# RSA kriptosustav

**Kokanović, Anamarija**

**Undergraduate thesis / Završni rad**

**2017**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Department of Mathematics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Odjel za matematiku**

*Permanent link / Trajna poveznica:* https://urn.nsk.hr/urn:nbn:hr:126:670719

*Rights / Prava:* In copyright / Zaštićeno autorskim pravom.

*Download date / Datum preuzimanja:* **2024-03-29**



*Repository / Repozitorij:*

Repository of School of Applied Mathematics and Computer Science









**Sveučilište J.J. Strossmayera u Osijeku**
**Odjel za matematiku**
**Sveučilišni preddiplomski studij matematike**

Anamarija Kokanović

# RSA kriptosustav

Osijek, 2017.

**Sveučilište J.J. Strossmayera u Osijeku**
**Odjel za matematiku**
**Sveučilišni preddiplomski studij matematike**

Anamarija Kokanović

# RSA kriptosustav

Završni rad

Voditelj: izv. prof. dr. sc. Ivan Matić

Osijek, 2017.

**Sažetak.** Velika je potreba za sigurnim slanjem informacija. U svakodnevnom životu nalazi se sve veća potreba za kriptiranjem i kriptosustavima. U radu ćemo se susresti s osnovnim pojmovima kriptologije, objasniti razliku kriptografije i kriptoanalize, definirati otvoren tekst, šifrat i kriptosustav. Upoznat i obradit ćemo RSA kriptosustave i njihovu implementaciju. U četvrtom poglavlju govoriti ćemo nešto o sigurnosti RSA kriptosustava te kriptoanalizi RSA te semantičkoj zaštiti. Peto poglavlje ćemo posvetiti efektivnosti RSA kriptosustava i modularnom potenciranju. U zadnjem poglavlju analizirat ćemo neke od ranijih napada na RSA kriptosustav kao što su napadi pomoću modula, Håsdatov i ciklički napad.

**Ključne riječi:** kriptosustavi, RSA kriptosustav, sigurnost RSA kriptosustava, efikasnost RSA kriptosustava, raniji napadi na RSA kriptosustave

**Abstract.** There is a huge need for sending information. In everyday life we find a greater need for encryption and cryptosistems. In this work we will find about the basic concept of cryptology, explain the difference between cryptography and cryptanalysis, define plaintext, ciphertext and cryptosystem. We will familiarize with RSA cryptosystem and their implementation. In the fourth chapter we will cover, in short, RSA security, the cryptoanalisis of RSA, and the semantic security. However, the fifth chapter we will dedicade to the efficiency of RSA and modular exponentiation. In the last chapter we will analyze some of the earlier attacks on RSA, such as common modulus attack, Håsdat's broadcast attack and cycling attacks.

**Key words:** cryptosystem, RSA cryptosystem, the security of RSA, effciency of RSA, early attacks on RSA

# Sadržaj

**1. Uvod** **4**

**2. Kriptosustavi** **5**
2.1. Osnovni pojmovi . . . . . 5

**3. RSA kriptosustavi** **7**
3.1. Implementacija RSA kriptosustava . . . . . 8

**4. Sigurnost RSA kriptosustava** **10**
4.1. Faktorizacija modulo $n$ i poznavanje tajnog eksponenta $d$ . . . . . 10
4.2. Kriptoanaliza RSA . . . . . 12
4.3. Homomorfno svojstvo RSA kriptosustava . . . . . 12
4.4. Semantička zaštita . . . . . 12

**5. Efikasnost RSA kriptosustava** **14**
5.1. Generiranje prostih brojeva . . . . . 14
5.2. Modularno potenciranje . . . . . 14

**6. Raniji napadi na RSA kriptosustav** **16**
6.1. Uobičajeni modul napadi . . . . . 16
6.2. Håstadov napad . . . . . 17
6.2.1. Napadi uobičajenog otvorenog teksta . . . . . 17
6.2.2. Napadi srodnih otvorenih tekstova . . . . . 18
6.3. Ciklički napadi . . . . . 18

# 1. Uvod

Kroz cijelu povijest postojala je potreba za sigurnom i tajnom komunikacijom. Ona je podrazumijevala nemogućnost treće osobe da dođe do neovlaštenih informacija. Ideja je bila napraviti algoritam koji bi osigurao da poruka bude nerazumljiva svima osim pošiljatelju i primatelju. Kriptografija se javila već u Egiptu (za ukrašavanje grobnica faraona), Mezopotamiji te kod Asiraca i Babilonaca. Spartanci su u 5. stoljeću prije Krista upotrebljavali napravu za šifriranje zvanu skital. Ona je zapravo bila drveni štap oko kojeg se namota vrpca od pergamenta i okomito napiše poruka. Nakon upisivanja poruke, vrpca se odmota, a na njoj ostanu izmiješani znakovi koje može pročitati samo onaj koji ima štap jednake debljine. U Bibliji se, primjerice, koristila hebrejska šifra koja se zasnivala na principu zamjene prvog i zadnjeg slova abecede, a Julije Cezar je upotrebljavao šifre tako da slova abecede ciklički pomakne za dva mjesta. U 20.st. William Frederick Friedman uveo je pojam kriptoanalize. Za vrijeme drugog svjetskog rata njemački nacisti koristili su, danas vrlo poznati stroj, Enigmu, a njene šifre uspješno je razbio Alan Turing. Sve veći razvoj računala uvelike je utjecao na razvoj kriptografije. 1976. godine javila se ideja javnog ključa te godinu dana kasnije prvi, javnosti poznat, kriptosustav s javnim ključem - RSA kriptosustav. Potreba za tajnom komunikacijom je sve veća zbog svakodnevnog korištenja interneta, u bankarstvu te zbog kontrole pristupa općenito (ustanove, računala, podatci i sl.).

U ovom radu ću govoriti o RSA kriptosustavu, njegovoj implementaciji, sigurnosti iefikasnosti. U zadnjem poglavlju ću proučiti neke od ranijih napada na RSA kriptosustav kao što su napad pomoću modula, Håstadov i ciklički napad.

