Sigurnost u računarskim mrežama

Kriptologija
(prvi deo)

Nemanja Maček

Sadržaj

- Osnovni pojmovi
- Teorijska i praktična sigurnost
- Pojam perfektne šifre
- Klasična kriptografija
- Simetrični blokovski algoritmi
- Generatori slučajnih i pseudoslučajnih brojeva
- Šenonov ideal i pseudo-ideal
- Testovi slučajnosti
- Sekvencijalno šifrovanje
- Dodatak beleškama: matematičke osnove kriptografije (izvod)

Fysis kriptesthai filei

- „**Priroda voli da se skriva**.“
  Heraklit, Fragment 123.
- „**Samopokazujuće izrastanje (biće u celini) voli da se krije**.“
  Heideger, Uvod u metafiziku.

* Slika i navodi preuzeti iz predavanja prof. dr Milana Milosavljevića

Towards a Grand Strategy for an Uncertain World

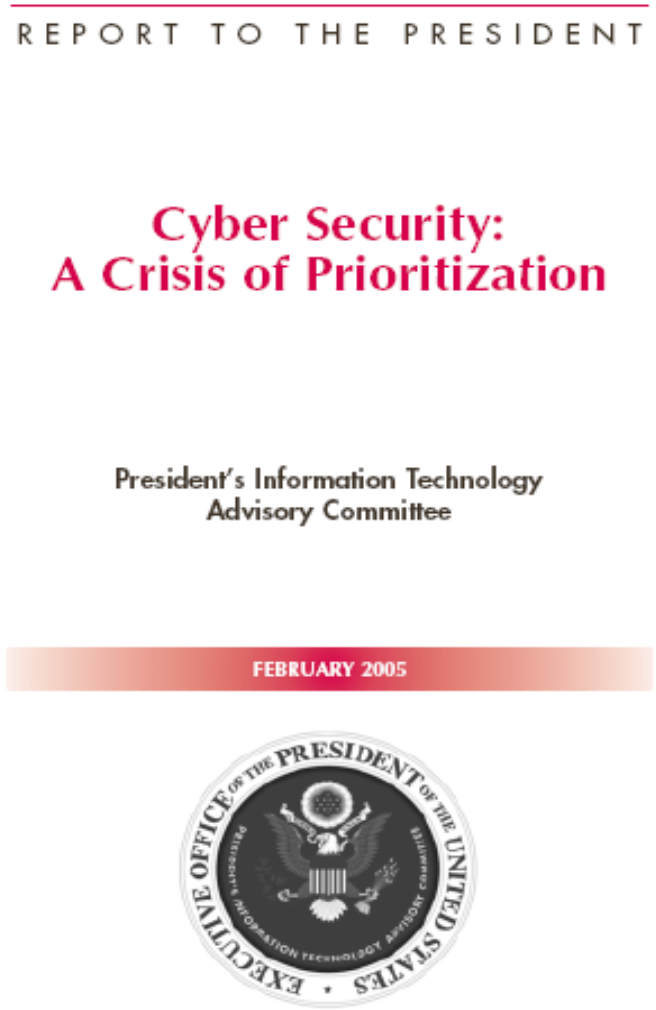
- *Renewing Transatlantic Partnership, by*
  - *General (ret.) Dr. Klaus Naumann, KBE, former Chief of the Defence Staff Germany, former Chairman Military Committee NATO*
  - *General (ret.) John Shalikashvili, former Chairman of the Joint Chiefs of Staff of the United States of America, former NATO Supreme Allied Commander in Europe*
  - *Field Marshal The Lord Inge, KG, GCB, PC, former Chief of the Defence Staff United Kingdom*
  - *Admiral (ret.) Jacques Lanxade, former Chief of the Defence Staff France, former Ambassador*
  - *General (ret.) Henk van den Breemen, former Chief of the Defence Staff the Netherlands*
  - *with Benjamin Bilski and Douglas Murray*
- *„Protection means taking all necessary reactive steps, including setting up **missile defence** and **cyber protection**, to prevent an enemy inflicting damage on the nation or alliance.“*
- *„In addition, there are certain other areas in which pre-delegation of a response capability will be necessary to protect NATO, where we cannot wait for the NATO Council to decide on a course of action, such as the acute crisis of a **missile attack** or **cyber attack**.“*

Kako Sjedinjene Američke Države vide kriptologiju?

- Zakon o izvozu oružja **ITAR** (*International Traffic in Arms Regulations*)
  - Strogo zabranjuje izvoz jakog kriptografskog softvera i algoritama svrstavajući ga u istu kategoriju sa delovima borbenih aviona, kao i sa klasičnim, hemijskim i biološkim naoružanjem.
- *USA Senate Bill 266*: predlog zakona koji „nije usvojen“:
  - *„It is the sense of Congress that providers of electronic communications services and manufacturers of electronic communications service equipment shall insure that communications systems permit the Government to obtain the plain text contents of voice, data, and other communications when appropriately authorized by law.“*

Kako Sjedinjene Američke Države vide kriptologiju?

- Da bi razumeli zašto je *Senate Bill 266* predložen, neophodno je izdvojiti tri ključne rečenice dokumenta *Report to the President* (2005):
  - *„Although current technical approaches address some of our immediate needs, they do not provide adequate computer and network security.“*
  - *„Fundamentally different architectures and technologies are needed so that the IT infrastructure as a whole can become secure.“*
  - *„At U.S. academic institutions today, the PITAC estimates, there are fewer than 250 active cyber security or cyber assurance specialists, ...“*



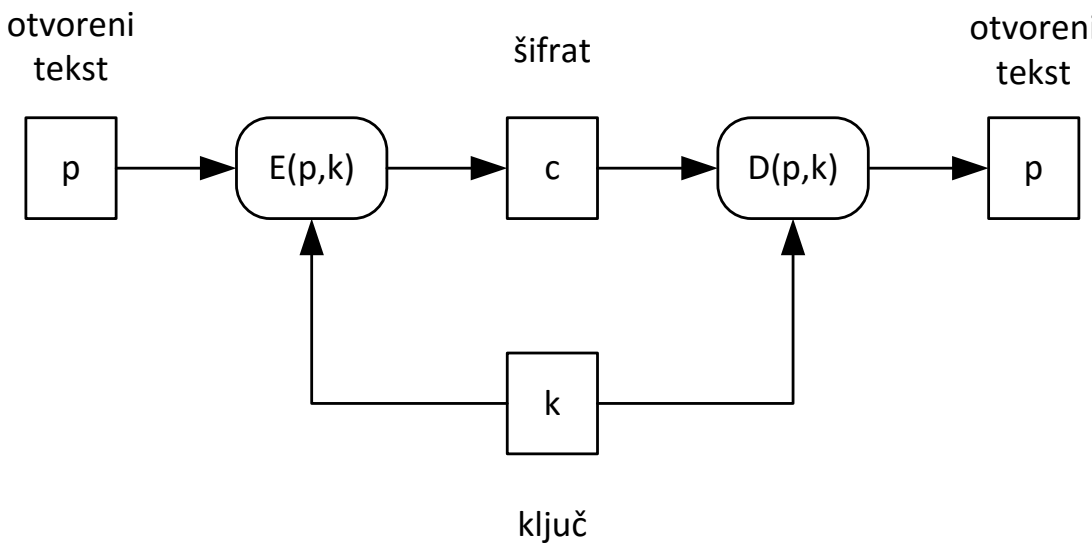
Osnovni pojmovi

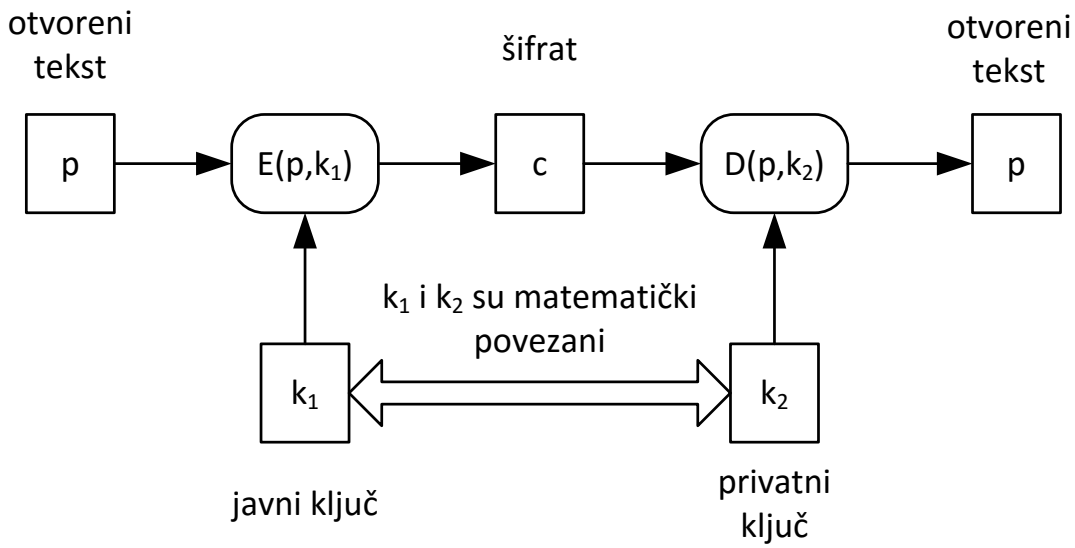
- **Algoritam za šifrovanje** je serija precizno definisanih koraka koji poruku – otvoreni tekst (engl. *plaintext*) uz pomoć ključa pretvara u šifrovanu poruku, tj. šifat (engl. *ciphertext*).
  - Procedura šifrovanja varira u zavisnosti od ključa koji menja neke detalje algoritma.
- Šifrat sadrži sve informacije koje sadrži i originalna poruka, ali nije u obliku čitljivom za čoveka ili računar bez primene odgovarajućeg algoritma za dešifrovanje i ključa.
  - To znači da je šifrat je zaštićen od neovlašćenog pristupa (korisnik bez ključa nema pristup podacima), pa se može preneti preko nesigurnog kanala ili čuvati na disku.
- Drugim rečima:
  - Šifrovanje je proces transformisanja otvorenog teksta pomoću ključa u šifrat.
  - Dešifrovanje je proces tranformisanja šifrata pomoću ključa u otvoreni tekst.
- **Kerchoffsov princip**: algoritam za šifrovanje može se smatrati sigurnim ukoliko sigurnost šifrata zavisi samo od tajnosti ključa, ali ne i od tajnosti algoritma.

