 1$ vrijedi

$$a^{\varphi(n)} \equiv 1 \pmod{n} \quad \text{za sve } a \in Z_n^*$$

45

Fermatov teorem

Posebno za proste brojeve p vrijedi

$$a^{p-1} \equiv 1 \pmod{p} \quad \text{za sve } a \in Z_p^*$$

S obzirom da je $Z_p^* = \{ 1, 2, ..., p-1 \}$, Fermatov teorem vrijedi za sve brojeve iz Z_p *s* izuzetkom broja *0 .*

Posljedica:

$$a^{p} \equiv a \pmod{p} \quad \text{za sve } a \in Z_p^*$$

Postoje i neki složeni brojevi (Carmichelovi brojevi) kod kojih za mnoge različite vrijednosti od a vrijedi gornji izraz, primjerice 561, 1105.

46

Primjer 11.5. $\quad a^{p-1} \equiv 1 \pmod{p}$

Neka je $p = 11$. $\quad a^{p} \equiv a \pmod{p}$

i	0	1	2	3	4	5	6	7	8	9	10	11
$2^i \bmod 11$	1	2	4	8	5	10	9	7	3	6	**1**	**2**
$5^i \bmod 11$	1	5	3	4	9	1	5	3	4	9	**1**	**5**
$7^i \bmod 11$	1	7	5	2	3	10	4	6	9	8	**1**	**7**

Kod nekih nizova potencija ne pojavljuju svi elementi Z_p^* .

Za $a = 5$ potencije generiraju podskup $\{1, 3, 4, 5, 9\}$.

Taj podskup ima 5 članova: $\quad \mathbf{ord_{11}(5) = 5}$. $\quad \square$

Osnovni korijen ili ***generator*** od Z_n^* :

$$ord_n(a) = |Z_n^*|$$

47

Ako Z_n^* ima osnovni korijen onda Z_n^* čini cikličku grupu.

Dokazano je da je Z_n^* ciklička grupa za:

$n = 2, 4, p^e, 2p^e$,

gdje su p prosti brojevi i e prirodni brojevi.

Diskretni logaritam ili indeks

Neka je a osnovni korijen od Z_n^* i $b \in Z_n^*$. Broj x je ***diskretni logaritam*** ili ***indeks*** broja $b \pmod{n}$ u odnosu na bazu a, ako vrijedi:

$$a^x \equiv b \pmod{n}$$

ili drugačije zapisano:

$$x = ind_{n,a}(b).$$

48

Primjena kineskog teorema ostataka

Neka je $n = n_1 \times n_2 \times \dots \times n_{1k}$, gdje su svi parovi faktora relativno prosti. Teorem kaže da je struktura Z_n identična kartezijevom produktu $Z_{n_1} \times Z_{n_2} \times \dots \times Z_{n_k}$.

Za proste brojeve p i q i za bilo koja dva cijela broja x i a vrijedi

$\boldsymbol{x \equiv a \ (mod\ p)}$ i

$\boldsymbol{x \equiv a \ (mod\ q)}$

onda i samo onda ako je

$\boldsymbol{x \equiv a \ (mod\ n)}$.

RSA (*Ron Rivest, Adi Shamir i Len Adleman*)

- zasniva se na teško rješivom matematičkom problemu faktoriziranja velikih brojeva ($>10^{150}$)
- 17. 3.1999. godine je 155 znamenkasti broj (512 binarnih znamenaka). Postupak faktorizacije trajao je nešto više od 5 mj. paralelno na 292 računala, uz 4 mjeseci pripreme na CRAY superračunalima.
- 2.11.2005. probijen RSA-640
- minimalna duljna ključa je 1024 bita (broj sa 309 znamenki), ključ od 2048 bita odgovara broju sa 617 znamenki.
- (U svemiru ima manje od 10^{80} atoma.)

Međutim:

"If all the personal computers in the world - 260 million - were put to work on a single PGP-encrypted message, it would still take an estimated 12 million times the age of the universe, on average, to break a single message."

William Crowell, Deputy Director, National Security Agency, March 20, 1997.

RSA

1. odabrati $p >>$ i $q >>$ ($p > 10^{100}$, $q > 10^{100}$)
2. izračunati $n = p \times q$
3. izračunati umnožak $\varphi(n) = (p - 1) \times (q - 1)$
4. odabrati $e < \varphi(n)$, tako da je $nzd\,[e, \varphi(n)] = 1$
5. izračunati broj d $< \varphi(n)$ tako da vrijedi
   $e \times d \equiv 1 \ (mod\ \varphi(n))$
   ili drugačije zapisano
   $e \times d = k \times \varphi(n) + 1$
6. $P = (e, n)$ – javni ključ (*public key*)
7. $S = (d, n)$ – privatni ključ (*private key*)

RSA

Kriptiranje: $C = RSA\,(M, S) = M^e \bmod n$, $P = (e, n)$

Dekriptiranje: $P = RSA^{-1}(C, P) = C^d \bmod n$, $S = (d, n)$

- Kriptiranje i dekriptiranje se može obaviti algoritmom modularnog potenciranja iz primjera 11.3.
- *e* je obično *mali* broj koji je zajednički grupi korisnika

Napadi na RSA

Rastaviti *n* na faktore: $n = p \times q$

- iz poznatog para (e, n) se relativno lako izračuna d

Ako je $C = M^e \bmod n$, tada je e –ti korijen iz $C \bmod n$ jednako M pa nam ni ne treba d.

- Nije poznat ni jedan uspješan napad na ovaj način osim za male brojeve *n*.

Zašto je RSA kriptosustav korektan?

$RSA^{-1}(RSA(M, S), P) = ?$ // tj., je li to jednako M?

$= (M^e \bmod n)^d \bmod n = M^{e\times d} \bmod n$.

Budući je

$e\times d = k\times \varphi(n) + 1 = k\times (p-1)(q-1) + 1,$

može se za one M koji nisu kongruentni s 0 $(\bmod p)$ uporabom Fermatova teorema pisati:

55

$$
\begin{aligned}
M^{e\times d} &\equiv M^{k\times (p-1)(q-1)+1} && (\bmod p)\\
&\equiv M\times (M^{(p-1)})^{k\times(q-1)} && (\bmod p)\\
&\equiv M\times (1)^{k\times(q-1)} && (\bmod p)\\
&\equiv M && (\bmod p)
\end{aligned}
$$

To je, također, trivijalno ispunjeno i za

$M \equiv 0 \quad (\bmod p)$.

Jednako tako vrijedi i:

$$
\begin{aligned}
M^{e\times d} &\equiv M^{k\times (p-1)(q-1)+1} && (\bmod q)\\
&\equiv M\times (M^{(p-1)})^{k\times(q-1)} && (\bmod q)\\
&\equiv M\times (1)^{k\times(q-1)} && (\bmod q)\\
&\equiv M && (\bmod q)
\end{aligned}
$$

Dakle $M^{e\times d} \equiv M \quad (\bmod p)$,
$M^{e\times d} \equiv M \quad (\bmod q)$,

vrijedi prema *kineskom teoremu ostataka* samo ako je

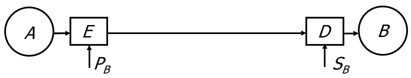
$$\boldsymbol{M^{e\times d} \equiv M \quad (\bmod n)}.$$

Prema tome *RSA* sustav je korektan. □

56

Komuniciranje uporabom kriptosustava RSA

- Branko i Ana kada žele komunicirati obznanjuju svoje javne ključeve P_A i P_B te čuvaju samo za sebe svoje privatne ključeve dekriptiranja S_A i S_B .
- Ana, koja želi poslati poruku M Branku, nekako doznaje njegov javni ključ P_B , kriptira poruku s tim ključem.
- Jedino Branko zna svoj privatni ključ dekriptiranja S_B i jedino on može dekriptirati poruku.