# 2. Kriptosustavi

Od davnih vremena do danas slale su se tajne poruke. Potreba za tajnom komunikacijom bila je najizraženija kod diplomacije i vojske. S razvojem tehnologije ta potreba je sve veća pa je danas prisutna svakodnevno. Primjer zato su bankovne transakcije. Veliki je interes napraviti poruke nedostupne svima osim primatelju.

## 2.1. Osnovni pojmovi

Disciplina koja proučava tajne sustave naziva se **kriptologija**. **Kriptografija** (grč. kryptos-skriven + grafo-pisati) je dio kriptologije koja se bavi proučavanjem metoda za slanje tajnih poruka tako da su one razumljive samo primatelju. Zadatak je omogućiti dvama subjektima (pošiljalac i primalac ili Alice i Bob) tajnu i nesmetanu komunikaciju u nesigurnom komunikacijskom kanalu tako da treći subjekt (Eve ili Oscar) ne razumije poruke. **Otvoren tekst** (eng. plaintext) nazivat ćemo tajnu pouku koju pošiljalac želi poslati primaocu, a kojeg je pošiljalac unaprijed transformirao pomoću ključa koji je oboma poznat. Taj postupak se naziva šifriranje. Poruka putuje komunikacijskim kanalom do primaoca i zove se **šifrat** (eng. ciphertext). Treći subjekt odnosno protivnik ne zna ključ i ne može dešifrirati poruku, ali primalac ga zna i u mogućnosti je dešifrirati poruku te tako dobiva otvoren tekst.

Znanstvena disciplina koja se bavi dešifriranjem poruka bez poznavanja ključa naziva se **kriptoanaliza** (grč. kryptos – skriven i analyein – odriješiti) . Kriptografija i kriptoanaliza zajedno čine kriptologiju.

Matematička funkcija koja služi za šifriranje i dešifriranje zove se **kriptografski algoritam**. Kriptografski algoritam zajedno s otvorenim tekstovima, šifratima i ključevima čini **kriptosustav**.

**Definicija 2.1.** *Kriptosustav je uređena petorka* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *za koju vrijedi:*

*(i)* $\mathcal{P}$ *je konačan skup svih mogućih osnovnih elementa otvorenog teksta;*

*(ii)* $\mathcal{C}$ *je konačan skup svih mogućih osnovnih elemenata šifrata;*

*(iii)* $\mathcal{K}$ *je prostor ključeva, tj. konačan skup svih mogućih ključeva;*

*(iv) Za svaki* $K \in \mathcal{K}$ *postoji funkcija šifriranja* $e_K \in \mathcal{E}$ *i odgovarajuća funkcija dešifriranja* $d_K \in \mathcal{D}$. *Pritom su* $e_K : \mathcal{P} \to \mathcal{C}$ *i* $d_K : \mathcal{C} \to \mathcal{P}$ *funkcije sa svojstvom da je* $d_K(e_K(x)) = x$ *za svaki otvoreni tekst* $x \in \mathcal{P}$.

Klasifikacija kriptosustava:

1. Tip operacija koje se koriste pri šifriranju:
   (i) supstitucijske šifre - svaki element otvorenog teksta zamijeni elementom šifrata.
   (ii) transpozicijske šifre - elementi otvorenog teksta se permutiraju.
2. Način obrade teksta:
   (i) blokovne šifre - obrađuje se blok elemenata otvorenog teksta pomoću istog ključa.
   (ii) protočne šifre u kojima se obrađuju elementi otvorenog teksta, jedan po jedan, pomoću više ključeva u nizu koji se paralelno generira.

3. Vrsta ključa:

   (i) kriptosustavi s tajnim ključem ili simetrični kriptosustavi - pošiljalac i primalac izabiru tajni ključ i pomoću njega generiraju funkcije šifriranja i dešifriranja. Kako je poznat ključ za šifriranje lako je pronaći ključ za dešifriranje i obrnuto.

   (ii) kriptosustavi s javnim ključem ili asimetrični kriptosustavi - poznat je ključ za šifriranje, ali ne i ključ za dešifriranje. Njega nije moguće lako otkriti u kratkom vremenu.

# 3. RSA kriptosustavi

RSA kriptosustav je dobio ime po svojim tvorcima Ron Rivestu, Adi Shamiru i Leomard Adlemanu. Bio je prvi javnosti poznat kriptosustav s javnim ključem. Martin Grdner predstavio ga je 1977. u časopisu Scientific American, međutim cijeli znanstveni članak objavljen je godinu kasnije od strane njegovih izumitelja. Sigurnost RSA kriptosustava je u činjenici da je faktorizacija velikih prirodnih brojeva na produkt dva prosta broja teška i dugotrajna, a parametri su modul $n$ koji je produkt dva velika prosta broja $p$ i $q$ te eksponenti $e$ i $d$ koji se upotrebljavaju za šifriranje i dešifriranje.

**Definicija 3.1** (Eulerova funkcija)**.** *Funkcija koja prirodnom broju $n$ pridružuje broj $\varphi(n)$ koji predstavlja broj elemenata u nizu $1, 2, \ldots, n$ koji su relativno prosti s $n$ naziva se Eulerova funkcija.*

**Definicija 3.2** (Carmichaelova lambda funkcija)**.** *Najmanji broj $n$ takav da je*

$$a^{\lambda(n)} \equiv 1 \pmod{n},$$

*pri čemu je cijeli broj $a$ relativno prost s $n$ označavamo s $\lambda(n)$ i nazivamo Carmichaelova lambda funkcija.*

**Definicija 3.3.** *Funkciju $f : \mathbb{N} \to \mathbb{C}$ za koju vrijedi:*

*(i)* $f(1) = 1$

*(ii)* $f(ab) = f(a)f(b)$*, za sve $a, b$ takve da su $a$ i $b$ relativno prosti tj. je $(a, b) = 1$*

*nazivamo multiplikativna funkcija.*

**Definicija 3.4** (RSA kriptosustav)**.** *Neka je $N$ produkt dva velika prosta broja $p$ i $q$, $\mathcal{P} = \mathcal{C} = \mathbb{Z}_N$ (skup svih ostataka modul $N$) , te*

$$\mathcal{K} = \{(N, p, q, e, d) : ed \equiv 1 \pmod{\varphi(N)}\}$$

*gdje je $\varphi(N) = (p-1)(q-1)$ Eulerova funkcija. Za svaki $K \in \mathcal{K}$ deniramo pravilo kriptiranja $e_K : \mathbb{Z}_N \to \mathbb{Z}_N$*

$$e_K(x) = x^e \pmod{N}$$

*i pravilo dekriptiranja $d_K : \mathbb{Z}_N \to \mathbb{Z}_N$*

$$d_K(y) = y^d \pmod{N}$$

*za $x, y \in \mathbb{Z}_N$.*

Par $(e, N)$ je RSA javni ključ, a trojka $(d, p, q)$ je RSA tajni ključ.