Osnovni pojmovi

- Algorimi za šifrovanje se mogu podeliti na simetrične algoritme i algoritme sa javnim ključem.
- **Simetrični algoritmi**.
  - Funkcija šifrovanja na osnovu ključa *k* i ulaznih podataka *p* proizvodi šifrat c.
  - Funkcija dešifrovanja na osnovu istog ključa i šifrata proizvodi originalnu poruku.
- **Algoritmi sa javnim ključem**.
  - Funkcija šifrovanja proizvodi šifrat *c* na osnovu **javnog ključa** (engl. *public key*) k_1 i otvorenog teksta *p*.
  - Funkcija dešifrovanja na osnovu **privatnog ključa** (engl. *private key*) k_2 i šifrata *c* proizvodi originalnu poruku p.
  - Javni ključ je poznat onim osobama s kojima korisnik želi da komunicira.
  - Privatni ključ je poznat samo korisniku koji je ovlašćen da dešifruje poruke.
  - Privatni i javni ključ su matematički povezani, ali se privatni ključ ne može odrediti na osnovu javnog ključa.

Osnovni pojmovi



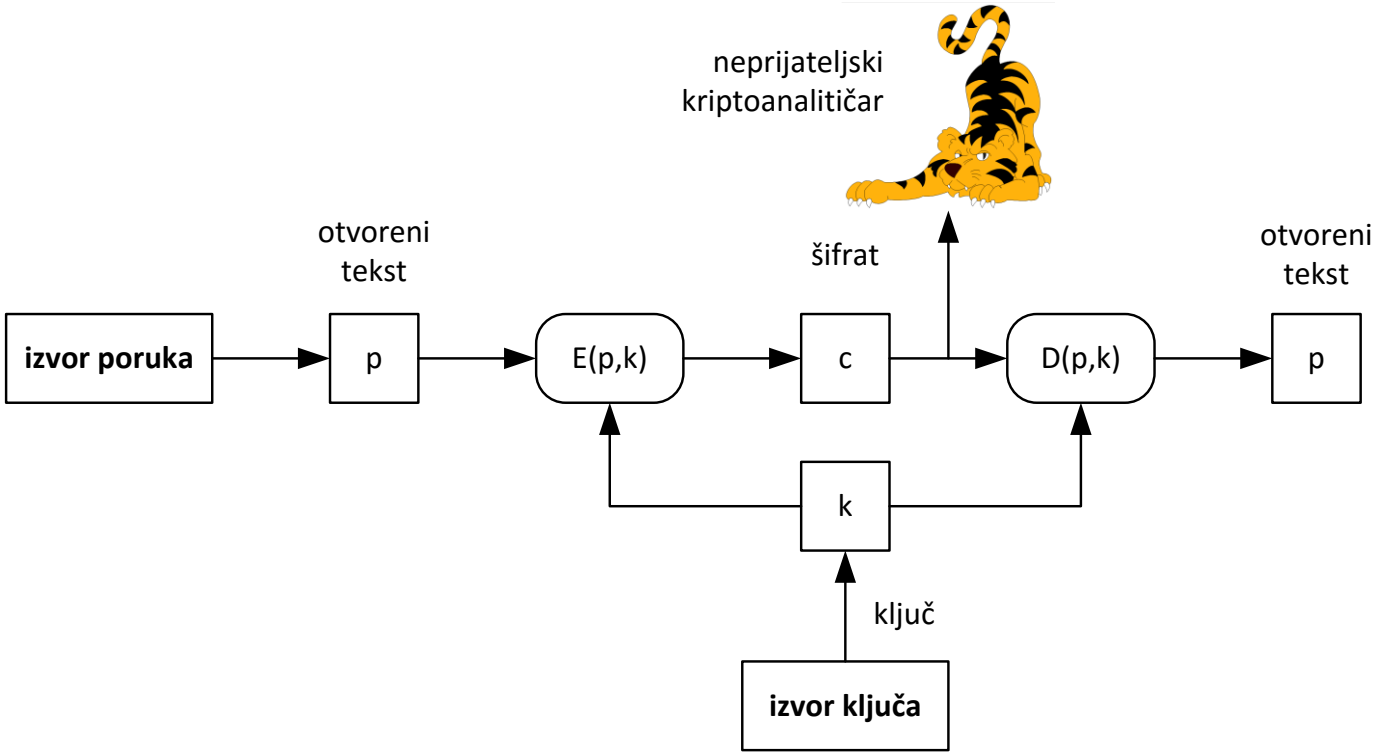


Osnovni pojmovi

- **Digitalni potpis** (engl. *digital signature*) je elektronska verzija potpisa, na osnovu kog se može identifikovati pošiljalac i dokazati verodostojnost poruke.
- Digitalni potpisi su usko povezani sa pojmom jednosmerne heš funkcije.
  - **Jednosmerna heš funkcija** (ili jednostavno, heš funkcija) na osnovu ulaznog podatka ma koje dužine proizvodi rezultujući niz tačno određene dužine – heš (engl. *hash*) koji, uslovno rečeno, jednoznačno identifikuje ulazni podatak.
- Proces potpisivanja i provere potpisa se odvija na sledeći način:
  - Pošiljaoc računa heš h_1 poruke p i potpisuje heš svojim privatnim ključem (uslovno se može shvatiti kao šifrovanje privatnim ključem), što rezultuje potpisom s_1.
  - Pošiljalac šalje originalnu poruku p i digitalni potpis s_1 primaocu.
  - Primalac određuje heš h_2 primljene poruke i proverava primljeni potpis s_1 javnim ključem pošiljaoca (uslovno se može shvatiti kao dešifrovanje javnim ključem).
  - Ukoliko je h_2 jednak rezultatu provere potpisa, identitet pošiljaoca je potvrđen.

Šenonova teorija

- Prvi korak u matematičkoj analizi kriptografije zahteva idealizaciju procesa komuniciranja na takav način da omogući definiciju kriptografskog sistema na matematički prihvatljiv način [5].
- Šenonova postavka problema tajnog komuniciranja: sa slike (tigar je dodat) se jasno vidi šta je Šenon podrazumevao pod napadom zasnovanim samo na poznavanju šifrata.