- Kriptiranje se obavlja tako da se razgovijetni tekst P podijeli na niz brojeva $M_i < n$ jednake bitovne duljine:

  $M = M_0\, M_1\, M_2 \ldots M_i \ldots$

57

Primjer 11.6

Odaberimo $p = 17$ i $q = 19$.

$\Rightarrow n = 17\times 19 = 323$ i $\varphi(n) = 16\times 18 = 288.$

Moramo odabrati d i e , tako da bude zadovoljeno:

$nzd[d, \varphi(n)] = 1, \quad nzd[e, \varphi(n)] = 1$ te

$e\times d = k\times \varphi(n) + 1.$

Pogledajmo sljedeće vrijednosti:

k	$k\times \varphi(n) + 1$	
0	1	
1	289	$= 17 \times 17$
2	577	*prosti broj*
3	865	$= 5\times 173$

Javni ključ kriptiranja je $P = (17, 323)$ i privatni ključ dekriptiranja $S = (17, 323)$.

58

Razgovijetni tekst možemo podijeliti na niz brojeva $M_i < n$. Za primjer neka je $M_i = \{17, 98, 62, 22, 73\}$:

M_i	17	98	62	22	73
$C_i = M_i^{17} \bmod 323$	85	13	232	260	158
$P_i = C_i^{17} \bmod 323$	17	98	62	22	73

Izaberemo li $P = (5, 323)$ i $S = (173, 323)$ dobivamo:

M_i	17	98	62	22	73
$C_i = M_i^5 \bmod 323$	272	319	180	167	99
$P_i = C_i^{173} \bmod 323$	17	98	62	22	73

□

59

Dobrota RSA kriptosustava

- zasniva se na teškoći faktoriziranja velikih brojeva
- uz poznati javni ključ $P = (e, n)$ uljez bi mogao odrediti privatni ključ $S = (d, n)$ ako uspije faktorizirati broj n tj. saznati proste brojeve p i q .
- tada bi on mogao izračunati $\varphi(n)$ i odrediti pripadni d iz uvjeta

  $e\times d = k\times \varphi(n) + 1$
- Međutim, faktoriziranje velikih brojeva je vrlo teško. Najjednostavnije je provesti dijeljenja nizom brojeva

  $2, 3, \ldots, \sqrt{n}$
- Do sada, osim faktoriziranja broja n , nisu pronađeni drugi načini za razbijanje RSA kriptosustava.
- S današnjom računalnog snagom je moguće faktorizirati 640 bitovne brojeve, ali je već nemoguće u razumnom vremenu faktorizirati 1024 bitovne brojeve.

60

Dobrota RSA kriptosustava

- uz 1024 bitni $n = p \times q$, brojevi p i q imaju po 512 bitova
- $2^{512} \approx 10^{150} \Rightarrow$ treba pronaći proste brojeve s oko 150 znamenaka
- Srećom prosti brojevi nisu previše rijetki!
- Funkcija gustoće prostih brojeva $\pi(n)$ je funkcija koja daje broj prostih brojeva manjih od n .

  $\pi(15) = 6$ (prosti brojevi manji od 15 su: 2,3,5,7, 11,13).

$$\lim_{n\to\infty} \frac{\pi(n)}{n/\ln n} = 1.$$

- za n >> vrijedi aproksimacija $\boldsymbol{\pi(n) \approx n / \ln n}$

n	10^{120}	10^{130}	10^{140}	10^{150}
$n/\ln n$	3.62×10^{117}	3.34×10^{127}	3.3×10^{137}	2.89×10^{147}

- u intervalu između 10^{140} i 10^{150} ima

  $2.89\times10^{147} - 3.3\times10^{137} \approx 2.89\times10^{147}$

  (podsjećamo: u svemiru ima manje od 10^{80} atoma)
- svakom atomu u svemiru mogli pridijeliti 10^{67} prostih brojeva

Pronalaženje prostih brojeva

- u prvih n brojeva ima približno $\pi(n) = n / \ln n$ prostih brojeva
- vjerojatnost da je nasumično odabran broj prost je $1 / \ln n$
- u okolici nasumično odabranog broja treba ispitati

  4 ln *n* / 10

  brojeva kako bi se pronašao prost broj (dijeliti s 2,3, ..., $\sqrt{n}$)
- jer su kandidati brojevi kojima je zadnja znamenka 1, 3, 7 ili 9

n	10^{120}	10^{130}	10^{140}	10^{150}
4ln(n)/10	112	121	130	140

Heurističko ispitivanje po Milleru - Rabinu

- zasniva se na Fermatovom teoremu

  $a^{p-1} \equiv 1 \pmod p$ za sve $a \in Z_p^*$
- problem: za Carmichaelove brojeve također vrijedi Fermatov teorem
- takvi su brojevi srećom rijetki: ima ih samo 255 manjih od 10^8
- Fermatov teorem za Carmichaelov broj $561 = 3 \cdot 11 \cdot 17$:

$2^{560} \bmod 561 = 1$	$8^{560} \bmod 561 = 1$
$3^{560} \bmod 561 = 375$	$9^{560} \bmod 561 = 375$
$4^{560} \bmod 561 = 1$	$10^{560} \bmod 561 = 1$
$5^{560} \bmod 561 = 1$	$11^{560} \bmod 561 = 154$
$6^{560} \bmod 561 = 375$	$12^{560} \bmod 561 = 375$
$7^{560} \bmod 561 = 1$	$13^{560} \bmod 561 = 1$

- Domaća zadaća: Carmichaelov složeni broj $1105 = 5 \cdot 13 \cdot 17$

Heurističko ispitivanje po Milleru - Rabinu

- kako bi se smanjila vjerojatnost pogreške, treba ispitati Fermatov teorem za više brojeva a
- osim Fermatovog teorema ispituje se da li za dani broj postoji netrivijalni drugi korijen:

  $x^2 \equiv 1 \pmod n$, gdje je $x \neq 1$ i $x \neq n-1$
- ako postoji, tada je n sigurno složeni broj

Funkcija će utvrditi složenost kada se ustanovi:

- da je za dani n Fermatov teorem nije zadovoljen ili
- da za dani n postoji netrivijalni drugi korijen od 1 mod n

- ako se utvrdi složenost, broj je sigurno složen!
- ipak, postoji (*jako*) mala vjerojatnost da za neki složeni broj spomenuti algoritam utvrdi da se radi o prostom broju


```
// modularno potenciranje
// d = b ^ a mod n
d = 1;
i = m;
dok je (i >= 0) {
    d = (d * d) mod n;
    ako je (a[i] == 1) {
        d = (d*b) mod n;
    }
    i --;
}
```

```
provjera_složenosti(a,n,G){
   G = 0;
   d = 1;
   c = n-1;
   i = -1;
   dok je c > 0 {// c zapisan binarno u b[]
      i ++;
      b[i] = c mod 2;
      c = c div 2;
   } // i je broj bitova-1 od c
   dok je ((i >= 0)∧(G == 0)){
      ds = d;
      d = (d * d) mod n;
      ako je ((d == 1)∧(ds != 1)∧(ds != n-1)){
          G = 1;          // postoji netrivijalni drugi korijen!
      }
      ako je (b[i] == 1) {
          d = (d*a) mod n;
      }
      i --;
   }
   ako je ((i == -1)∧(d != 1)){
      G = 1; //Fermatov teorem
                nije zadovoljen!
   }
}
```

```
G = 0;
i = k ;
dok je ((i >= 0)∧(G == 0)){
   a = random (1,n-1);
   provjera_složenosti(a,n,G);
   i--;
}
ako je (G == 1) {
   n je složeni broj // sigurno!
}
inače {
   n je prosti broj  // skoro sigurno!
}
```