Kako je Eulerova funkcija multiplikativna, a $p$ prost broj slijedi prema Definiciji 3.1 $\varphi(n) = p - 1$ pa $\varphi(n) = \varphi(pq) = \varphi(p)\varphi(q) = (p - 1)(q - 1)$.

**Teorem. 3.1** (Eulerov teorem)**.** *Neka je $a$ cijeli broj i $n$ prirodan broj. Ako su brojevi $a$ i $n$ relativno prosti, onda vrijedi $a^{\varphi(n)} \equiv 1 \pmod{n}$.*

**Dokaz.** Pokažimo sada da su funkcije $e_K$ i $d_K$ jedna drugoj inverzne.

Imamo: $d_K(e_K(x)) \equiv x^{de} \pmod{n}$.

Iz $de \equiv 1 \pmod{\varphi(n)}$ slijedi da postoji prirodan broj $m$ takav da je $de = m\varphi(n)+1$. Sada imamo

$$x^{de} = x^{(m\varphi(n)+1)} = (x^{\varphi(n)})^m x.$$

U ovisnosti od $n$ i $x$ promatramo dva slučaja:

1. $(x, n) = 1$. Prema Eulerovom teoremu je

$$x^{\varphi(n)} \equiv 1 \pmod{n}$$

iz čega dobijemo

$$x^{de} \equiv 1^m x \equiv x \pmod{n}.$$

2. $(x, n) \neq 1$. Ako je $(x, n) = n$ onda je

$$x^{de} \equiv x \equiv 0 \pmod{n}.$$

Promotrimo još $(x, n) = p$ ili $(x, n) = q$.

Bez smanjenja općenitosti, uzmemo $(x, n) = p$.

Tada je

$$x^{de} \equiv x \equiv 0 \pmod{p}$$

i

$$x^{de} = (x^{(q-1)})^{(p-1)m} x \equiv x \pmod{q},$$

iz čega je

$$x^{de} \equiv x \pmod{n}.$$

Slučaj $(x, n) = q$ je analogan. Zaista, $x^{de} \equiv x \pmod{n}$ pa je $d_K(e_K(x)) = x$. □

## 3.1. Implementacija RSA kriptosustava

Postupak generiranja ključeva za RSA kriptosustav:

1. Izabrati dva, po mogućnosti, što veća prosta broja $p_1$ i $p_2$.
2. Izračunati $n = p_1 p_2$ i $\varphi(n) = \varphi(p_1 p_2) = (p_1 - 1)(p_2 - 1)$.
3. Odabrati slučajan broj $e$ takav da je $e < \varphi(n)$ i $(\varphi(n), e) = 1$. Upotrebom proširenog Euklidovog algoritma izračunati $d$ takav da je $de \equiv 1 \pmod{\varphi(n)}$, tj. $d \equiv e^{-1} \pmod{\varphi(n)}$.
4. Staviti ključ za šifriranje $(n, e)$. Tajni ključ je $d$.

Računanje šifrata $y = x^e \pmod{n}$ naziva se modularno potenciranje. Ono se može izvesti vrlo efikasno metodom „kvadriraj i množi“ („binarne ljestve“). Računanje $x^e \bmod n$ algoritmom:

$y = 1$
za $i = m - 1, \ldots, 1, 0$
$y = y^2 \pmod{n}$
ako je $e_i = 1$ tada je $y = yx \pmod{n}$
gdje je $e = \sum_{i=0}^{m-1} e^i 2^i = e_0 + 2e_1 + \ldots + 2^{m-1} e_{m-1}$ binarni zapis broja $e$.

**Primjer 3.1.** *Primjenom RSA kriptosustava šifrirati poruku KRIPTOGRAFIJA.*

*Najprije prikažimo otvoreni tekst u obliku niza prirodnih brojeva gdje je* $A = 01, B = 02, \ldots, Z = 26$. *Dakle,* $KRIPTOGRAFIJA = 11180916201507180106091001$.

*Odabiremo proste brojeve* $p_1 = 13, p_2 = 17$. *Tada je* $n = p_1 \cdot p_2 = 221$ *i* $\varphi(n) = 192$. *Enkripcijski eksponent e mora biti relativno prost s* 192 *pa uzmimo da je* $e = 15$. *Pomoću Euklidovog algoritma izračuna se* $d = 13$ *iz* $15d \equiv 1 \pmod{192}$.

$e_K(x) = x^e \mod 221$
$x_1 = 111,\ e_K(x_1) = 111^{15} \mod 221 = 70$
$x_2 = 809,\ e_K(x_2) = 809^{15} \mod 221 = 131$
$x_3 = 162,\ e_K(x_3) = 162^{15} \mod 221 = 151$
$x_4 = 015,\ e_K(x_4) = 15^{15} \mod 221 = 8$
$x_5 = 071,\ e_K(x_5) = 71^{15} \mod 221 = 26$
$x_6 = 801,\ e_K(x_6) = 801^{15} \mod 221 = 213$
$x_7 = 060,\ e_K(x_7) = 60^{15} \mod 221 = 70$
$x_8 = 910,\ e_K(x_8) = 910^{15} \mod 221 = 104$
$x_9 = 010,\ e_K(x_9) = 10^{15} \mod 221 = 12$
*Šifrat :* 070131151008026213070104012

*Dešifriranje* $e_K(x)$ *se provodi pomoću* $d = 13$ *iz* $x = d_K(e_K(x)) = (x^e)^d \mod n$ *i dobiva se otvoreni tekst:* $KRIPTOGRAFIJA$.