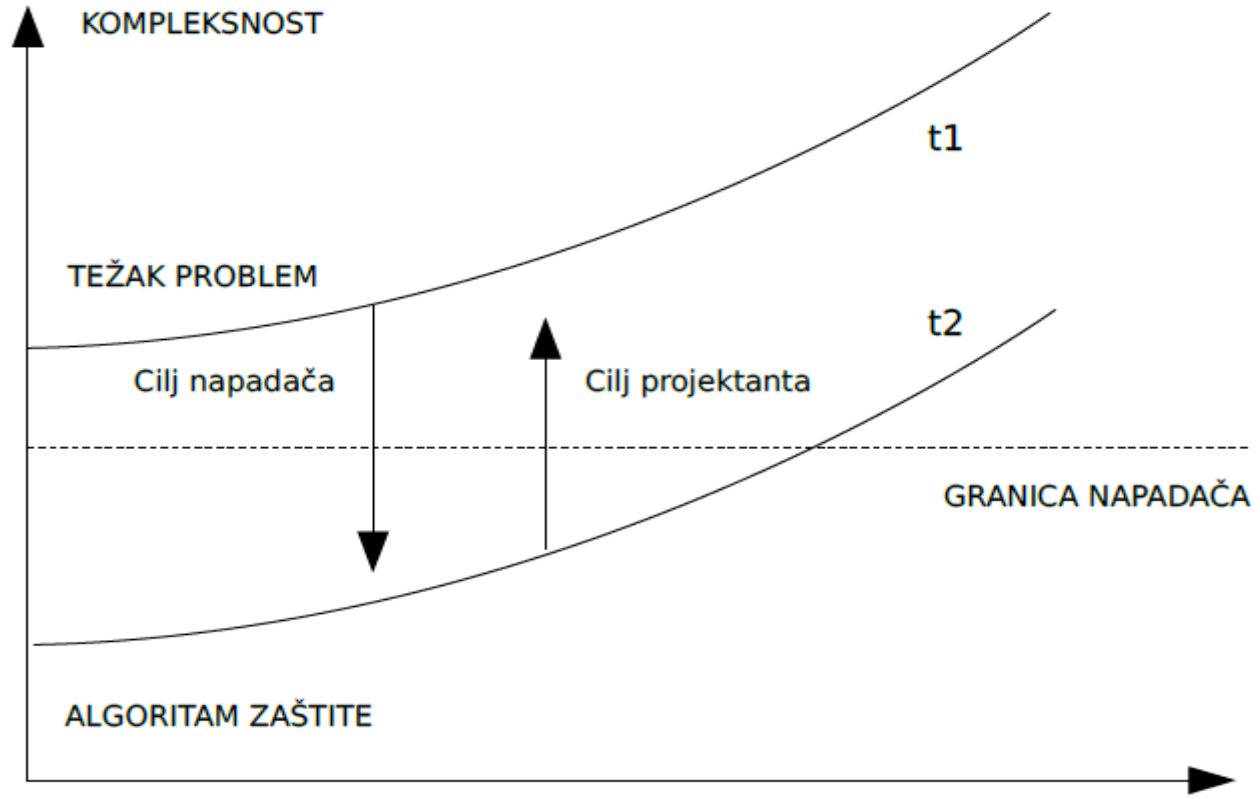


Teorijska i praktična sigurnost

- Šenon razlikuje dve vrste sigurnosti:
  - **Bezuslovna (teorijska)** koja podrazumeva sigurnost u odnosu na protivnika koji raspolaže neograničenim vremenom i računarskim resursima.
    - Danas se ova sigurnost naziva informaciono-teorijska sigurnost.
  - **Računarska (praktična)** sigurnost koja podrazumeva sigurnost u odnosu na protivnika koji ima specificirane ograničene vremenske i računarske resurse.
- Šta znači dokaziva sigurnost?
  - **Dokaziva bezuslovna sigurnost** znači da se za zadati kriptografski sistem može rigorozno dokazati nominovana sigurnost u odnosu na protivnika koji raspolaže neograničenim vremenskim i računarskim resursima.
  - **Dokaziva računarska sigurnost** znači da se za zadati kriptografski sistem može rigorozno dokazati nominovana sigurnost u odnosu na protivnika koji poseduje specificirane vremenske i računarske resurse.
- **Paradoks**: dokaziva bezuslovna sigurnost je realnost, dokaziva računarska sigurnost je mit!

Praktična sigurnost

- t_1 – kompleksnost najefikasnijeg rešenja teških problema
- t_2 – dokaziva donja granica algoritma zaštite u funkciji od t



Pojam perfektne šifre

- **Konfuzija** je složena relacija izmedju otvorenog teksta i šifrata.
- **Difuzija** je širenje statistike otvorenog teksta u okviru šifrata.
- Kriptosistem je **perfektno tajan** ako šifrat ne otkriva nikakvu informaciju o otvorenom tekstu, tj. ako važi: $I(M, C) = 0$.
- Šenonova teorema o perfektnim šifarskim sistemima.
- **Teorema 1**: Šifarski sistem je perfektan ako i samo ako:
  - Svi ključevi se koriste sa jednakom verovatnoćom.
  - Za svako x iz M i za svako y iz C, postoji samo jedan ključ k takav da je $E(x,k)=y$.
- **Teorema 2**: Ukoliko je šifra perfektna, mora biti najmanje onoliko ključeva koliko ima mogućih poruka.
- Drugim rečima, šifra je perfektna ako:
  - Za svaki šifrat, verovatnoća da odgovara proizvoljnoj poruci je ista.
  - Za svaki otvoreni tekst, verovatnoća da odgovara proizvoljnom šifratu je jednaka.

Klasična kriptografija

- Osnovu većine klasičnih algoritama za šifrovanje čine matematički postupci supstitucije (zamene) i transpozicije (permutacije).
- **Supstitucija** je zamena delova originalne poruke (pojedinačnih znakova ili grupa znakova konstantne dužine) drugim znakovima ili rezultatom neke funkcije čiji su ulaz ti znakovi i ključ (npr. TAJNA $\rightarrow$ XIWOI). Šifrovanje supstitucijom obuhvata sledeće vrste zamena:
  - **Monoalfabetska** zamena (Cezarova i Afina šifra) je bijektivno preslikavanje: svaki znak poruke preslikava se u tačno jedan znak šifrata.
  - **Polialfabetska** zamena (Vigenèreova i Playfairova šifra) je preslikavanje tipa 1-n: svaki znak poruke može se preslikati u jedan od n dozvoljenih znakova šifrata, zavisno od algoritma koji se koristi i od dužine ključa.
  - **Poligramska** zamena (Playfairova i Hillova šifra) je bijektivno preslikavanje, pri čemu se kao osnovna jedinica nad kojom se izvršava supstitucija uzima poligram (niz od više znakova).
- **Permutacijom** se originalna poruka preuređuje po nekom algoritmu (npr. ALEKSANDAR $\rightarrow$ RADNASKELA).

- Neka su promenljive x i y numerički ekvivalenti odgovarajućih slova.
- Korespondencija slova engleskog alfabeta (Z_{26}) i njihovih numeričkih vrednosti data je sa:
  - A = 0, B = 1, C = 2, D = 3, …, W = 22, X = 23, Y = 24, Z = 25.
- **Afina šifra**.
  - Šifrovanje i dešifrovanje ključem $k=\{(a,b) \mid (a,b) \in Z_n \times Z_n\}$ definisano je sa:
    - $E(x, k) = a \cdot x + b \pmod{n}$.
    - $D(y, k) = a^{-1} \cdot (y - b) \pmod{n}$.
  - Da bi dešifrovanje bilo moguće, neophodno je da parametar a ima multiplikativni inverzni element a^{-1} u prstenu Z_n
  - Drugim rečima, potrebno je da postoji element a^{-1} takav da važi $a \cdot a^{-1} = 1 \pmod{n}$.
  - Z_{26}, kojim je opisan engleski alfabet, ima samo 12 inverzibilnih elemenata, što znači da je ukupan broj ključeva koji se mogu iskoristiti za šifrovanje jednak $12 \cdot 26 = 312$.

Klasična kriptografija

- Primer šifrovanja i dešifrovanja afinom šifrom:
  - Otvoreni tekst „KRIPTO“ koji pripada engleskom alfabetu
  - Ključ $k = (3, 6)$.
- Šifrovanje: $y = 3 \cdot x + 6 \pmod{26}$
  - K: $10 \cdot 3 + 6 \pmod{26} = 10 \rightarrow$ K
  - R: $17 \cdot 3 + 6 \pmod{26} = 5 \rightarrow$ F
  - I: $8 \cdot 3 + 6 \pmod{26} = 4 \rightarrow$ E
  - P: $15 \cdot 3 + 6 \pmod{26} = 25 \rightarrow$ Z
  - T: $19 \cdot 3 + 6 \pmod{26} = 11 \rightarrow$ L
  - O: $14 \cdot 3 + 6 \pmod{26} = 22 \rightarrow$ W

Dešifrovanje: $x = 9 \cdot (y - 6) \pmod{26}$

K: $9 \cdot (10 - 6) \pmod{26} = 10 \rightarrow$ K

F: $9 \cdot (5 - 6) \pmod{26} = 17 \rightarrow$ R

E: $9 \cdot (4 - 6) \pmod{26} = 8 \rightarrow$ I

Z: $9 \cdot (25 - 6) \pmod{26} = 15 \rightarrow$ P

L: $9 \cdot (11 - 6) \pmod{26} = 19 \rightarrow$ T

W: $9 \cdot (22 - 6) \pmod{26} = 14 \rightarrow$ O

- Afina šifra ima prostor ključeva od 312 mogućih ključeva za slučaj engleskog alfabeta.
- Napad grubom silom se može efikasno izvesti.
- Alternativni način: ukoliko je tekst dovoljno dugačak, vrlo je verovatno da slova koja se najčešće pojavljuju u šifratu odgovaraju najučestalije korišćenim slovima govornog jezika.
- Primer:
  - Šifrat „KNBERXKBYLKIRYHBCKXKNYFE“ je dobijen šifrovanjem poruke na srpskom jeziku (engleski alphabet).
  - U šifratu se najčešće pojavljuju slova „K“ i „Y“.
  - Može se pretpostaviti da su ta dva slova šifrati najfrekventnijih slova srpskog jezika „A“ i „I“.
  - Dakle, dobija se sistem jednačina zasnovan na operacijama šifrovanja slova „A“ i „I“:
    - $0 \cdot a + b = 10 \pmod{26}$
    - $8 \cdot a + b = 24 \pmod{26}$
  - Rešavanjem se dobija $(a, b) = (5, 10)$.
  - Na osnovu toga sledi: $x = 21 \cdot (y - 10) \pmod{26}$.