Digitalna omotnica

- osigurava tajnost
- pošiljatelj kriptira poruku **proizvoljnim** ključem K simetričnim algoritmom kriptiranja
- simetrični (sjednički) ključ K se kriptira javnim ključem primatelja P_B
- kriptirana poruka i kriptirani ključ čine digitalnu omotnicu

$$A \xrightarrow{\{\, E(m,K);\ E(K,P_B)\ \}} B$$

P_A, S_A, P_B — P_B, S_B, P_A

67

Digitalni potpis

- pošiljatelj iz poruke izračunava **sažetak** koristeći *hash* funkciju (MD5, SHA-1, ...)
- sažetak se potom kriptira privatnim ključem pošiljatelja S_A i dodaje se izvornoj poruci
- **ne osigurava tajnost**!
- služi za utvrđivanje besprijekornosti informacije i za identifikaciju pošiljatelja i ...

$$A \xrightarrow{\{\, m;\ E[\ H(m),\ S_A\ ]\ \}} B$$

P_A, S_A, P_B — P_B, S_B, P_A

68

Digitalni potpis osigurava

- **autentičnost** - identitet pošiljatelja utvrđuje se dešifriranjem sažetka poruke
- **integritet** - provjerom sažetka poruke utvrđuje se je li se poruka mijenjala na putu do primatelja
- **neporecivost** - pošiljatelj ne može poreći sudjelovanje u transakciji, jer jedino on ima pristup do svog privatnog ključa kojim je potpisao poruku

69

Digitalni pečat

- digitalni pečat je digitalno potpisana digitalna omotnica
- digitalnim potpisom nije osigurana tajnost poruke (poruku svatko može pročitati), ali su osigurani autentičnost, integritet i neporecivost
- digitalnom omotnicom je osigurana samo tajnost
- digitalni pečat osigurava sva četiri sigurnosna zahtijeva: tajnost, autentičnost, integritet i neporecivost

70

Pregled asimetričnih kriptosustava

Za razmjenu simetričnih ključeva:	- Diffie-Hellman - RPK - KEA
Asimetrični algoritmi s parom ključeva:	- RSA - Blum-Goldwaser - **ECC** - El Gamal - LUC

71

Funkcije za izračunavanje sažetka poruke

Funkcije sažimanja ili *hash* funkcije

MD5

- *Message Digest* = sažetak poruke
- proizvodi sažetak duljine **128** bitova
- izvorni tekst dijeli se na blokove duljine **512** bitova
- zadnji blok teksta se nadopunjuje do 512 bitova tako da se:
  - iza zadnjeg bita teksta dodaje jedna jedinica
  - nakon 1 upisuju se nule tako da u bloku preostanu 64 bita
  - u ta 64 bita se upisuje bitovna duljina izvorne poruke
- svaki blok se dijeli na 16 podblokova po 32 bita:

$$M_0, M_1, M_2, \ldots, M_{15}$$

72

MD5

- postupak se obavlja u 64 koraka podijeljena u 4 kruga
  $\Rightarrow$ svaki krug se sastoji od 16 koraka
- u svakom krugu koristi se jedna od četiri funkcije:

$F_i(x,y,z) = (x \wedge y) \vee (!x \wedge z)$, $\quad 1 \le i \le 16$
$F_i(x,y,z) = (x \wedge z) \vee (y \wedge !z)$, $\quad 17 \le i \le 32$
$F_i(x,y,z) = x \oplus y \oplus z$, $\quad 33 \le i \le 48$
$F_i(x,y,z) = y \oplus (x \vee !z)$, $\quad 49 \le i \le 64$

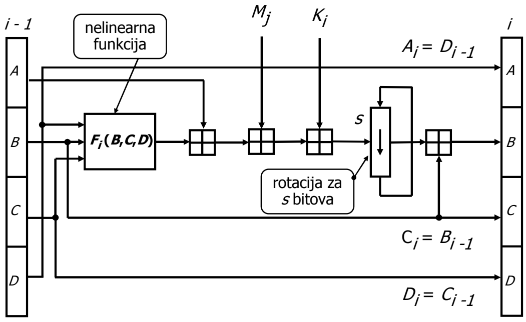
- u svakom koraku koristi se sljedeća varijabla:

$K_i = 2^{32} \times \mathrm{abs}(\sin(i))$, $\quad 1 \le i \le 64$

- 128 - bitovni sažetak S sastoji se od nadovezane četiri 32-bitovne varijable koje su početno inicijalizirane:

$A_0 = 01234567_{16}$ $\quad B_0 = 89ABCDEF_{16}$
$C_0 = FEDCBA98_{16}$ $\quad D_0 = 76543210_{16}$

73



$B_i = B_{i-1} + ((A_{i-1} + F_i(B_{i-1}, C_{i-1}, D_{i-1}) + M_j + K_i) <<< s)$

$S = ABCD$, gdje su: $A = A_{64} + A0$ $\quad B = B_{64} + B0$
$C = C_{64} + C0$ $\quad D = D_{64} + D0$

74

SHA (*Secure Hash Algorithm*)

1995. – NSA je predložila SHA-1 kao zamjenu za SHA-0
1998. – objavljen uspješan napad na SHA-0, ali ne i na SHA-1
2004. – uspješan napad na MD4, MD5, Haval-128, RIPEMD, SHA-0, ali ne i na SHA-1
2005. – uspješan napad na SHA-1

- proizvodi 160-bitovni sažetak
- podjela na podblokove i nadopuna izvornog teksta isto kao i kod MD5: $M_0, M_1, M_2, \ldots, M_{15}$
- sažetak S od 160 bitova sastoji se od 5 nadovezanih 32-bitovnih varijabli koje se inicijaliziraju s vrijednostima:

$A_0 = 67452301_{16}$ $\quad B_0 = EFCDAB89_{16}$
$C_0 = 98BADCFE_{16}$ $\quad D_0 = 10325476_{16}$ $\quad E_0 = C3D2E1F0_{16}$

75

SHA-1

- podblokovi $M_0, \ldots, M_{15}$ služe za stvaranje 80 riječi

$W_i = M_{i-1}$, $\quad 1 \le i \le 16$
$W_i = (W_{i-3} \oplus W_{i-8} \oplus W_{i-14} \oplus W_{i-16}) <<< 1$, $\quad 17 \le i \le 80$

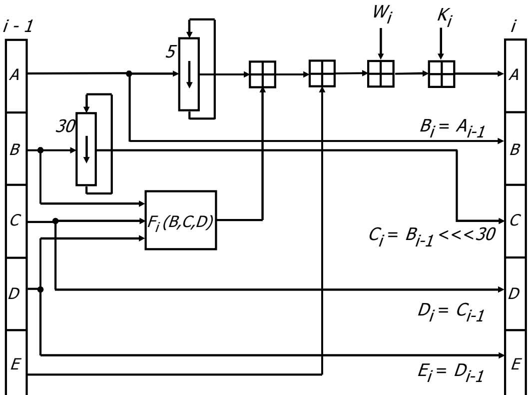
- sastoji se od 4 kruga, ali s 20 koraka, tj. ukupno 80 koraka
- u svakom krugu koriste se sljedeće funkcije i konstante:

$F_i = (X \wedge Y) \vee (\neg X \wedge Z)$, $\quad 1 \le i \le 20$
$F_i = X \oplus Y \oplus Z$, $\quad 21 \le i \le 40$
$F_i = (X \wedge Y) \vee (X \wedge Z) \vee (Y \wedge Z)$, $\quad 41 \le i \le 60$
$F_i = X \oplus Y \oplus Z$, $\quad 61 \le i \le 80$