# 4. Sigurnost RSA kriptosustava

Sigurnost RSA kriptosustava oslanja se na činjenicu da je funkcija šifriranja jednosmjerna funkcija. Napadač ne zna tajni ključ i pa ne može pronaći inverz funkcije šifriranja te ne može dobiti otvoreni tekst. Međutim, algoritam dešifriranja vezan je za faktorizaciju modulo $n = p_1p_2$. Ako se faktorizacija $n = p_1p_2$ može pronaći onda možemo izračunati privatni ključ $(d, p, q)$ za bilo koji javni ključ $(e, N)$. Stoga, možemo riješiti RSA problem za bilo koji javni ključ sa modulom $n$. Prema tome, važno je osigurati tajnost prostih brojeva $p_1$, $p_2$ kako bi onemogućili otkrivanje tajnog eksponenta $d$ i spriječili dešifriranje poruke.

## 4.1. Faktorizacija modulo $n$ i poznavanje tajnog eksponenta $d$

Ukoliko napadač zna faktorizaciju broja

$$n = p_1p_2$$

u mogućnosti je otkriti

$$\varphi(n) = (p_1 - 1)(p_2 - 1)$$

i pomoću Euklidovog algoritma znati tajni eksponent $d$. Kada napadač zna $d$ u mogućnosti je dešifrirati poruku i saznati otvoren tekst.

**Lema 4.1** (Vjerojatnosni algoritam)**.** *Poznavanje tajnog eksponenta d, koji odgovara javnim n i e, može omogućiti faktorizaciju od n.*

U RSA kriptosustavu uzimaju se $p_1$ i $p_2$ takvi da imaju sto znamenki, pa trenutno ne postoji algoritam koji bi dovoljno brzo napravio faktorizaciju od $n$. Dakle, problem računanja tajnog eksponenta $d$ iz javnog ključa i problem faktorizacije broja $n$ su računski ekvivalentni. Za generiranje $p_1$ i $p_2$ potrebno je voditi računa da faktorizacija $n = p_1p_2$ bude nemoguća. Također, ukoliko nam je poznata Eulerova funkcija $\varphi(n)$ može se uspješno faktorizirati modulo $n$ rješavajući sustav jednadžbi

$$n = p_1p_2$$

$$\varphi(n) = (p_1 - 1)(p_2 - 1)$$

za nepoznate proste brojeve $p_1$ i $p_2$. Zapravo, $p_1$ i $p_2$ su rješenja kvadratne jednadžbe

$$x^2 - (n - \varphi(n) + 1)x + n = 0.$$

Za dani $\varphi(n)$ može se uspješno faktorizirati modulo. Moguće je faktorizirati modulo poznavajući jedan od brojeva: $\varphi(n)$ ili $\lambda(n)$. Simmons je pokazao da je najveći zajednički djelitelj $(p_1 - 1, -p_2 - 1)$ jedini parni broj koji zadovoljava

$$\frac{n}{\lambda(n)} - 2 < (p_1 - 1, p_2 - 1) < \frac{n}{\lambda(n)}.$$

Stoga, za dane $\lambda(n)$ i $(p_1 - 1, p_2 - 1)$ možemo izračunati

$$\varphi(n) = (p_1 - 1, p_2 - 1)\lambda(p_1 - 1, p_2 - 1).$$

Neka je $n = p_1^{v_1} \cdot \cdots \cdot p_r^{v_r}$ faktorizacija u produkt prostih faktora nekog neparnog cijelog broja $n$. Gledamo tri funkcije

$$\varphi(n) = p_1^{v_1-1} \cdots p_r^{v_r-1}(p1-1)\cdots(p_r-1)$$

$$\lambda(n) = p_1^{v_1-1} \cdots (p_r^{v_r-1}(p_1-1), \ldots, (p_r-1))$$

$$\lambda'(n) = ((p_1-1), \ldots, (p_r-1)).$$

Koristeći se rezultatima Millera, algoritam se može konstruirati da faktor $n$ u polinomu vremena daje višekratnik $\lambda'(n)$. Pošto su i $\varphi(n)$ i $\lambda(n)$ višekratnici $\lambda'(N)$, slijedi da možemo faktorizirati $n$ bilo koje funkcije gore navedene. Dakle, znati višekratnik $\varphi(n)$ i $\lambda(n)$ je dovoljno da faktoriziramo RSA modul.

Sada, ukoliko je privatni eksponent poznat, iz jednadžbe

$ed = 1 + k\varphi(n)$,

gdje $\varphi(N)$ je jednak $\varphi(n)$ ili $\lambda(n)$, ovisno kako su javni i privatni eksponent definirani, slijedi da sa poznatim $d$ možemo izračunati

$ed - 1 = k\varphi(n)$.

Kako je $\varphi(n)$ višekratnik od $\lambda'(n)$, koristeći Millerove rezultate, moguće je faktorizirati modul. Stoga, ako nam je poznat privatni eksponent možemo slomiti RSA.

**Lema 4.2.** *Neka je $n = p_1p_2$ produkt dva različita prosta broja. Ako znamo $n$ i $\varphi(n)$, tada možemo brzo naći $p_1$ i $p_2$.*

**Dokaz.** Vrijedi

$$\varphi(n) = (p_1-1)(p_2-1),$$

Tada je

$$\varphi(n) = p_1p_2 - p_1 - p_2 + 1 = n - p_1 - p_2 + 1 = n - (p+q) + 1.$$

Sada je

$$n - \varphi(n) + 1 = p + q$$

pa nam je poznata i suma $p_1 + p_2$. Definiramo polinom

$$f(X) = (X - p_1)(X - p_2) = X^2 - (p_1 + p_2)X + p_1p_2$$

$$f(X) = X^2 - (n - \varphi(n) + 1)X + n.$$

Formula za rješavanje kvadratne jednadžbe:

$$p_1, p_2 = \frac{(n - \varphi(n) + 1) \pm \sqrt{(n - \varphi(n) + 1)^2 - 4n}}{2}.$$

□

**Primjer 4.1.** *Neka je $n = 2773$, a $\varphi(n) = 2668$. Odredi $p_1$ i $p_2$ tako da je $n = p_1p_2$.*

*Izračunamo*

$$n - \varphi(n) + 1 = 2773 - 2668 + 1 = 106,$$

*pa je*

$$f(x) = x^2 - 106x + 2773.$$

*Pomoću formule za rješenja kvadratne jednadžbe dobijemo $p_1 = 47$ i $p_2 = 59$.*

## 4.2. Kriptoanaliza RSA

Postoji mnogo različitih vrsta napada na RSA. Primjerice, postoje tzv. side-channel napadi, koji iskorištavaju fizikalna svojstava uređaja na koji je RSA implementiran.