- **Vigenèreova šifra** je polialfabetska, što znači da se svako slovo otvorenog teksta može preslikati u jedno od *m* mogućih slova, gde je *m* dužina ključa.
  - Neka je: *n* broj slova u alfabetu, $k = (k_1, k_2, \ldots, k_m)$ ključ dužine *m*, $x = (x_1, x_2, \ldots, x_m)$ deo otvorenog teksta dužine *m*, $y = (y_1, y_2, \ldots, y_m)$ deo šifrata dužine *m*.
  - Operacije šifrovanja i dešifrovanja date su sa:
    - $E(x, k) = x_1 + k_1 \pmod n, x_2 + k_2 \pmod n, \ldots x_m + k_m \pmod n$
    - $D(y, k) = y_1 - k_1 \pmod n, y_2 - k_2 \pmod n, \ldots y_m - k_m \pmod n$.
- Primer:

Otv. tekst	D	U	S	K	**O**	D	U	G	**O**	U	S	K	**O**
Ključ	K	O	J	O	**T**	K	O	J	**O**	T	K	O	**J**
Šifrat	N	I	B	Y	**H**	N	I	P	**C**	N	C	Y	**X**

- Prostor ključeva: $n^1 + n^2 + \ldots + n^m$, gde je *n* broj slova u alfabetu, a *m* najveća dužina ključa.
  - Primer za za n=26, k=5 treba ispitati 12.356.630 ključeva

Klasična kriptografija

- **Playfairova** šifra je polialfabetska i poligramska, što znači da operaciju šifrovanja obavlja nad parovima slova i to tako da rezultat zavisi i od jednog i od drugog slova.
- Ključ Playfairove šifre je matrica koja se formira na osnovu ključne reči.
  - Ukoliko se koristi engleski alfabet, matrica je veličine 5x5 elemenata.
  - Konstrukcija matrice: u polja počev od prvog reda prve kolone upisuje se ključna reč (pri čemu se slova koja su već upisana ne ponavljaju) a zatim ostatak alfabeta.
  - Engleski alfabet ima 26 slova, matrica 25 elemenata: po dogovoru se „I“ i „J“ poistovećuju.
  - Primer: ključna reč „KRIPTOGRAFIJA“.

K	R	I / J	P	T
O	G	A	F	B
C	D	E	H	L
M	N	Q	S	U
V	W	X	Y	Z

Klasična kriptografija

- Poruka koja se šifruje deli se na parove slova.
- Poruka se šifruje zamenom parova slova otvorenog teksta na osnovu ključa.
  - Slova koja se nalaze u istom redu menjaju se slovima koja se nalaze jedno mesto u desno (ciklički): GF $\rightarrow$ AB, EL $\rightarrow$ HC;
  - Slova koja se nalaze u istoj koloni menjaju se slovima koja se nalaze jedno mesto ispod (ciklički): AQ $\rightarrow$ EX, DW $\rightarrow$ NR;
  - U suprotnom, slova formiraju pravougaonik u matrici i zamenjuju se slovima koja se nalaze u preostala dva ugla tog pravougaonika: GY $\rightarrow$ FW, PN $\rightarrow$ RS.

K	R	I / J	P	T
O	G	A	F	B
C	D	E	H	L
M	N	Q	S	U
V	W	X	Y	Z

Klasična kriptografija

- Primer: „MOJA PORUKA“
  - Deli se na parove „MO JA PO RU KA“.
  - Nakon šifrovanja dobijaju se sledeći parovi slova: „VC AE KF TN IO“.
- Napomene:
  - U slučaju da se u otvorenom tekstu nalazi par za šifrovanje koji čine ista slova, između njih se po dogovoru ubacuje slovo X.
  - U sličaju da je broj slova neparan, poslednji par se formira dodavanjem slova A. Na primer, reč „MITTWOCH“ se pre šifrovanja deli na parove „MI TX TW OC HA“.

K	R	I / J	P	T
O	G	A	F	B
C	D	E	H	L
M	N	Q	S	U
V	W	X	Y	Z

One Time Pad

- Operacija ekskluzivno ILI (*eXclusive OR*, *XOR*) definisana je jednačinama:
  - $0 \oplus 0 = 1 \oplus 1 = 0$.
  - $0 \oplus 1 = 1 \oplus 0 = 1$.
- Šifrovanje zasnovano na ovoj logičkoj operaciji oslanja se na njenu osobinu:
  - $(A \oplus B) \oplus B = A$.
- Drugim rečima, ako se operacija ekskluzino ILI primeni dva puta na vrednost *A* sa istim operandom *B*, kao rezulat se dobija vrednost *A*.
- Otvoreni tekst se šifruje primenom operacije ekskluzivno ILI na bitove otvorenog teksta i bitove ključa. Operacija dešifrovanja je identična operaciji šifrovanja.

Otv. tekst	1	0	0	1	0	1	0	1	1	0	1	0	0	0	1	1
Ključ	1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	1
Šifrat	0	1	1	0	0	1	0	0	0	1	0	1	0	0	1	0

One Time Pad

- **Jednokratna beležnica** (engl. *One Time Pad*, *OTP*) može se posmatrati kao varijanta XOR algoritma kod kog je dužina ključa jednaka dužini otvorenog teksta.
- Ključ i otvoreni tekst šalju se primaocu različitim kanalima.
- Šifrat je siguran, a poruku je nemoguće ispravno dešifrovati bez pravog ključa.
- U ovom slučaju veći broj ključeva tokom dešifrovanja može (ali ne mora) proizvesti smisaone otvorene tekstove, ali samo jedan odgovara originalnoj poruci.
  - Primer: u zavisnosti od toga čime dešifrujete šifrat AOISUAOHDOIA možete dobiti smislene otvorene tekstove „OPA MILE LALE“ ili „JOJ CRNO DETE“, ali su njihova značenja suprotna.
- Sigurnost OTP algoritma se značajno uvećava ukoliko se svaki put koristi različit ključ (time se sprečava mogućnost da treće lice dođe do ključa i dešifruje neke poruke).
- Problemi:
  - Premašenje (dužina ključa je jednaka dužini šifrata).
  - Sigurnan kanal za distribuciju ključa.
  - Problem generisanja pravih slučajnih brojeva (NE pseudoslučajnih sekvenci) koji će imati ulogu ključa.

Simetrični algoritmi

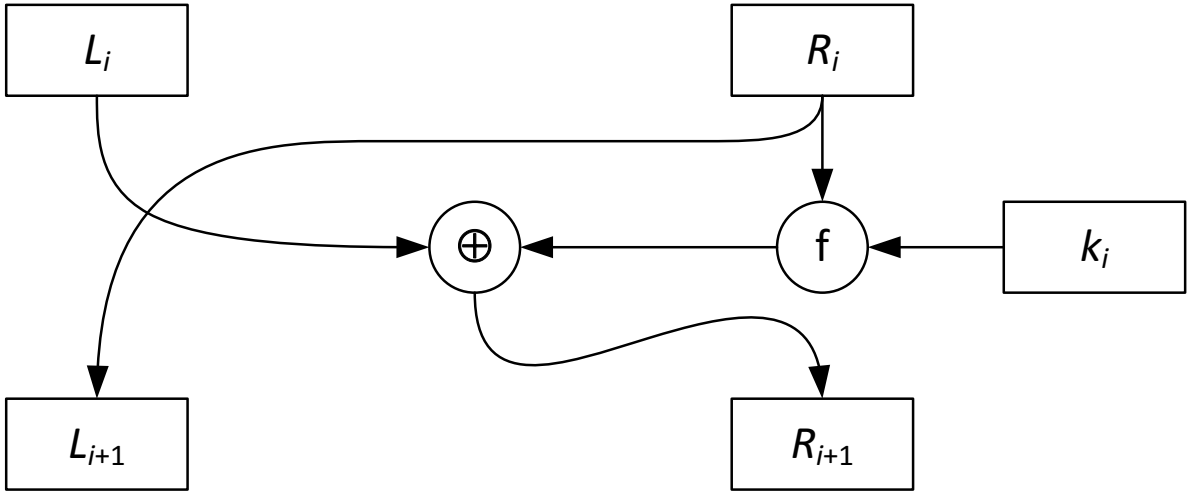
- Simetrični algoritmi se mogu podeliti na blokovske (engl. *block*) i sekvencijalne (engl. *stream*).
- **Blokovski algoritmi** šifruju otvoreni tekst na nivou bloka bitova konstantne dužine
  - Jedan blok otvorenog teksta pomoću istog ključa se uvek prevodi u isti šifrat.
  - Ova pretpostavka važi pod uslovom da ne postoji povratna sprega (tipa CBC).
- **Sekvencijalni algoritmi** šifruju otvoreni tekst bit po bit (ili znak po znak) koristeći transformaciju koja se obično menja u vremenu.
  - Najčešće su zasnovani na generatorima pseudoslučajnih brojeva čije je početno stanje zadato ključem.
  - Šifrat se dobija primenom operacije ekskluzivno ILI generisanog niza sa otvorenim tekstom.
  - Otvoreni tekst se dobija primenom operacije ekskluzivno ILI generisanog niza sa šifratom.

DES

- Američki nacionalni biro za standarde (engl. *National Bureau of Standards*, *NBS*) raspisao javni konkurs za kriptosistem koji je trebalo da zadovolji sledeće uslove:
  - Visok nivo sigurnosti koja leži u ključu, a ne u tajnosti algoritma
  - Efikasnost
  - Potpuna specifikacija i lako razumevanje algoritma,
  - Ekonomičnost implementacije u elektronskim uređajima,
  - Dostupnost svim korisnicima
  - Mogućnost izvoza (zbog američkih zakona o izvozu).
- Pobednik konkursa je prihvaćen kao ***Data Encryption Standard*** (*DES*) 1976. godine, nakon izmena u kojima je učestvovala i *NSA*). Algoritam je:
  - Simetričan, blokovski orijentisan algoritam koji koristi Feistelove mreže sa 16 rundi.
  - Šifruje tekst u blokovima dužine 64 bita, koristeći ključ *k* dužine 56 bitova (+8 bita za proveru parnosti) što rezultuje šifratom dužine 64 bita.
  - Tri osnovna koraka u algoritmu su: inicijalna permutacija, 16 rundi obrade podataka (proširenje, XOR sa ključem, supstitucija) i završna inverzna permutacija.