$K_i = 5A827999_{16}$, $\quad 1 \le i \le 20$
$K_i = 6ED9EBA1_{16}$, $\quad 21 \le i \le 40$
$K_i = 8F1BBCDC_{16}$, $\quad 41 \le i \le 60$
$K_i = CA62C1D6_{16}$, $\quad 61 \le i \le 80$

76

$A_i = (A_{i-1} <<< 5) + F_i(B_{i-1}, C_{i-1}, D_{i-1}) + E_{i-1} + W_i + K_i$



$S = A_{80}B_{80}C_{80}D_{80}E_{80}$

77

Važna svojstva funkcija za izračunavanje sažetka poruke

Otpornost na izračunavanje originala
(*preimage resistance*)

- $H = h(M) \quad \Rightarrow M = h^{-1}(H)$ ne postoji

Otpornost na izračunavanje poruke koja daje isti sažetak
(*2-nd preimage resistance*)

- za poznati M i $H = h(M)$ je nemoguće pronaći M' koji daje isti H

Otpornost na kolizije (*collision resistance*)

- nemoguće je pronaći bilo koje dvije poruke M_1 i M_2 za koje se dobiva isti sažetak $h(M_1) = h(M_2)$

78

SHA-2

- osmislila je NSA, a NIST publicirao 2001
- 2.11.2007. – NIST raspisuje natječaj za SHA-3
- dok se ne odabere algoritam SHA-3, preporuča se SHA-2
- to je skup funkcija (broj označava veličinu sažetka):
  - SHA-224
  - SHA-256
  - SHA-384
  - SHA-512

Algoritam	Sažetak	Stanje	Blok	Poruka	Arhitektura	Broj rundi	Funkcije
SHA-1	160	160	512	$2^{64}-1$	32	80	+, and, or, xor, rot
SHA-256/224	256/224	256	512	$2^{64}-1$	**32**	64	+, and, or, xor, shift, rot
SHA-512/384	512/384	512	1024	$2^{128}-1$	**64**	80	+, and, or, xor, shift, rot

SHA-2

- http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf
- zadnji blok teksta se nadopunjuje do 512 bitova na isti način kao i SHA-1
- poruka se podijeli na blokove od po 512 bita:

$$M^{(1)}, M^{(2)}, ..., M^{(N)}$$

- svaki blok se dijeli na 16 podblokova po 32 bita :

$$M_0 , M_1 , M_2, \ldots, M_{15}$$

- $H^{(0)}=a_0b_0c_0d_0e_0f_0g_0h_0$

```
a0=6a09e667   e0=510e527f
b0=bb67ae85   f0=9b05688c
c0=3c6ef372   g0=1f83d9ab
d0=a54ff53a   h0=5be0cd19
```


SHA-2

- koristi se zbrajanje po modulu 2^{32}
- koriste se sljedeće funkcije:

$$Ch(X,Y,Z) = (X \wedge Y) \oplus (\neg X \wedge Z)$$
$$Maj(X,Y,Z) = (X \wedge Y) \oplus (X \wedge Z) \oplus (Y \wedge Z)$$

$$S_0(x) = ROTR^2(x) \oplus ROTR^{13}(x) \oplus ROTR^{22}(x)$$
$$S_1(x) = ROTR^6(x) \oplus ROTR^{11}(x) \oplus ROTR^{22}(x)$$
$$F_0(x) = ROTR^7(x) \oplus ROTR^{18}(x) \oplus SHR^3(x)$$
$$F_1(x) = ROTR^{17}(x) \oplus ROTR^{19}(x) \oplus SHR^{10}(x)$$

- koriste se sljedeće konstante:

K_t=5a827999, za $0 \le t \le 19$
K_t=6ed9eba1, za $20 \le t \le 39$
K_t=8f1bbcdc, za $40 \le t \le 59$
K_t=ca62c1d6, za $60 \le t \le 79$

SHA-2

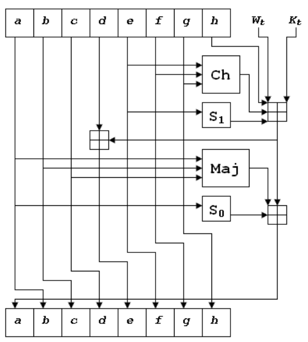
Za i=1 do N radi{

- Priprema (izračunavanje Wt)
- $W_t = M_t^{(i)}$, $0 \le t \le 15$
  $W_t = F_1(W_{t-2}) + W_{t-7} + F_0(W_{t-15}) + W_{t-16}$, $16 \le t \le 63$
- $a=H_0^{(i-1)}$ $b=H_1^{(i-1)}$ $c=H_2^{(i-1)}$ ... $h=H_7^{(i-1)}$
- Za t=0 do 63 radi{ // u 64 koraka računaj *abcdefgh*
  Računaj a,b,c,d,e,f,g,h
  }
- $H_0^{(i)}=a+H_0^{(i-1)}$
  $H_1^{(i)}=b+H_1^{(i-1)}$
  ...
  $H_7^{(i)}=h+H_7^{(i-1)}$

}

SHA-2

Računaj a,b,c,d,e,f,g,h:



SHA-3

- postupak odabira jednog od 64 pristigla algoritma je još uvije u tijeku:
  – 1.krug veljača, 2009.
  – travanj 2009: preostao 41 kandidat
  – 2.krug 2010.
  – Travanj 2010: preostalo 14 kandidata
  – 9.12.2010. objavljen je popis 5 finalista:
    - BLAKE
    - Groestl (Knudsen)
    - JH
    - Keccak (Daemen)
    - Skein (Schneier)
  – konačan odabir 2012.

Rođendanski napad (birthday attack)

- vjerojatnost da dvije poruke iz skupa od $k=1.2(2^n)^{1/2} = =1.2 \cdot 2^{n/2}$ poruka daju isti sažetak je veća od 50%, gdje je n duljina sažetka
- analogno: vjerojatnost da dvije osobe u dvorani u kojoj je ukupno $k=1.2 \cdot 365^{1/2} = 23$ ljudi imaju isti dan rođendan je veća od 50%

M_1: "UGOVOR: Za 657200 kn je Ana Twofish kupila stan od Branka Horvata."

M_2: "UGOVOR: Za 176450 kn je Ana Twofish kupila stan od Branka Horvata."

85

```
M1.txt
00000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
00000010 2f ca b5 87 12 46 7e ab 40 04 58 3e b8 fb 7f 89
00000020 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 71 41 5a
00000030 08 51 25 e8 f7 cd c9 9f d9 1d bd f2 80 37 3c 5b
00000040 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
00000050 35 73 9a c7 f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
00000060 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79 cc 15 5c
00000070 ed 74 cb dd 5f c5 d3 6d b1 9b 0a d8 35 cc a7 e3
M2.txt
00000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
00000010 2f ca b5 07 12 46 7e ab 40 04 58 3e b8 fb 7f 89
00000020 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 f1 41 5a
00000030 08 51 25 e8 f7 cd c9 9f d9 1d bd 72 80 37 3c 5b
00000040 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
00000050 35 73 9a 47 f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
00000060 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79 4c 15 5c
00000070 ed 74 cb dd 5f c5 d3 6d b1 9b 0a 58 35 cc a7 e3

$ md5sum M1.txt
MD5 Sum = a4c0d35c95a63a805915367dcfe6b751
$ md5sum M2.txt
MD5 Sum = a4c0d35c95a63a805915367dcfe6b751
```