Druge vrste napada se fokusiraju na ljudsku komponentu zaštite. Ovdje je dio informacija koje su uzete od korisnika nekom vrstom manipulacije. Primjera radi, zaporka koja osigurava RSA privatni ključ se može otkriti nazvavši osobu u sred noći pomahnitalim glasom, tvrdeći kako se dogodio hitan slučaj na poslu i kako je potrebna zaporka. U tzv. napadima gumenog crijeva, dio informacija se može izvući silom ili prijetnjom silom.

Napadi koje mi uzimamo u obzir su nisu iz fizičkog dijela koje koristi RSA. Napadi su bazirani na matematičkoj strukturi RSA kriptosustava (forma modula ili ključ jednadžbe) i iskorištavanje određenih parametara izbora (kao što su korištenje malih javnih ili privatnih eksponenata).

## 4.3. Homomorfno svojstvo RSA kriptosustava

RSA ima multiplikativno svojstvo da šifrirani tekst odgovara otvorenom tekstu $m = m_1 m_1 \pmod{n}$. Ova se svojstva često naziva homomorfno svojstvo RSA.

Pretpostavimo da je protivniku dan šifrirani tekst $c = m^e \mod n$ i želi izračunati $m$. Odabirom slučajnog $x \in \mathbb{Z}n$, protivnik traži otvoreni tekst šifrata $c_0 = cx^e \mod n$. Budući da traženi otvoreni tekst $m_0$ zadovoljava

$$m_0 = c_0^d \mod n = (cx^e)^d \mod n = c^d x^{ed} \mod n = mx \mod n$$

protivnik, s obzirom na $m_0$, može jednostavno izračunati $m = m_0 x^{-1} \mod n$ kako bi otkrio željeni tekst. Još jedan napad koji koristi homomorfno svojstvo RSA je Boneh, Joux i Nguyen. Njihov napad koristi činjenicu da se u praksi RSA uglavnom koristi za šifriranje kratkih poruka. U stvari, njihov napad pretpostavlja da željeni $l$-bitni tekst se može faktorizirati na dva $l/2$ – bitna faktora $m_1$ i $m_1$ ($m = m_1 m_1$). Napad počinje izradom tablice koja sadrži svaki $l/2$-bitni broj $m_1'$ i njegovu enkripciju $(m_1')^e \mod n$. Zatim, za svaki mogući $l/2$-bitni broj $m_1'$, vrijednost $c(m_1')^{-e} \mod n$ se provjerava prema svakom šifriranju u tablici. Kada je $m_1' = m_1$ slijedi

$$c(m_1)^{-e} \mod n = (m_1 m_1)^e (m_1)^{-e} \mod n = (m_1)^e \mod n,$$

i nalaziti će se u tablici. Kada se pronađe podudaranje, otkriva se faktorizacija otvorenog teksta $m$. Napad treba izračunati $2^{\frac{\ell}{2+1}}$ modularnih potencija, pohraniti $2^{\frac{\ell}{2}}$ $\ell$ parova brojeva i uspjeti s vjerojatnošću 18%. Ti se napadi lako izbjegavaju postavljanjem neke strukture na otvoren tekst.

## 4.4. Semantička zaštita

Semantički siguran kriptosustav je kriptosustav u kojemu se nijedna informacija o otvorenom tekstu danog šifrata (i poznatog javnog ključa) ne može odrediti s nezanemarivom vjerojatnošću.

RSA kriptosustav očito nije semantički siguran kriptosustav. Posebice, bilo koji deterministički kriptosustav ne može biti semantički siguran. Dana su nam dva otvorena teksta i šifrat jednog od njih, protivnik može odrediti koji otvoren tekst odgovara šifriranom tekstu (jednostavno šifriranjem otvorenog teksta i usporedbom).

Dodatno, jednostavno se pokaže da Jacobijev simbol otvorenog teksta (i modula) otkriva Jacobijev simbola šifriranog teksta (i modula). Posebno, može se pokazati da je

$$\left(\frac{c}{n}\right) = \left(\frac{m}{n}\right),$$

za svaki otvoren tekst $m$ i njegov odgovarajući šifrat $c$. Znači, neke informacije o otvorenom tekstu su otkrivene zato što su dani šifrirani tekst i javni ključ.

# 5. Efikasnost RSA kriptosustava

Ukratko ćemo razmotriti efikasnost RSA kriptosustava. Razmatrat ćemo troškove za šifriranje odnosno dešifriranje algoritma.

## 5.1. Generiranje prostih brojeva

Algoritam za generiranje ključa za RSA kriptosustav mora generirati dva nasumična prosta broja koji su svaki otprilike jednake veličine. Koristeći se Miller-Rabin testom možemo generirati $n$-bitni nasumični (vjerojatni) prost broj s očekivanim vremenom izvođenja

$$\mathcal{O}\left(\frac{n^4}{\log(n)} + tn^3\right).$$

Metoda ima pogrešan output koji je složeni broj umjesto prost broj s vjerojatnošću najviše $4^{-t}$. Ova složenost pretpostavlja jednostavnu kvadratnu aritmetiku i može se unaprijediti koristeći se bržim metodama množenja, ali složenost je najmanje $\mathcal{O}(n^2)$ čak i sa najbržim znanim metodama. Za veće module, ove veličine mogu biti skupe operacije, osobito ako se dosta prostih brojeva mora generirati.