DES

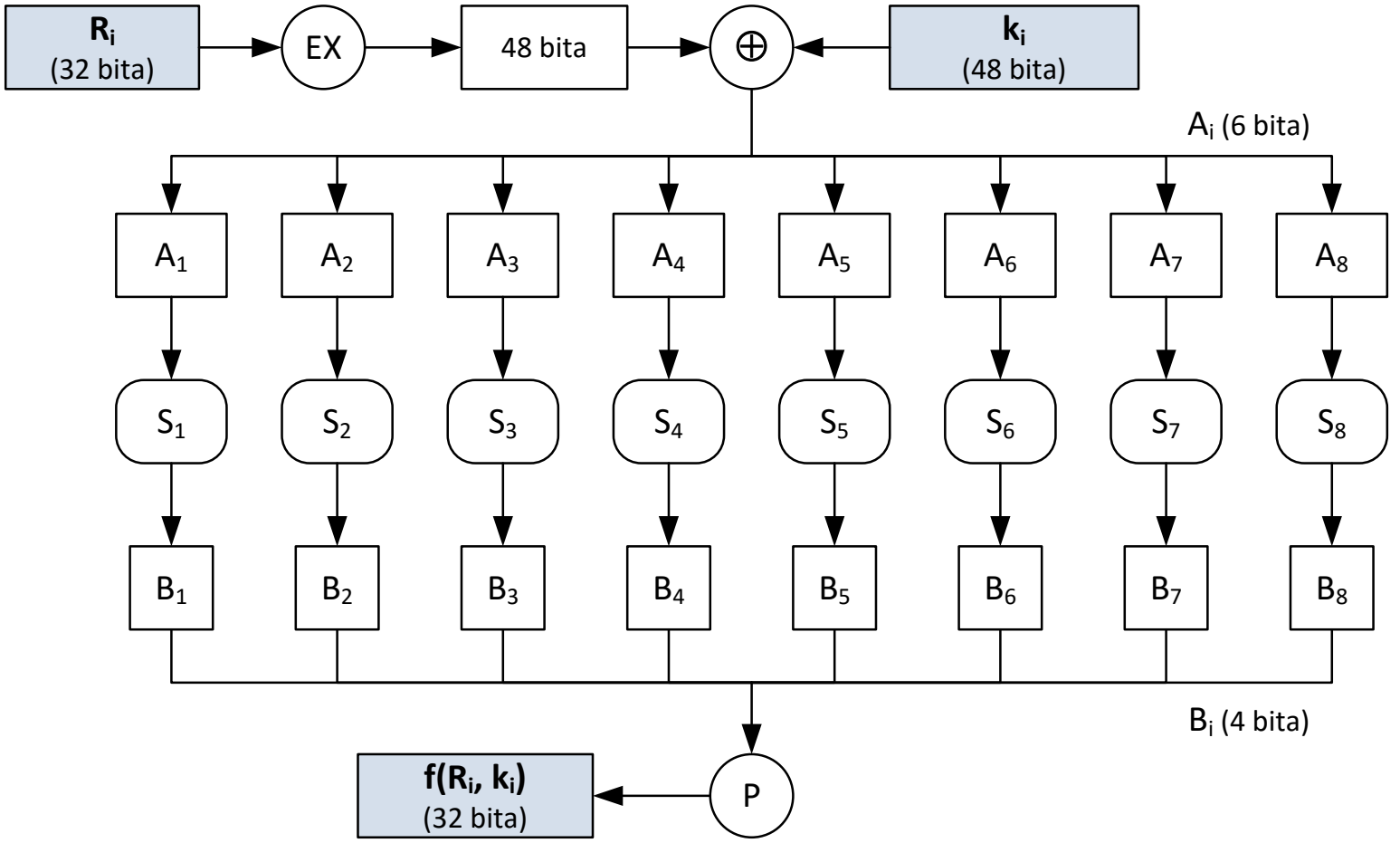
- Runde DES algoritma su zasnovane na **Feistelovoj mreži**.
- Ulaz se deli na dva dela od po 32 bita: L_i i R_i.
  - $L_{i+1} = R_i$
  - $R_{i+1} = L_i \oplus f(R_i, k_i)$
- Vrednosti k_i su **podključevi** (ključevi runde) dužine 48 bita koji se generišu na osnovu ključa.



DES

- **Funkcija *f*** (v. str. 29) prihvata dva ulazna argumenta: niža 32 bita izlaza iz prethodne runde (R_i) i potključ dužine 48 bitova (k_i). Kao rezultat se dobija niz dužine 32 bita.
- Niz R_i permutuje se i proširuje se do niza dužine 48 bitova prema fiksnoj funkciji proširenja *EX*.
- Računa se vrednost $A = EX(R_i) \oplus k_i$.
  - Rezultat se zapisuje u obliku osam 6-bitnih nizova: $A = A_1A_2A_3A_4A_5A_6A_7A_8$.
- Svaka supstitucijska kutija S_j (S_1, S_2, ... , S_8) predstavlja fiksnu matricu dimenzija 4×16, čiji su elementi celi brojevi između 0 i 15.
- Za dati niz od 6 bitova, $A_j = a_1a_2a_3a_4a_5a_6$, rezultat supstitucije $S_j(A_j)$ računa se na sledeći način:
  - Dva bita a_1a_6 određuju binarni zapis reda ($0 \leq red \leq 3$) u S_j.
  - Četiri bita $a_2a_3a_4a_5$ određuju binarni zapis kolone ($0 \leq kol \leq 15$) u S_j.
  - $B_j = S_j(A_j) = S_j(red,kol)$ zapisano kao binarni broj dužine 4 bita.
  - Na ovaj način se određuje $B = B_1B_2...B_8 = S_1(A_1)S_2(A_2)...S_8(A_8)$.
- Niz bitova *B* dužine 32 bita permutuje se pomoću fiksne završne permutacije *P*.
- Tako se dobija $P(B)$, tj $f(R_i, k_i)$.

DES


R_i (32 bita)
EX
48 bita
⊕
k_i (48 bita)
A_i (6 bita)
A_1 A_2 A_3 A_4 A_5 A_6 A_7 A_8
S_1 S_2 S_3 S_4 S_5 S_6 S_7 S_8
B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8
B_i (4 bita)
P
$f(R_i, k_i)$ (32 bita)

DES

- **Generisanje potključeva**:
  - Ključ *k* dužine 56 bitova, koji se koristi prilikom šifrovanja, čuva se u obliku *K*, dužine 64 bita.
  - Bitovi parnosti na pozicijama 8, 16, 24, 32, 40, 48, 56 i 64 definisani su tako da svaki bajt sadrži neparan broj jedinica.
  - Preostalih 56 bitova ključa *k* permutuje se pomoću fiksne permutacije PK_1.
  - Zapisuje se $PK_1(k) = C_0D_0$, gde su C_0 i D_0 viših i nižih 28 bitova u $PK_1(k)$.
  - Za $i = 1, 2, \ldots, 16$ računa se:
    - $C_i = LS_i(C_{i-1})$
    - $D_i = LSi(D_{i-1})$,
    - $k_i = PK_2(C_iD_i)$.
  - *LSi* je ciklički pomeraj ulevo za jednu poziciju, ako je i=1, 2, 9 ili 16, a u svim ostalim slučajevima za dve pozicije.
  - PK_2 je fiksna permutacija.
- Za dešifrovanje DES šifrata koristi se isti algoritam kao i za šifrovanje.
  - Potključevi se koriste u obrnutom redosledu: $k_{16}, k_{15}, \ldots, k_1$.

Sigurnost DES algoritma

- Dužina ključa je problem: prvobitna dužina je bila 112 bita, pa je NSA „ošišao“ ključ na 56 bita, što je nedovoljno.
- Postoji 64 ključeva koji se ne koriste:
  - Slabi (4 ključa) generišu iste potključeve u svakoj rundi: šifrovanje i dešifrovanje u svakoj rundi jednako.
  - Delimično slabi (12 ključeva) generišu samo dva različita potključa: šifrovanje s jednim ključem isto je što i dešifrovanje s drugim.
  - Potencijalno slabi (48 ključeva) generišu samo četiri različita potključa.
- Ni linearna ni diferencijalna kriptoanaliza nisu dovele do razbijanja DES-a, ali je njihova važnost u tome što su primenljive na svaki simetričan blokovski algoritam.
- **Linearna kriptoanaliza** (Mitsuru Matsui, 1993), najverovatnije nepoznata tvorcima DES-a, zasniva se na činjenici da se neki bitovi ključa, iako nisu linearne funkcije otvorenog teksta i šifrata, mogu dobro aproksimirati linearnom funkcijom.
  - Pomoću linearne kriptoanalize Matsui je opisao napad tipa poznati otvoreni tekst koji za razbijanje DES-a zahteva prosečno 2^{43} otvorenih tekstova.