86

```
M1.txt
00000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
00000010 2f ca b5 87 12 46 7e ab 40 04 58 3e b8 fb 7f 89
00000020 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 71 41 5a
00000030 08 51 25 e8 f7 cd c9 9f d9 1d bd f2 80 37 3c 5b
00000040 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
00000050 35 73 9a c7 f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
00000060 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79 cc 15 5c
00000070 ed 74 cb dd 5f c5 d3 6d b1 9b 0a d8 35 cc a7 e3
M2.txt
00000000 d1 31 dd 02 c5 e6 ee c4 69 3d 9a 06 98 af f9 5c
00000010 2f ca b5 07 12 46 7e ab 40 04 58 3e b8 fb 7f 89
00000020 55 ad 34 06 09 f4 b3 02 83 e4 88 83 25 f1 41 5a
00000030 08 51 25 e8 f7 cd c9 9f d9 1d bd 72 80 37 3c 5b
00000040 96 0b 1d d1 dc 41 7b 9c e4 d8 97 f4 5a 65 55 d5
00000050 35 73 9a 47 f0 eb fd 0c 30 29 f1 66 d1 09 b1 8f
00000060 75 27 7f 79 30 d5 5c eb 22 e8 ad ba 79 4c 15 5c
00000070 ed 74 cb dd 5f c5 d3 6d b1 9b 0a 58 35 cc a7 e3

$ md5sum M1.txt
MD5 Sum = a4c0d35c95a63a805915367dcfe6b751
$ md5sum M2.txt
MD5 Sum = a4c0d35c95a63a805915367dcfe6b751
```

87

Pregled funkcija za izračunavanje sažetka

HAVAL	128, 160, 192, 224, 256	MD4, MD5	128
Panama	256	RIPEMD	160
Sapphire II	128, 136, 144,...,256	SNEFRU	128, 256
SHA-1	160, 256, 384 i 512	Tiger	192

Preporuke

	Minimalno	Preporuka	US1	US2
Simetrični (ne DES!)	96	256	64	56 (40)
Asimetrični: RSA	1024	4096	1024	512
Asimetrični: EC	192	256	160	112
HASH	160	SHA-256	-	-
Klasa certifikata	2	3	-	-

88

Sigurnosni protokoli

Diffie - Hellmanov postupak

- služi za razmjenu tajnog ključa
- Ana i Branko se unaprijed slože o dva vrlo velika broja n i g:

  $$\text{nzd}(g, n) = 1$$
- najpraktičnije: za n odabrati veliki prosti broj p
- g i p se mogu javno objaviti
- Ana odabire veliki nasumični prirodni broj x i šalje Branku:

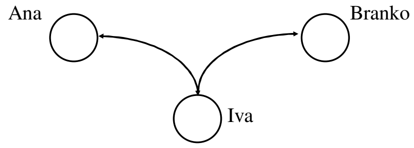
  $$\boldsymbol{X = g^x \bmod p}$$
- Branko odabire veliki nasumični prirodni broj y i šalje Ani:

  $$\boldsymbol{Y = g^y \bmod p}$$
- Ana dobiva: $\boldsymbol{K = Y^x \bmod p = (g^y)^x \bmod p = g^{xy} \bmod p}$
- Branko također: $\boldsymbol{K = X^y \bmod p = (g^x)^y \bmod p = g^{xy} \bmod p}$

89

Diffie - Hellmanov postupak

- napad *man in the middle*



- napadač (Iva) računa na temelju objavljenih g i p:

  $$\boldsymbol{Z = g^z \bmod p}$$
- napadač komunicira s Anom i Brankom (lažno se predstavljajući) uz pomoć dva ključa K_A i K_B:

  $$K_A = X^z \bmod p = (g^x)^z \bmod p = g^{xz} \bmod p$$

  $$K_B = Y^z \bmod p = (g^y)^z \bmod p = g^{yz} \bmod p$$

90

Raspodjela ključeva u zatvorenom simetričnom kriptosustavu

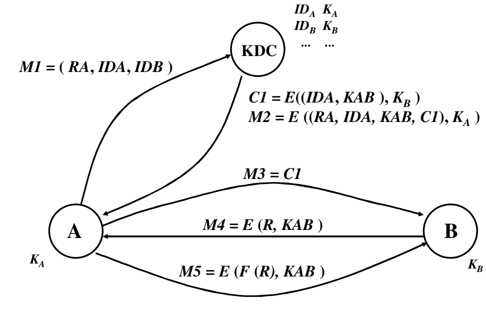
Raspodjela ključeva prema Needhamu i Schroederu

- za N sudionika: ukupno $N \cdot (N\text{-}1) / 2$ tajnih ključeva i svaki sudionik bi morao pohraniti N - 1 ključeva
  $\Rightarrow$ ozbiljno je ugrožena sigurnost!
- rješenje: pouzdani poslužitelj u kojem imaju svi povjerenje

Centar za raspodjelu ključeva (*Key Distribution Center - KDC*)

- potencijalni sudionici moraju se unaprijed prijaviti
- dodjeljuje im se tajni ključ za komuniciranje s KDC
- KDC obznanjuje identifikatore svih prijavljenih sudionika a zadržava u tajnosti pripadnu tablicu tajnih ključeva

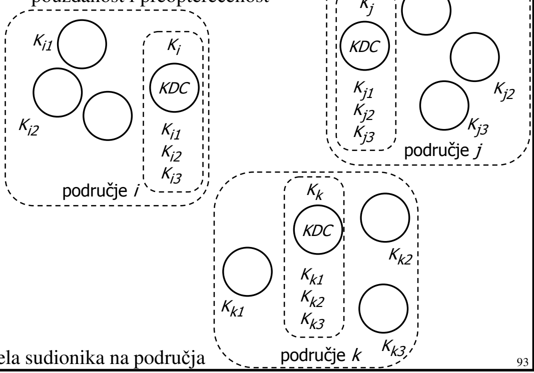
Raspodjela ključeva u zatvorenom simetričnom kriptosustavu



R = nonce (*number used only once*)

Raspodijeljena raspodjela ključeva

- nedostatak jednog KDCa: smanjena pouzdanost i preopterećenost



Raspodjela sudionika na područja

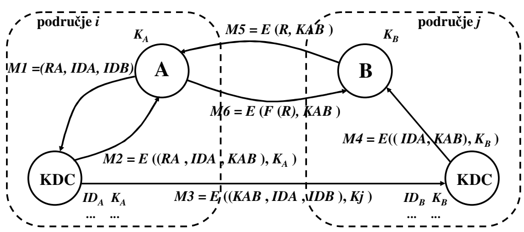
Primjer 11.9. - u zatvorenom sustavu je 100 sudionika

a) Bez centra treba unaprijed podijeliti $100 \times 99 / 2$ = 4950 ključeva. Svaki sudionik mora čuvati 99 ključeva.

b) S jednim *KDC* treba unaprijed podijeliti 100 ključeva. Svaki sudionik čuva samo svoj (1) ključ, a *KDC* 100 ključeva.

c) U raspodijeljenom sustavu postoji $10 \times 9 / 2$ = 45 ključeva za komunikaciju između centara. Svaki *KDC* čuva njih *9*.
   U svakom područnom KDC postoji 10 ključeva za komuniciranje sa sudionicima unutar područja.
   Svaki sudionik čuva samo svoj (1) ključ za komuniciranje s područnim centrom.
   Područni KDC mora čuvati i tih 10 ključeva, tako da on čuva ukupno $9 + 10 =$ 19 ključeva.
   U tom se sustavu koristi se ukupno
   $10 \times 10 + 45 = 145$ ključeva.