Postoji mnogo brzih algoritama za generiranje prostih brojeva, ali nijedan nije efikasniji od prethodno navedenih.

## 5.2. Modularno potenciranje

Enkripcija i deskripcija u RSA počiva na modularnom potenciranju. Ove operacije mogu biti veoma skupe kada su eksponenti i moduli veliki. Za $b$-bitni eksponent $B$ i $n$-bitni modul $n$, uzme se u obzir

$$X^B \mod n,$$

za neki $X \in \mathbb{Z}_n$. Postoji mnogo različitih algoritama za modularno potenciranje, ali, u srži, složenost ovih izračuna se može smanjiti na brojanje broja modularnih množenja. Primjerice, korištenje standardne kvadriraj i množi metode zahtjeva otprilike $b$ množenja i $wt(B)$ kvadriranja, gdje je $wt(x)$ broj jedinica u binarnom prikazu broja $x$. Kada je broj jedinica i nula u binarnom prikazu $B$ otprilike jednak, to dovodi do otprilike ukupno $\frac{3}{2}b$ množenja. U globalu, broj modularnih množenja je linearan u dužini bitova eksponenta, osim ako se ne dopusti eksponencijalan broj pred-izračunavanja i pohrane za ove vrijednosti. Mi ćemo pretpostaviti, zbog jednostavnosti, da je očekivan broj množenja $\frac{3}{2}b$. Koristeći ovo pojednostavljenje, kada uspoređujemo troškove izračunavanja dva modularna potenciranja sa istim modulima, $X^B \mod n$ i $Y^A \mod n$, očekujemo da omjer troška (ili vrijeme izvođenja) bude jednak omjeru dužine bita eksponenta. Ako $A$ je $a$-bitni cijeli broj i $B$ je $b$-bitni cijeli broj, onda $a/b$ je očekivani omjer vremena izvođenja za ove potencije.

Kada uspoređujemo troškove modularnog potenciranja sa drugačijim veličinama modula, troškovi množenja se moraju uzeti u obzir. Neka $M(n)$ bude složenost općeg modularnog množenja sa $n$-bitnim multiplikatorima i $n$-bitnim modulom. Ovdje, $M(n)$ može predstavljati broj bitnih operacija ili riječ koju operacije trebaju. Očekivani trošak izračunavanja $X^B \mod n$ je

$$\frac{3}{2}bM(n).$$

Ovisno o detaljima implementacije, složenost $M(n)$ može varirati između linearne i kvadratne. U najgorem slučaju $M(n)$ je kvadratna u $n$, što odgovara klasičnoj kvadratnoj

aritmetici. Tipično je u literaturi pretpostaviti ovu jednostavnu kvadratnu složenost. Postoje brže metode poput Karatsubove metode množenja, koja ima složenost $n^{(\log_2(3))} \approx n^{1.59}$, i metode koje približavaju linearnu složenost. U praksi, izbor metoda množenja ovisi o veličini modula. Brže metode množenja generalno zahtijevaju više općeg i zapravo nisu brže dok se ne koriste jako veliki moduli. Kao rezultat, u uspoređivanju složenosti dva modularna potenciranja sa drugačijim veličinama modula, opseg veličine modula i specifičan algoritam korišteni za implementaciju moraju biti unaprijed poznati.

# 6. Raniji napadi na RSA kriptosustav

U ovom poglavlju upoznat ćemo neke od ranije poznatih napada na RSA. Nekolicina ovih napada su primjeri pogreški protokola. Pogreška protokola se događa kada se sigurnost kriptosustava ne koristi pravilno, što rezultira pogreškom željenih sigurnosnih ciljeva tog protokola. Sve pogreške protokola ovdje dopuštaju protivniku da izračuna otvoreni tekst nekoliko šifriranih tekstova, bez da moramo slomiti RSA.

## 6.1. Uobičajeni modul napadi

Uobičajeni protokol modula je bio raniji prijedlog u kojemu bi središnje ključno tijelo (npr. provjerena treća strana) generiralo RSA modul i distribuiralo valjane parove ključeva, sve sa jednakim modulom, korisnicima unutar sustava. Korisnikov privatni ključ u ovome protokolu je $(d, n)$, tako da faktorizacija modula nije poznata korisniku. Namjera je bila da samo središnje ključno tijelo ima znanje o faktorizaciji uobičajenih modula.

Godine 1983. Simmons je pokazao da pogreška protokola postoji kada se isti otvoren tekst šifrira s dva različita javna ključa koji imaju jednake module i relativno proste javne eksponente. Obzirom na dva šifrirana teksta i dva javna ključa, pokazao je da se lako može izračunati otvoreni tekst. Neka su $(e_1, n)$ i $(e_2, n)$ dva valjana RSA javna ključa sa relativno prostim javnim eksponentima. Pošto su $e_1$ i $e_2$ relativno prosti, možemo jednostavno izračunati cijele brojeve $a_1$ i $a_2$ tako da $a_1e_1 + a_2e_2 = 1$. Za bilo koji otvoreni tekst $m$, obzirom da $c_1 = m^{e_1} \mod n$ i $c_2 = m^{e_2} \mod n$, otvoreni tekst se obnavlja jednostavno izračunavajući $c_1^{a_1}c_2^{a_2} \mod n$, pošto izračunavajući u $\mathbb{Z}_n$ imamo

$$c_1^{a_1}c_2^{a_2} = m^{a_1e_1}m^{a_2e_2} = m^{a_1e_1+a_2e_2} = m.$$

Ovaj napad se može izvršiti bilo tko tko ima pristup javnim ključevima i promatrao je dva šifrirana teksta.

Godine 1984. DeLaurentis je pokazao da je protokol bio potpuno nesiguran. On je pokazao da znanje bilo kojega para javnog i privatnog ključa bi bio dovoljan za izračunavanje valjanog privatnog ključa za bilo koji drugi javni ključ sa istim modulom. Mi ponovno utvrđujemo ovaj rezultat u sljedećem teoremu.