Sigurnost DES algoritma

- **Diferencijalna kriptoanaliza** (Eli Biham i Adi Shamir, 1990) spada u napade tipa odabran otvoreni tekst.
  - Osnovna ideja diferencijalne kriptoanalize je poređenje rezultata operacije ekskluzivno ILI, izvršene nad dva otvorena teksta sa rezultatom iste operacije izvršene nad dva odgovarajuća šifrata.
  - Drugim rečima, posmatraju se dva bloka otvorenog teksta sa specifičnim razlikama i analizira se evolucija te razlike pri prolasku kroz algoritam.
  - Na osnovu razlika u dobijenim šifratima, različitim ključevima se dodeljuju verovatnoće.
  - Nakon ispitivanja većeg broja parova otvorenog teksta, određuje se najverovatniji ključ.
  - Metoda je očigledno bila poznata konstruktorima DES-a, što se vidi iz načina na koji su konstruisane supstitucijske kutije i permutacija *P*.

Sigurnost DES algoritma

- Postoje derivati DES algoritma koji su korišćeni za povećanje sigurnosti.
- **Dvostruki DES**, koji koristi dva različita ključa:
  - $y = e_{k2}(e_{k1}(x))$
  - $x = d_{k1}(d_{k2}(y))$
  - Za razbijanje DES-a *brute force*-om potrebno je u najgorem slučaju ispitati 2^{112} ključeva.
  - Pokazano je da je broj operacija potreban za kriptoanalizu dvostrukog DES-a pomoću napada susret u sredini (engl. *meet-in-the-middle*, Diffie, 1997.) reda veličine 2^{57}.
  - To je neznatno više od broja operacija potrebnih za kriptoanalizu običnog DES-a
  - U ovom slučaju se dupliranjem operacija sigurnost neznatno uvećava!
- **Trostruki DES**, koji koristi tri različita ključa.
  - $x = d_{k1}(e_{k2}(d_{k3}(y)))$
  - $y = e_{k3}(d_{k2}(e_{k1}(x)))$
  - Broj potrebnih operacija prilikom napada „susret u sredini“ je reda veličine 2^{112}, što znači da je sigurnost trostrukog šifrovanja onakva kakvu smo očekivali od dvostrukog.

AES

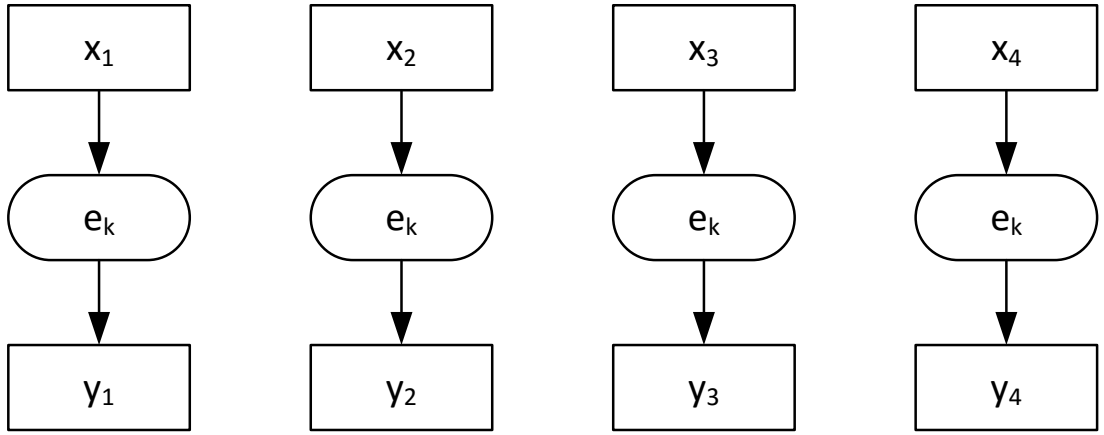
- NIST je 1997. godine raspisao konkurs za kriptosistem koji menja DES.
- Uslovi konkursa su:
  - Algoritam je simetričan blokovski.
  - Algoritam obavlja šifrovanje blokova otvorenog teksta dužine 128 bitova.
  - Ključ dužine 128, 192 i 256 bitova.
  - Ne postoje slabi ključevi.
- Finalisti:
  - RIJNDAEL (Joan Daemen i Vincent Rijmen) – pobednik konkursa.
  - TWOFISH (*Counterpane Systems*) – Feistelova mreža sa 16 rundi, S-kutije se dinamički menjaju u zavisnosti od ključa, što otežava diferencijalnu i linearnu kriptoanalizu.
  - RC6 (*RSA Data Security Inc.*) – Feistelova mreža, 20 rundi, otpornost na kriptoanalizu. pomoću rotacije koja zavisi od samih podataka.
  - MARS (IBM).
  - SERPENT.

AES

- Algoritam RIJNDAEL koristi veličina bloka za šifrovanje 128, 192 ili 256 bitova, dužinu ključa 128, 192 ili 256 bitova, a broj rundi zavisi od dužne ključa i veličine bloka.
- Nije Feistelova mreža!
- Karakterističan je po tome što prilikom konstrukcije supstitucijskih kutija koristi operacije u konačnom polju GF(2^8).
- Jedna RIJNDAEL runda sadrži tri sloja.
  - Linearni difuzioni sloj (engl. *linear mixing layer*): obezbeđuje veliku difuziju bitova nakon nekoliko rundi – funkcije *ShiftRow* (pomeranje okteta u redovima matrice stanja) i *MixColumn* (mešanje podataka u kolonama matrice stanja).
  - Nelinearni sloj (engl. *non-linear layer*): upotreba supstitucijskih kutija optimizovanih za najgori slučaj – funkcija *ByteSub* (zamena okteta na osnovu tabele supstitucije).
  - Sloj dodavanja ključa (engl. *key addition layer*) – operacija ekskluzivno ILI nad potključem runde sa trenutnim stanjem bloka (funkcija *AddRoundKey*).
- Broj rundi zavisi od veličine bloka i dužine ključa: u RIJNDAEL algoritmu određen je onim što je duže, a što se AES standarda tiče, blok je uvek dužine 128 bitova.

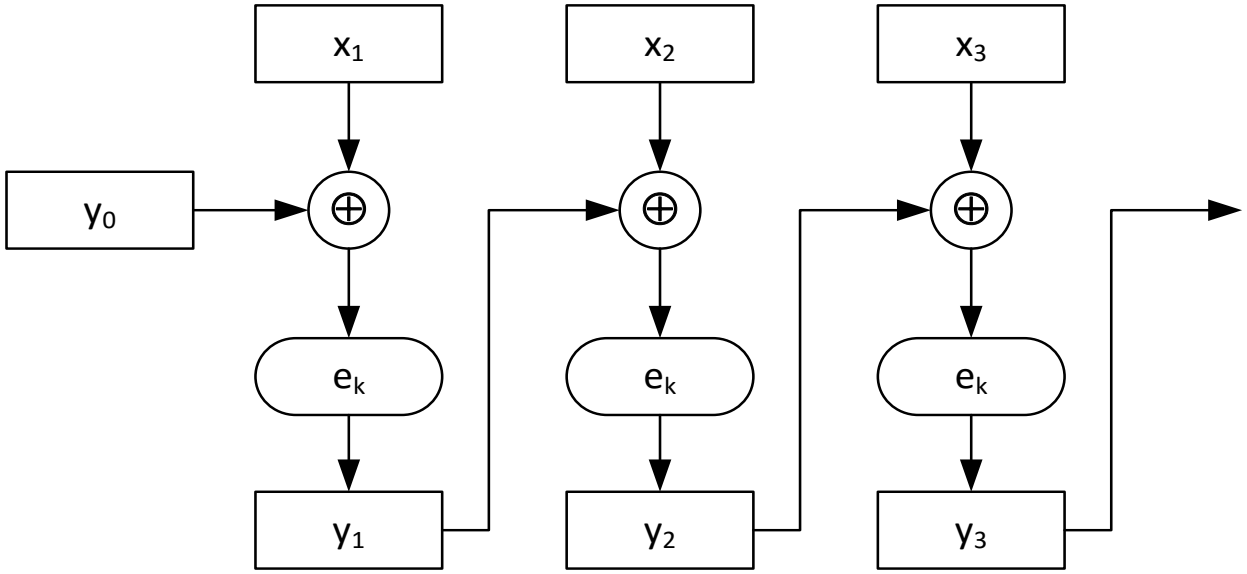
Režimi rada

- **ECB** (engl. *Electronic Code Book*).
  - Poruka se deli na blokove čije dužine odgovaraju dužinama bloka za šifrovanje algoritma koji se koristi.
  - Po potrebi poslednji blok se dopunjuje slučajno generisanim nizom.
  - Šifrovanje se obavlja blok po blok pomoću istog ključa.
  - Identičnim blokovima otvorenog teksta odgovaraju identični blokovi šifrata.
  - Greška u prenosu jednog bloka šifrata ne propagira se duž ostatka šifrata.