- sudionici se nalaze u različitim područjima: uspostavljanje sigurnog kanala se može obaviti razmjenom 6 poruka



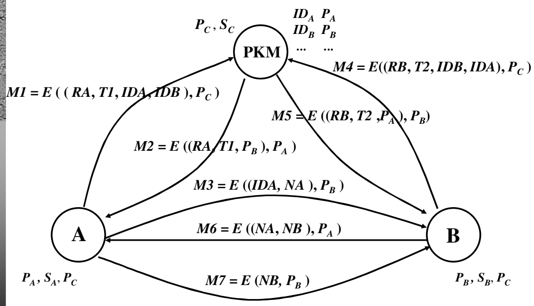
Raspodjela ključeva u zatvorenom asimetričnom kriptosustavu

- raspodjeljuju se samo javni ključevi
- problem: svatko se može lažno predstaviti

Centar za raspodjelu javnih ključeva (*public key manager* - PKM)

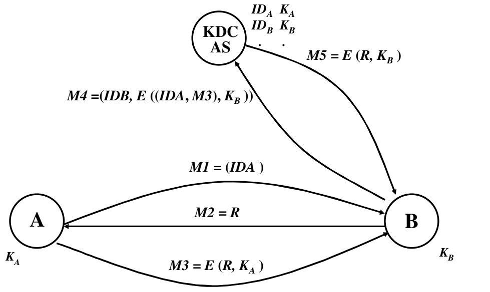
- potencijalni sudionici moraju se unaprijed prijaviti i autentificirati
- privatni ključ sudionici čuvaju za sebe
- PKM obznanjuje identifikatore svih prijavljenih sudionika i čuva pripadnu tablicu javnih ključeva
- *prije raspodjele ključeva potrebno je obaviti autentifikaciju*

Raspodjela ključeva u zatvorenom asimetričnom kriptosustavu



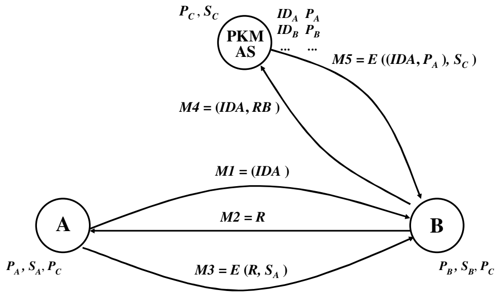
97

Jednostrana autentifikacija u zatvorenom simetričnom kriptosustavu



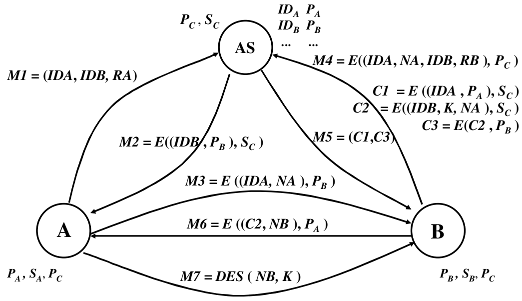
98

Jednostrana autentifikacija u zatvorenom asimetričnom kriptosustavu



99

Obostrana autentifikacija u zatvorenom asimetričnom sustavu



100

Prijava za rad

- **ime korisnika** i **lozinka** (*user name*, *password*)
- iz imena se izvodi *identifikator* korisnika (*user identifier*)
- slabosti:
  - korisnici mogu sami lako otkriti svoje podatke jer ih obično zapisuju
  - napadači su vrlo domišljati pri otkrivanju identifikatora i lozinki
  - datoteke s identifikatorima i lozinkama su meta napadača
  - engleski rječnik ima približno 350 000 riječi
    ⇒ napad "grubom silom" traje u najgorem slučaju 35 s

101

Prijava za rad

- metode za povećanje sigurnosti:
  - broj mogućih lozinki mora biti velik čime se smanjuje vjerojatnost pogađanja lozinke
  - postupak prijave mora biti takav da se dozvoljava samo ograničeni broj ponavljanja netočne lozinke
  - operacijski sustav mora pohranjivati pokušaje neovlaštenog pristupa kako bi se olakšala naknadna istraga
  - kriptiranje lozinki:
    = umjesto $C = E(P, K)$ koristiti $C = E(P, P)$ ili *hash* fju
  - OS prilikom registracije treba onemogućiti jednostavnu i/ili kratku lozinku ili predložiti slučajno generiranu
  - nadopuniti lozinku nasumičnim brojem koji se čuva u posebnoj tablici i mijenja se kod svake promjene lozinke

102

Zaštita pristupanja pojedinim sredstvima - autorizacija

- autentifikacija + provjera prava pristupa (access control)
- mehanizmi *dopuštanja pristupa* (*access control*) sredstvima nazivaju se *autorizacijom* pristupa (*authorization*)
- **subjekti**: korisnici ili njihovi procesi ili čak neke dretve unutar tih procesa
- **objekti** zaštite: sredstva koja se zaštićuju
- **zaštitna pravila** (*protection rules*)
  - za svaki par subjekt-objekt treba odrediti pravo pristupa (obuhvaća i način na koji se objekt smije upotrebljavati)
  - r,w,x ili prazno polje ≡ nema prava pristupa
  - mogu prikazati u obliku matrice pristupa (*access matrix*)
  - svaki subjekt dobiva svoj redak i svaki objekt svoj stupac

OBJEKTI

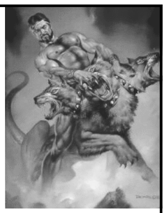
SUBJEKTI

Č, P	I		Č	
		P		P
Č				
	I			
		Č		

Alternativni način zapisa - liste :

- **Lista prava pristupa objektu** (*access control list*)
  - neprazni elementi stupaca matrice pristupa
- **Lista dozvola za pristup objektima** (*capability tickets*)
  - neprazni elementi pojedinih redova matrice

Autentifikacijski protokol Kerberos

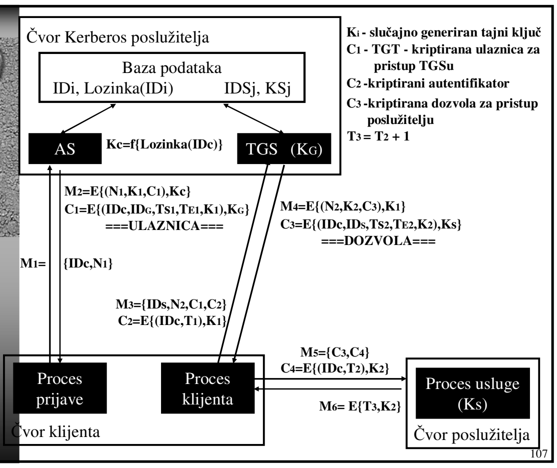


- počeo se razvijati 1978. godine na Massachusets Instutute of Technology (MIT)
- pretpostavka: pouzdana računala, ali je mreža nepouzdana
- koristi simetrični kriptosustav (izvorno: DES)
- treća strana kojoj svi vjeruju
- traži unos lozinke samo jednom (single sign-on) i to na početku sjednice
- lozinka ne putuje mrežom
- osjetljivi podaci se prenose u kriptiranom obliku

Kerberos

Sustav se sastoji od

- čvora klijenta (*client node*)
- čvora poslužitelja (*application server node*) – obavlja traženu uslugu
- čvora Kerberos poslužitelja – sastoji se od:
  - baze podataka:
    - = identifikatori
    - = lozinke
    - = tajni ključevi svih poslužitelja u sustavu
  - poslužitelja (ili procesa) za utvrđivanje autentičnosti (*authentication server – AS*)
  - poslužitelja za dodjelu ulaznica za pristup pojedinim uslugama (*ticket granting server – TGS*)



- tri nivoa zaštite:
  - = provjera autentičnosti samo na početku (mrežni datotečni sustav na MIT mreži)
  - = "sigurne poruke" – uz poruku u jasnom obliku šalje se i kriptirani autentifikator
  - = "privatne poruke" – kriptirana poruka i kriptirani autentifikator
- distribucije Kerberosa donose kerberizirane verzije najpopularnijih aplikacija (npr. rlogin, telnet, ftp...)
- ograničenja i nedostaci:
  - = svaki program treba biti "kerberiziran"
  - = nema autorizacije
  - = Kerberos server mora biti fizički zaštićen
  - = kako sigurno pohraniti tajne ključeve?
  - = podliježe strogim američkim zakonima o izvozu kriptotehnologije