**Teorem. 6.1.** *Neka je $(e, n)$ valjani RSA javni ključ sa odgovarajućim privatnim ključem $(d, n)$, i neka je $(e_1, n)$ još jedan valjani javni ključ tako da $e_1 \neq e$. Obzirom na $e$, $d$, $n$ i $e_1$, valjani dešifrirani eksponent za javni ključ $(e_1, n)$ dan s*

$$d_1 = e_1^{-1} \mod \frac{ed-1}{(e_1, ed-1)},$$

*se može izračunati u polinomijalnom vremenu* $\log(n)$.

**Dokaz.** Ključna jednadžba za već poznati par javni-privatni ključ se može napisati u obliku

$$ed - 1 = k\lambda(n),$$

gdje je $k$ neki pozitivni cijeli broj. Pošto je $e_1$ valjani javni eksponent on mora zadovoljiti

$$(e_1, \lambda(n)) = 1 \text{ i } (e_1, k\lambda(n)) = k',$$

za neki cijeli broj $k'$ koji zadovoljava $k'k$. Dopuštajući $\tilde{k} = k/k'$, imamo

$$\frac{ed-1}{(e_1, ed-1)} = \frac{k\lambda(n)}{k'} = \tilde{k}\lambda(n),$$

tako da privatni eksponent $d_1$ zadovoljava

$$e_1 d_1 = 1 + k_1(\tilde{k}\lambda(n)),$$

za neke pozitivne cijele brojeve $k_1$. Stoga,

$$e_1 d_1 \equiv 1 \pmod{\lambda(n)}$$

tako da je $d_1$ valjani privatni eksponent za javni ključ $(e_1, n)$. Pošto se sva izračunavanja mogu izvršiti u polinomijalnom vremenu $\log(n)$, rezultat slijedi. □

Dodatno, koristeći ideju koja se pripisuje Simmonsu, DeLaurentis je pokazao da ako nam je poznat par javni-privatni ključ, modul se može faktorizirati mnogočlanim vremenskim Las Vegas algoritmom. Obzirom na $e$ i $d$, jednostavno izračunavajući višekratnik $\varphi(n)$, znajući $ed - 1 = k\varphi(n)$, primjeni se rezultat Millera (koji može faktorizirati $n$ poznatim višekratnikom $\varphi(n)$). Stoga, protokol je sasvim nesiguran. Bilo koji korisnik u sustavu može jednostavno faktorizirati modul koristeći samo njihov vlastiti par javni-privatni ključ.

Kao rezultat ovih napada, jasno je da svaki RSA modul treba biti poznat samo jednome korisniku.

## 6.2. Håstadov napad

Još jedna pogreška protokola se pojavljuje kada se nekolicina srodnih poruka otvorenog teksta šifrira s malim javnim eksponentom i različitim modulima. Kolektivno, napadi na ove pogreške protokola su nazvani Håstadov napad. Uzimamo u obzir dvije vrste napada: napadi uobičajenog otvorenog teksta i napadi srodnih poruka.

### 6.2.1. Napadi uobičajenog otvorenog teksta

Pogreška protokola nastupa kada se otvoreni tekst $m$ šifrira s nekoliko javnih ključeva $(e, n_i)$, od kojih svaki ima jednak javni eksponent $e$ i različiti modul $n_i$. Napad na ovu pogrešku protokola prvi puta objavio je Håstad 1985. godine. Napad navodimo u sljedećem teoremu.

**Teorem. 6.2.** *Neka je $(e, n_1), \ldots, (e, n_l)$, $l \geq e$, valjani RSA javni ključevi i po parovima relativno prosti moduli, neka je $n_0 = \min(n_1, ..., n_l)$ i neka je $n = \prod_{i=1}^{l} n_i$. Za bilo koju poruku otvorenog teksta $m < n_0$, obzirom da $c_i = m^e \mod n_i$ i $(e, n_i)$ za $i = 1, \ldots, l$ otvoreni tekst $m$ se može izračunati u polinomijalnom vremenu* $\log(n)$.

**Dokaz.** Pošto su moduli po parovima relativno prosti, koristimo Kineski teorem o ostatcima za određivanje

$$C \equiv m^e \pmod{n}$$

koristeći $c_i$ i $n_i$ (za $i = 1, \ldots, l$) kao unos. Pošto $m < n_0$, slijedi

$$m^e < n_1 n_2 \cdots n_l = n,$$

i tako je

$$C = m^e.$$

Izračunavajući $e$-ti korijen od $C = m^e$ preko cjelobrojnog otvorenog teksta $m$. Pošto se sva izračunavanja mogu izvršiti u polinomijalnom vremenu $\log(n)$, rezultat slijedi. □

### 6.2.2. Napadi srodnih otvorenih tekstova

Posljednji propust u protokolu koji razmatramo pojavljuje se kada se nekoliko povezanih otvorenih tekstova šifrira s malim javnim eksponentima i različitim modulima. U ovom kontekstu, otvoreni tekst $m_i$ je povezan ako je $m_i = f_i(m)$ za neke (poznate) polinome $f_i$. Ovdje je $m$ jedini nepoznat dio svakog otvorenog teksta i bit će naveden kao otvoren tekst.

Predstavit ćemo taj napad, ponekad znan kao jaki Håstadov napad, u sljedećem teoremu.

**Teorem. 6.3.** *Neka su $(e, n_1), ..., (e, n_l)$ valjani RSA javni ključevi i po parovima relativno prosti moduli, neka je $n_0 = \min\{n_1, ..., n_l\}$ i neka je $n = \prod_{i=1}^{l} n_i$. Neka su $f_i(x) \in \mathbb{Z}_{n_1}[x], \ldots, f_l(x) \in \mathbb{Z}_{n_l}[x]$ poznati polinomi. Za svaki otvoreni tekst $m < n_0$ , ako je $l \geq \max_i\{e_i \deg(f_i(x))$ tada za dani $c_i = f_i(m)^{e_i} \mod n_i$ i $(e_i, n_i)$ za $i = 1, \ldots, l$ otvoreni tekst $m$ se može izračunati u polinomijalnom vremenu* $\log(n)$.