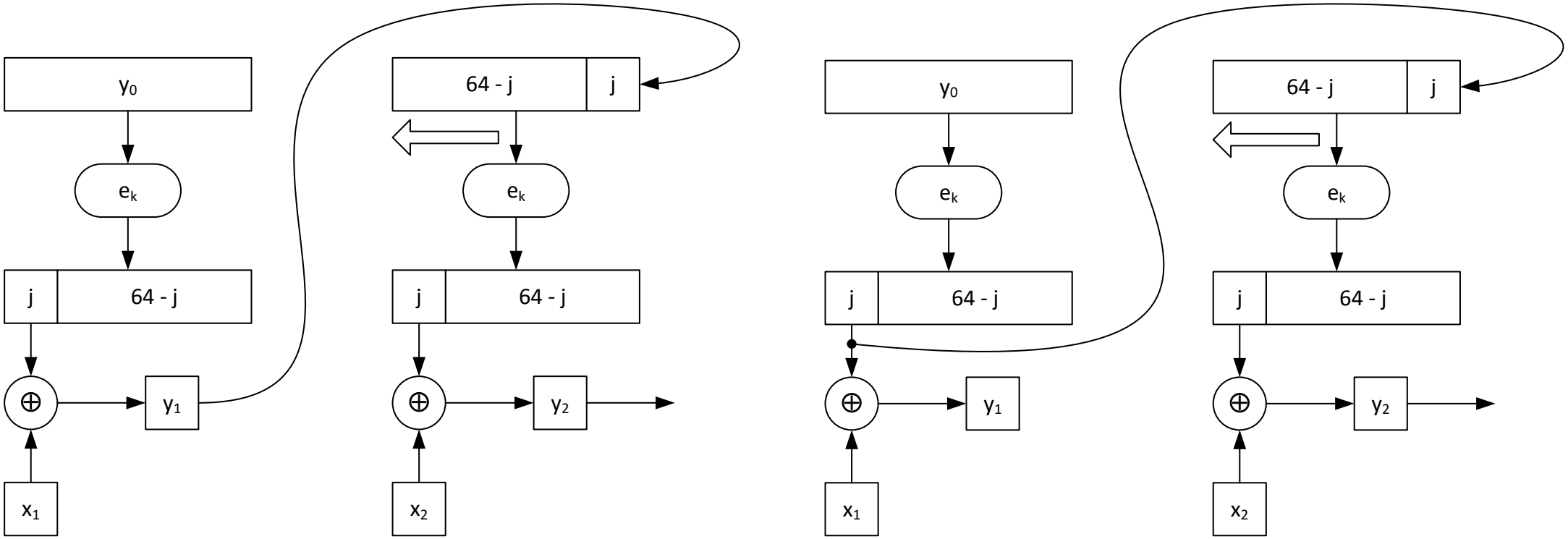
Režimi rada

- **CBC** (engl. *Cipher Block Chaining*):
  - Najpre se računa rezultat operacije XOR izvršene nad trenutnim blokom otvorenog teksta i šifratom prethodnog bloka, a rezultat se šifruje ključem.
  - Identičnim blokovima otvorenog teksta u opštem slučaju odgovaraju različiti šifrati.
  - Vrednost y_0 je inicijalizujući vector (IV) koji mora biti poznat i primaocu i pošiljaocu.
  - Greška u transmisiji jednog bloka propagira se duž šifrata.



Režimi rada

- **CFB** (*Cipher Feedback*, levo) i **OFB** (*Output Feedback*, desno) nameću sekvencijalno šifrovanje!
  - U OFB režimu greške u transmisiji nemaju uticaja na na ostatak šifrata (npr. greška u delu šifrata $y_{1,}$ prilikom dešifrovanja proizvešće neispravan samo deo otvorenog teksta x_1).

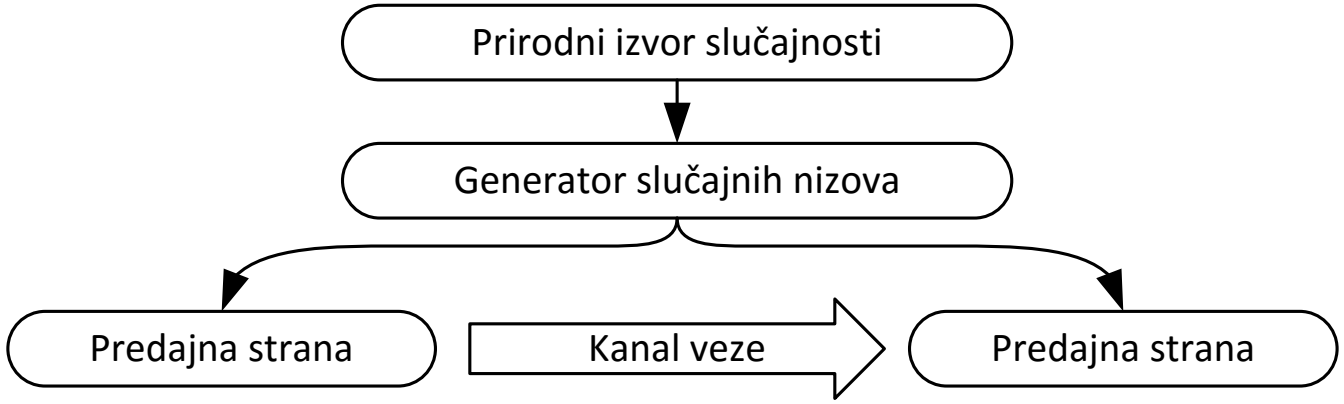


Generatori slučajnih i pseudoslučajnih brojeva

- **Generator slučajnih brojeva** (engl. *True Random Number Generator*, *TRNG*).
  - Na osnovu hardverskog ili softverskog izvora slučajnosti na izlazu daje niz statistički nezavisnih binarnih brojeva.
  - Niz generisanih bitova je uvek različit.
  - Tipičan primer: bacanje novčića više puta, verovatnoća pismo/glava je 0,5, niz je slučajan.
    - Ignorisati 10 uzastopnih pisama koje je izveo Derren Brown! To nije slučajno!
- **Generator pseudoslučajnih brojeva** (engl. *Pseudo Random Number Generator*, *PRNG*).
  - Na osnovu slučajne početne vrednosti (ključa *s*) na izlazu daju niz *k* znatno duži od ključa.
  - Deterministički je algoritam: za svako *s* uvek generiše identičnu sekvencu *k*.
  - Generisani niz u prihvatljivoj meri zadovoljava određene statističke testove.
- Za razliku od *TRNG*, *PRNG* odlikuje periodičnost!
- Primer *PRNG* je linearni kongruentni generator (LCG):
  - $x_1 = a \cdot s + b \pmod{n}$, $x_n = a \cdot x_{n-1} + b \pmod{n}$.

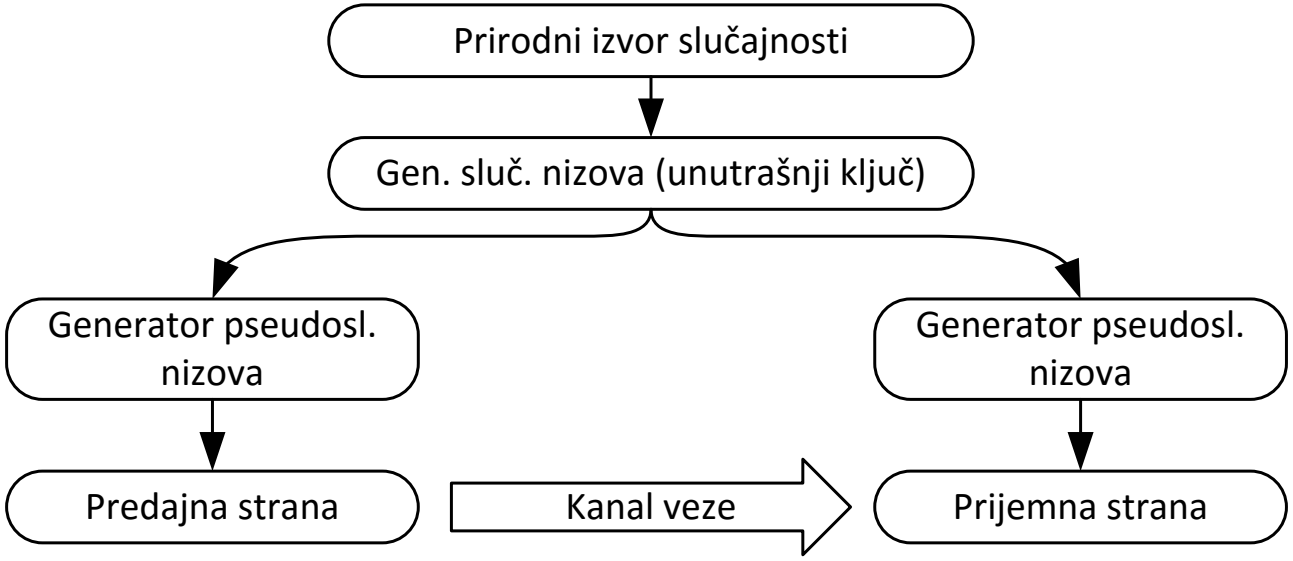
Šenonov ideal

- Slučajna sekvenca se generiše i distribuira pošiljaocu i primaocu.
  - Pošiljaoc XOR-uje poruku sa sekvencom i šalje šifrat primaocu.
  - Primalac XOR-uje šifrat sa sekvencom i dobija otvoreni tekst, tj. poruku.
- Problemi:
  - Dobar generator pravih slučajnih brojeva.
  - Sigurna distribucija slučajne sekvence pošiljaocu i primaocu.