Infrastruktura javnih ključeva *PKI – Public Key Infrastructure*

- skup tehnologija, protokola, normi i usluga koji zajedno omogućuju sigurnu komunikaciju temeljenu na sustavu javnih ključeva preko nesigurnih mreža

PKI infrastruktura trebala bi pružiti sljedeće:

- integritet elektronički primljene ili poslane poruke
- sigurnost u identitet pošiljaoca i primaoca informacije
- pouzdanost vremena i datuma slanja informacije
- formalnopravnu valjanost elektroničke poruke u sudskim procesima

109

Osnovna zadaća PKI sustava

- nedvojbeno povezivanje javnih ključeva sa korisnicima te provjera jesu li ključevi trenutno važeći
- off-line provjera identiteta: certifikatima

Digitalni certifikat

- rješava problem dokazivanja identiteta stranaka
- skup bitnih informacija koje identificiraju korisnika (pošiljatelja) i poslužitelja (davatelja usluge)
- izdaje pouzdano certifikacijsko tijelo: izdavači cerifikata (CA - *Certificate Authorities*)
- Certifikat je svjedodžba koja potvrđuje da je određeni korisnik u trenutku izdavanja certifikata posjedovao privatni ključ koji odgovara javnom ključu u certifikatu.

PKI

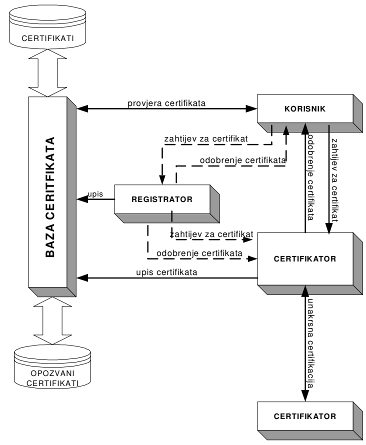
110

Dijelovi PKI sustava

1. Korisnik
2. Certifikator (CA - *Certificate Authority*)
   – stvara i izdaje certifikate, potvrđuje da neki javni ključ pripada određenoj osobi
3. Registrator (RA - *Registration Autrhority*)
   – prima zahtjeve od korisnika, provjerava njihov identitet i prosljeđuje zahtjev CA, ali ne izdaje certifikate
   – opcionalni element PKI sustava
4. Baza certifikata
   – važeći certifikati sa datumom isteka
   – opozvani certifikati s datumom opoziva
5. Sustav za upravljanje certifikatima (objavljivanje, provjera, dohvat certifikata po zadanim uvjetima)
6. Sustav za rekonstrukciju izgubljenih ključeva
7. Sustav za pouzdano vremensko označavanje dokumenata i potpisa (TSA – *Time Stamp Autrhority*)
   – dodaje se vremenska oznaka (*time stamp*)

PKI

111



PKI

112

Certifikat je u digitalnom obliku, a sadrži minimalno:

- identifikator certifikata
- osnovne podatke o nositelju certifikata
- vrijeme i datum izdavanja certifikata
- rok valjanosti certifikata
- klasu certifikata
- identitet izdavatelja certifikata
- digitalni potpis izdavatelja certifikata i identifikaciju algoritma
- javni ključ nositelja certifikata i identifikaciju algoritma
- namjena javnog ključa nositelja certifikata

Norme koje propisuju sadržaj certifikata:

- X.509
- SPKI
- PGP

PKI

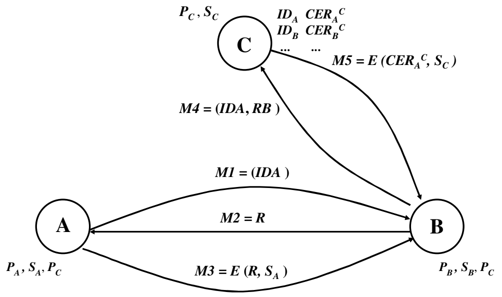
113

```
Certificate:
   Data:
       Version: 3 (0x2)
       Serial Number: 75 (0x4b)
       Signature Algorithm: md5WithRSAEncryption
       Issuer: C=US, ST=WA, L=Seattle, O=Thawte Consulting cc,
               OU=Certification Services Division,
               CN=Thawte Server CA/emailAddress=certs@thawte.com
       Validity
           Not Before: May 13 23:33:08 2008 GMT
           Not After : Dec 31 23:59:59 2020 GMT
       Subject: C=HR, ST=Grad Zagreb, L=Zagreb, O=FER, OU=CIP,
                CN=webmail.fer.hr/emailAddress=korisnik@webmail.fer.hr
       Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
           RSA Public Key: (1024 bit)
               Modulus (2048 bit):
                    00:cd:66:28:fb:b8:b3:b7:e0:72:77:48:2d:08:04:
                    e1:6d:1c:c5:4f:57:73:0c:e6:db:3b:8e:cd:c6:25:
                    61:7f:60:c9:da:a3:9f:1d:fa:d8:ef:00:7b:f9:54:
                    65:ab:7e:9e:9b:6d:ff:d4:12:ad:f8:ac:87:6e:83:
                    ec:65:5f:b4:2d:eb:b8:dc:1c:d7:32:b7:46:a5:e3:
                    a1:6c:0b:4c:1b:0c:89:0a:fb:0e:3a:c0:0f:af:b2:
                    62:1d:2f:60:e4:b1:27:b4:7c:59:00:2c:19:e9:f3:
                    a3:88:fe:01:d6:56:be:26:c7:f8:42:b1:79:39:98:
                    a1:b4:4a:84:dd:20:ca:e7:a9:db:6d:a6:73:88:e7:
                    81:8b:3e:81:3d:00:e5:5d:7f:3d:9b:cd:ba:9b:28:
                    88:88:7f:d7:69:2c:66:eb:8f:79:b8:ec:bc:bb:76:
                    67:b1:00:2a:70:bd:f1:21:66:6f:ba:74:81:82:30:
                    02:c0:a8:57:f8:9f:76:02:df:7f:49:44:4a:32:93:
                    48:a4:25:73:47:10:21:20:fe:b6:d2:09:1a:60:4f:
                    a5:d9:df:ea:55:49:43:c6:ce:96:0b:7d:a7:22:c1:
                    3e:5b:28:2e:2c:04:7a:b2:93:89:db:d8:2b:59:86:
                    a3:0a:c1:6f:f9:56:b2:a5:71:4c:4b:74:f3:b8:a1:
                    b4:65
               Exponent: 65537 (0x10001)
       X509v3 extensions:
           X509v3 Basic Constraints: critical
               CA:TRUE
   Signature Algorithm: md5WithRSAEncryption
       07:fa:4c:69:5c:fb:95:cc:46:ee:85:83:4d:21:30:8e:ca:d9:
       a8:6f:49:1a:e6:da:51:e3:60:70:6c:84:61:11:a1:1a:c8:48:
       3e:59:43:7d:4f:95:3d:a1:8b:b7:0b:62:98:7a:75:8a:dd:88:
       4e:4e:9e:40:db:a8:cc:32:74:b9:6f:0d:c6:e3:b3:44:0b:d9:
       8a:6f:9a:29:9b:99:18:28:3b:d1:e3:40:28:9a:5a:3c:d5:b5:
       e7:20:1b:8b:ca:a4:ab:8d:e9:51:d9:e2:4c:2c:59:a9:da:b9:
       b2:75:1b:f6:42:f2:ef:c7:f2:18:f9:89:bc:a3:ff:8a:23:2e:
       70:47
```

PKI

114

Postupak jednostrane autentifikacije uz pomoć certifikata

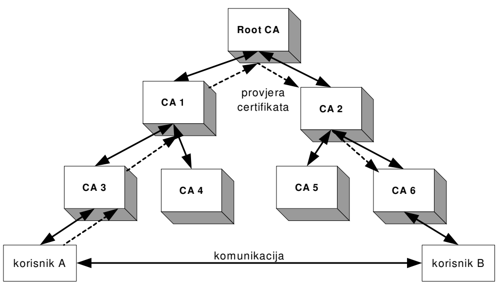


Klasa certifikata

1. nositelj je identificiran samo po svojoj e-mail adresi
2. nositelj je identificiran podacima o identitetu koje je sam podnio
3. nositelj je identificiran provjerom službene isprave podnositelja
4. nositelj je identificiran provjerom službene isprave i fizičkom provjerom: fotografija, biometrika - otisak prsta...