**Dokaz.** Bez smanjenja općenitosti, pretpostavimo da je $f_i(x)$ normirani polinom. Ako $f_i(x)$ nije normiran polinom, možemo ga jednostavno pomnožiti inverznim vodećim članom modulo $n_j$. Ako inverz ne postoji, faktorizacija od $n_j$ je otkrivena i dopušta nam jednostavno dešifriranje $c_i$ za otkrivanje $m$.

Neka je $\delta = \max_i\{e_i \deg(f_i(x))\}$. Za $i = 1, \ldots, l$ definiramo stupanj $\delta$ normiranog polinoma

$$g_i(x) = x_i^h(f_i(x)^{e_1}) - c_i) \in \mathbb{Z}_{n_i},$$

gdje je $h_i = \delta - \deg(f_i(x)^{e_i})$. Primjetimo da svaki od ovih polinoma zadovoljava

$$g_i(m) \equiv 0 \pmod{n_i}.$$

Kako su moduli u parovima relativno prosti, koristimo Kineski teorem o ostatcima za računanje stupnja $\delta$ normiranog polinoma $G(x) \in \mathbb{Z}_{n_i}[x]$ koristeći $g_i(x)$ i $n_i$ kao input. Novi polinom zadovoljava

$$G(m) \equiv 0 \pmod{n}$$

Gdje je $m < n_0 < n^{1/l} < n^{1/D}$. Koristeći Coppersmithovu metodu za invarijantne polinome možemo izračunati $m$. Kako se sva izračunavanja mogu izvršiti u polinomijalnom vremenu $\log(n)$, rezultat slijedi. □

**Teorem. 6.4** (Coppersmithova metoda za polinome)**.** *Neka je $n$ broj sa nepoznatom faktorizacijom čiji je djelitelj $b \geq n^{\beta}$. Neka je $f_b(x)$ normirani polinom stupnja $d$ i neka je $c > 1$ konstantan. Svi $x_0$ koji zadovoljavaju*

$$f_b(x_0) \equiv 0 \pmod{b}$$

*i*

$$|x_0| \leq cn^{\beta^2/d},$$

*mogu se pronaći u polinomijalnom vremenu* $\log(n)$.

## 6.3. Ciklički napadi

Posljednji od napada koje ćemo promatrati su ciklički napadi. Godine 1997. Simmons i Norris su primjetili da se otvoreni tekst može dobiti opetovanim dešifriranjem šifrata sve dok se ciklus ne vrati originalnom šifratu. Dan je šifrat $c = m^e \mod n$ i javni ključ $(e, n)$, ako je nakon $l + 1$ dešifriranja šifrat otkriven, tada je

$$c^{e^{l+1}} \equiv c \pmod{n},$$

tada slijedi

$$c^{e^l} \equiv m \pmod{n}.$$

Tada je otvoreni tekst otkriven nakon $l$ dešifriranja. Izvorni ciklički napad služio je kako bi se našao najmanji $l$ takav da je otvoreni tekst otkriven. Ova najmanja vrijednost od $l$ se naziva eksponent za oporavak (recovery exponent) otvorenog teksta $m$. Primijetimo da će uvijek postojati poruke otvorenog teksta koje će imati jako mali eksponent za oporavak. Naprimjer, otvoreni tekst $m = \pm 1$ ima $l = 1$ eksponent za oporavak ($e$ je neparan pa slijedi $c = m$).

**Teorem. 6.5.** *Neka je $(e, n)$ valjani RSA javni ključ. Za svaki otvoren tekst $m \in \mathbb{Z}_n^*$ eksponent za oporavak dijeli $\lambda(\lambda(n))$.*

Teorem implicira da je $\lambda(\lambda(n))$ najveći mogući eksponent za oporavak. Stoga, ako izaberemo dovoljno mali $\lambda(\lambda(n))$, napad je izvediv za sve otvorene tekstove. Nadalje, Friedlander, Pomerance i Shparlinski pokazuju da za gotovo sve izbore u prostih brojeva i javnih eksponenata, osim zanemarive vrijednosti otvorenog teksta, imati će eksponent za oporavak $l > n^{1-\varepsilon}$, za neki mali $\varepsilon$. Stoga, za dovoljno veliki $n$, ciklički napad je neisplativ.

Godine 1979. Williams i Schmidt generalizirali su ciklički napad tako da se traži ciklički modulo $p$ umjesto modulo $n$ kako je izvorno radila metoda. Jedan način traženja najmanjeg $k$ koji zadovoljava

$$G = (c^{e^k} - c, n) > 1.$$

Ako je $1 < g < n$, tada je pronađen ciklički modulo $p$ ili $q$ i $g$ otkriva faktorizaciju modula ( $g = p$ ili $g = q$).

Ako je $g = n$, tada je ciklički modulo $n$ pronađen kao i u originalnom napadu i $c^{e^{k-1}} \equiv m \pmod{n}$.

Efikasnost modificiranog napada očito ovisi o veličini $k$. Ako je prost broj $p$ izabran tako da $\lambda(\lambda(p)) = \lambda(p-1)$ ima samo mali prosti faktor ili je sam dovoljno mal, tada ciklički modulo $p$ može biti pronađen s relativno malim $k$ (slično za prosti $q$). Očekivano je svi osim jako malih brojeva otvorenog teksta će imati $k > n^{\frac{1}{2}-\varepsilon}$, kada su prosti brojevi izabrani slučajno. Stoga je modificirani ciklički napad neisplativ za dovoljno velike slučajne proste brojeve.

# Literatura

[1] M. J. Hinek, Cryptanalysis of RSA and Its Variants, Chapman & Hall/ CRC cryptography and network security, Boca Raton, 2009.

[2] M. Jović, Napadi na RSA kriptosustav s malim tajnim eksponentom, Diplomski rad, Osijek, 2016.

[3] I. Matić, Uvod u teoriju brojeva, Odjel za matematiku Sveučilišta J. J. Strossmayera, Osijek, 2015.

[4] K. H. Rosen, Elementary Number Theory and Its Applications, Addison-Wesley Publishing Company, Massachusetts, 1992.