- Baza certifikata je javna baza podataka koja se sastoji od:
  - liste certifikata
  - liste opozvanih certifikata (CRL – *Certificate Revocation List*)

Provjera certifikata u hijerarhijskoj strukturi certifikatora



Problem opoziva certifikata

- u slučaju gubitka ili kompromitiranosti privatnog ključa, korisnik je dužan od certifikatora tražiti opoziv certifikata
- ovaj postupak je najslabija točka PKI sustava, budući nije moguće u istodobno obavijestiti sve zainteresirane
- ovaj problem nije moguće riješiti bez on-line veze i centralizirane baze podataka, što je u suprotnosti s idejom PKI sustava sa certifikatima
- "Is PKI dead or is it just resting?"

Preporučeni X.509 autentifikacijski protokoli

- protokol s jednom porukom (*one - way protocol*)
  - autentificiraju se oba sudionika A i B
  - osigurava integritet sadržaj koji se prenosi sudioniku B
  - uporabom vremenske oznake sprječava napad ponavljanjem poruke
- protokol s dvije poruke (*two - way protocol*)
  - pridodaje se odgovor sudionika B
  - utvrđuje da je upravo sudionik B a ne neki napadač odgovorio na prvu poruku
  - uporabom vremenske oznake sprječava napad ponavljanjem druge poruke;
- protokol s tri poruke (*three - way protocol*)
  - sudionik A vraća treću poruku sudioniku B
  - ne upotrebljavaju se u svim porukama vremenske oznake

Protokol s jednom porukom (*one - way protocol*)

1. Kada *A* želi komunicirati sa sudionikom *B*:

- generira N_A i oblikuje vremensku oznaku T_A (vrijeme, trajanje valjanosti oznake)
- pronalazi put $A \rightarrow B$ te iz CER_B^{CB} saznaje P_B

$$P_B = P_{CA} \bullet (A \rightarrow B) \bullet CER_B^{CB}$$

- oblikuje četvorku (IDB, T_A, N_A, D), gdje je D podatkovna komponenta koja može biti kriptirana sa P_B
- šalje sudioniku B poruku

$$M_1 = (CER_A^{CA}, E((IDB, T_A, N_A, D), S_A))$$

Protokol s jednom porukom (*one - way protocol*)

2. Kada B primi poruku M_1:

- pronalazi u tablicama put $B \rightarrow A$ te iz $\boldsymbol{CER_A^{CA}}$ saznaje i utvrđuje $\boldsymbol{P_A}$

$$P_A = P_{CB} \bullet (A \rightarrow B) \bullet CER_A^{CA}$$

- uz pomoću ključa $\boldsymbol{P_A}$ dobiva ($\boldsymbol{IDB, T_A, N_A, D}$)
- na temelju *IDB* utvrđuje da je poruka stvarno upućena njemu
- na temelju vremenske oznake $\boldsymbol{T_A}$ utvrđuje da je poruka još valjana
- dekriptira svojim privatnim ključem $\boldsymbol{S_B}$ podatkovnu komponentu $\boldsymbol{D}$ ako je bila kriptirana
- može usporediti dobiveni $\boldsymbol{N_A}$ s pohranjenim nasumičnim brojevima iz prethodnih poruka kako bi ustanovio da poruka nije ponovljena

Protokol s dvije poruke (*two - way protocol*)

3. Sudionik B:

- generira $\boldsymbol{N_B}$ i vremensku oznaku $\boldsymbol{T_B}$;
- oblikuje ($\boldsymbol{IDA, T_B, N_A, N_B, D}$), gdje $\boldsymbol{D}$ može biti kriptiran sa $\boldsymbol{P_A}$
- šalje sudioniku *A* poruku:

$$M_2 = E\,((IDA, T_B, N_A, N_B, D\,), S_B\,)$$

4. Kada sudionik *A* primi poruku $\boldsymbol{M_2}$:

- uz pomoću $\boldsymbol{P_B}$ dobiva ($\boldsymbol{IDA, T_B, N_A, N_B, D}$)
- na temelju ***IDA*** utvrđuje da je poruka njemu upućena
- na temelju $\boldsymbol{T_B}$ utvrđuje da je poruka još valjana
- po potrebi dekriptira $\boldsymbol{D}$ uz pomoć $\boldsymbol{S_A}$
- može usporediti $\boldsymbol{N_B}$ s pohranjenim brojevima iz prethodnih poruka kako bi ustanovio je li poruka ponovljena

Protokol s tri poruke (*three - way protocol*)

- u prethodnim porukama ignorira vremenske oznake

6. Sudionik *A* :

- uspoređuje dobiveni $\boldsymbol{N_A}$ iz poruke $\boldsymbol{M_2}$ s izvornom vrijednošću i utvrđuje da je poruka $\boldsymbol{M_2}$ odgovor na $\boldsymbol{M_1}$
- uz pomoću ključa $\boldsymbol{S_A}$ kriptira dobiveni NB i šalje poruku

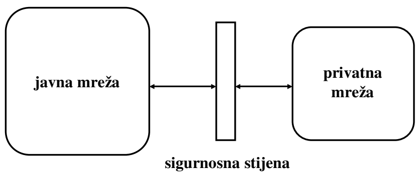
$$M_3 = \mathrm{E}\,(N_B, S_A\,)$$

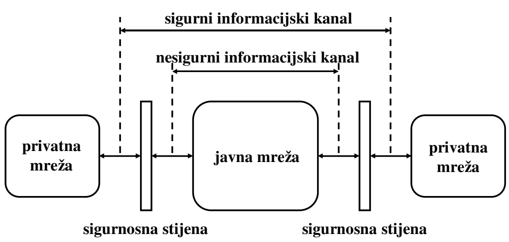
6. Po primitku poruke $\boldsymbol{M_3}$ sudionik *B* :

- uz pomoću ključa $\boldsymbol{P_A}$ dekriptira poruku $\boldsymbol{M_3}$ i dobiva $\boldsymbol{N_B}$
- uspoređuje dobiveni $\boldsymbol{N_B}$ iz poruke $\boldsymbol{M_3}$ s izvornom vrijednošću i utvrđuje da je $\boldsymbol{M_3}$ odgovor na $\boldsymbol{M_2}$

Sigurnosna zaštitna stijena

- računalo ili neka nakupina komunikacijskih naprava koje fizički razdvajaju dvije mreže
- uobičajeno, sigurnosna zaštitna stijena ograničava pristup nekoj privatnoj lokalnoj mreži (ili čak samo jednom računalu) iz javne mreže





- stijene koje filtriraju komunikacijske pakete (*packet filter*)
- stijene koje djeluju kao prividni poslužitelji (*proxy server*), može prikriti pravu IP adresu
- stijene koje djeluju kao stvarni poslužitelji (*full server*), primjerice *FTP* ili *telnet* poslužitelj