Šenonov pseudo-ideal

- Slučajna sekvenca se generiše i distribuira pošiljaocu i primaocu koji koriste isti *PRNG*.
  - Pošiljaoc prima slučajnu sekvencu, koristi je kao ključ za *PRNG*, generiše pseudoslučajnu sekvencu, XOR-uje poruku sa tom sekvencom i šalje šifrat.
  - Primalac prima šifrat i slučajnu sekvencu, koristi je kao ključ za *PRNG*, generiše pseudoslučajnu sekvencu i XOR-uje šifrat sa sekvencom.
- Problemi pseudo-ideala: problemi ideala + dobar PRNG + resinhronizacija.



Kriptografski generatori pseudoslučajnih brojeva

- Izlaz *PRNG* se koristi za:
  - XOR šifrovanje / dešifrovanje (v. Šenonov pseudo-ideal).
  - *Salt* vrednost.
  - Nonce (*Number used ONCE*).
- Ukoliko se radi o kriptografskom *PRNG* neophodno je:
  - Odabrati početnu vrednost koju je nemoguće otkriti je metodom pogađanja ili pretraživanja svih mogućih kombinacija (drugim rečima, iz izvora slučajnosti).
  - Osigurati nepredvidivost brojeva u generisanom nizu.
    - Napadač zna način funkcionisanja generatora, ali je mogućnost kriptoanalitičkih napada svedena na minimum (na osnovu prethodnih *n* vrednosti ne može odrediti sledeću).
    - Primer: LCG zadovolja statističke testove, ali je predvidljiv.
  - Obezbediti veliki broj stanja, odnosno veliku periodu.
  - Izbeći korelaciju (zavisnost podnizova generisanog niza).

Kriptografski generatori pseudoslučajnih brojeva

- Primer upotrebljivog generatora: **RSA** generator pseudoslučajnih brojeva.
  - Generišu se dva velika prosta broja p i q
  - Računa se proizvod $n = p \cdot q$
  - Računa se Ojlerova funkcija $\varphi(n) = (p-1) \cdot (q-1)$
    - Napomena: Ojlerova funkcija broja n jednaka je broju brojeva manjih od n koji su uzajamno prosti sa n.
      - Za proste brojeve p i q važi $\varphi(n) = (p-1) \cdot (q-1)$.
  - Bira se celobrojna vrednost e sa intervala $1 < e < \varphi(n)$ koja je uzajamno prosta sa $\varphi(n)$
  - Na intervalu $[1, n-1]$ bira se slučajan broj x_0
  - Generator u petlji obavlja sledeću operaciju: $x_i = x_{i-1}^{\,e} \pmod{n}$
  - Nakon n iteracija, izlaz generatora su najmanje značajni bitovi brojeva $x_0, x_1, \ldots x_n$

Ispitivanje slučajnosti

- Dva problema:
  - Nemoguće je matematički dokazati da li su generisane sekvence slučajne.
  - Univerzalni test kojim se meri kvalitet generatora pseudoslučajnih brojeva ne postoji.
- Slučajnost se ispituje:
  - Većim brojem testova od kojih svaki meri neku od poželjnih karakteristika niza.
  - Definišu se statističke vrednosti na osnovu koje se određeni rezultat prihvata ili odbacuje i vrednost za koju se očekuje da neće biti premašena.
  - Nakon serije testova moguće je utvrditi da li generator poseduje željena svojstva.
- Baterija testova definisana **FIPS 140-1** standardom (zastarelo):
  - Monobitni test (da li je broj 0 i 1 podniza koji se testira u određenom intervalu).
  - Poker test (ispitivanje pojavljivanja različitih blokova dužine 4).
  - Ispitivanje broj pojavljivanja monotonih podnizova različitih dužina (1-6).
  - Test dugačkih podnizova (da li ima monotonih podnizova dužih od 43).

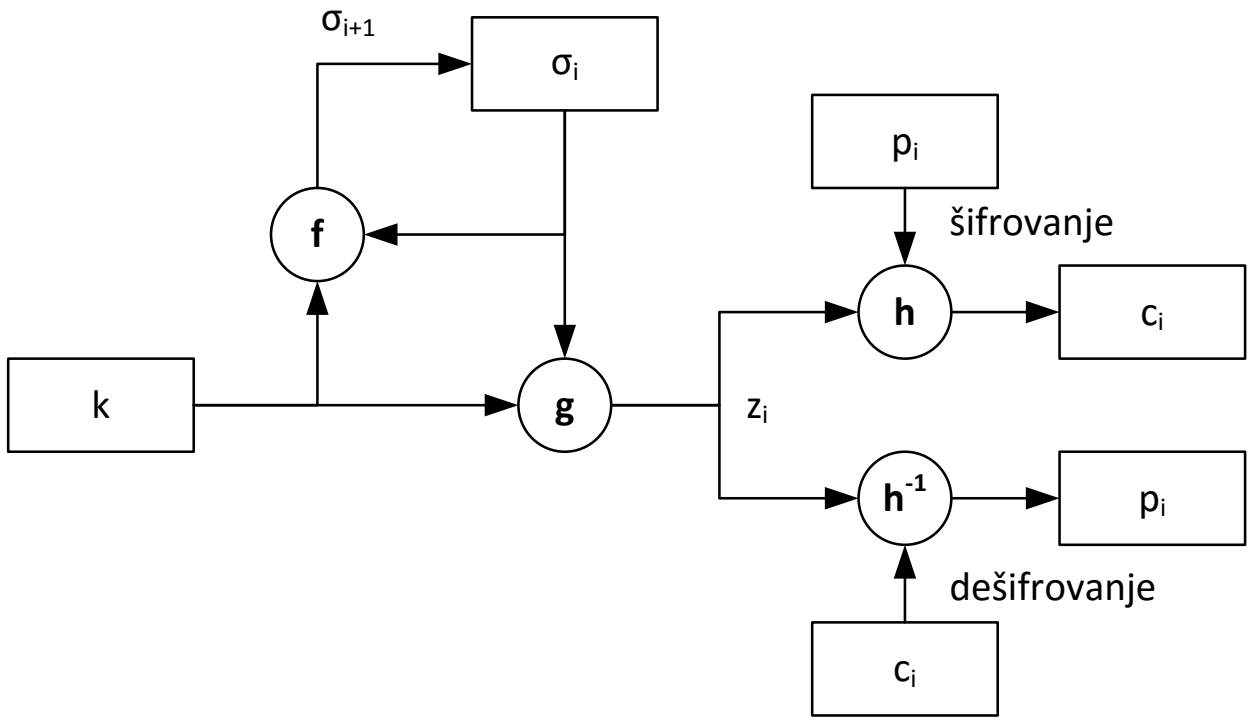
Ispitivanje slučajnosti

- NIST preporučuje nekoliko testova za ispitivanje generatora pseudoslučajnih brojeva, uključujući modifikovane testove iz FIPS 140-1 standarda, kao i neke nove, kao što su:
- Mauerov univerzalni statistički test.
  - Služi za ispitivanje mogućnosti kompresije niza bez gubitka informacije.
  - Kompresija je efikasna ukoliko niz pokazuje periodična svojstva.
  - Za niz koji se može značajno komprimovati smatra se da nije niz slučajnih bitova.
- Ispitivanje preklapajućih uzoraka.
  - Posmatrana karakteristika u testu preklapajućih uzoraka je učestalost pojave svih mogućih *n*-bitnih uzoraka uz preklapanje u celom ispitivanom nizu.
  - Test bi trebalo da otkrije da li je broj pojava preklapajućih uzoraka približno jednak broju koji se očekuje za niz slučajnih brojeva.
- Ispitivanje približne entropije.
  - Analizira se učestalost pojave svih mogućih preklapajućih *n*-bitnih uzoraka u nizu.
  - Cilj testa je poređenje učestalosti preklapajućih blokova sa očekivanim rezultatima.
- Test zasnovan na diskretnoj Furijeovoj transformaciji.

Sinhroni sekvencijalni kriptosistemi

- Ključna sekvenca (engl. *keystream*) generiše se nezavisno od otvorenog teksta i šifrata.
- Za sinhroni kriptosistem važi:
  - $\sigma_{i+1} = f(\sigma_i, k)$
  - $z_i = g(\sigma_i, k)$
  - $c = h(z_i, m_i)$
- Funkcija f prevodi sistem iz stanja σ_i u sledeće stanje σ_{i+1}, pri čemu je σ_0 inicijalno stanje koje se izvodi iz ključa k.
- Funkcija g generiše ključnu sekvencu na osnovu trenutnog stanja i ključa, a izlazna funkcija h generiše šifrat na osnovu otvorenog teksta i ključne sekvence.
- Dešifrovanje se obavlja slično, s tim što se prilikom generisanja otvorenog teksta na osnovu šifrata koristi inverzna funkcija h^{-1}.
- Nedostatak: pošiljalac i primalac moraju biti sinhronizovani po pitanju stanja i ključa.
  - U slučaju gubitka sinhronizacije, dešifrovanje je nemoguće, pa se u kriptosistem ugrađuju posebni resinhronizacioni mehanizmi.
- Dobra osobina: problem propagacije greške duž šifrata ne postoji.

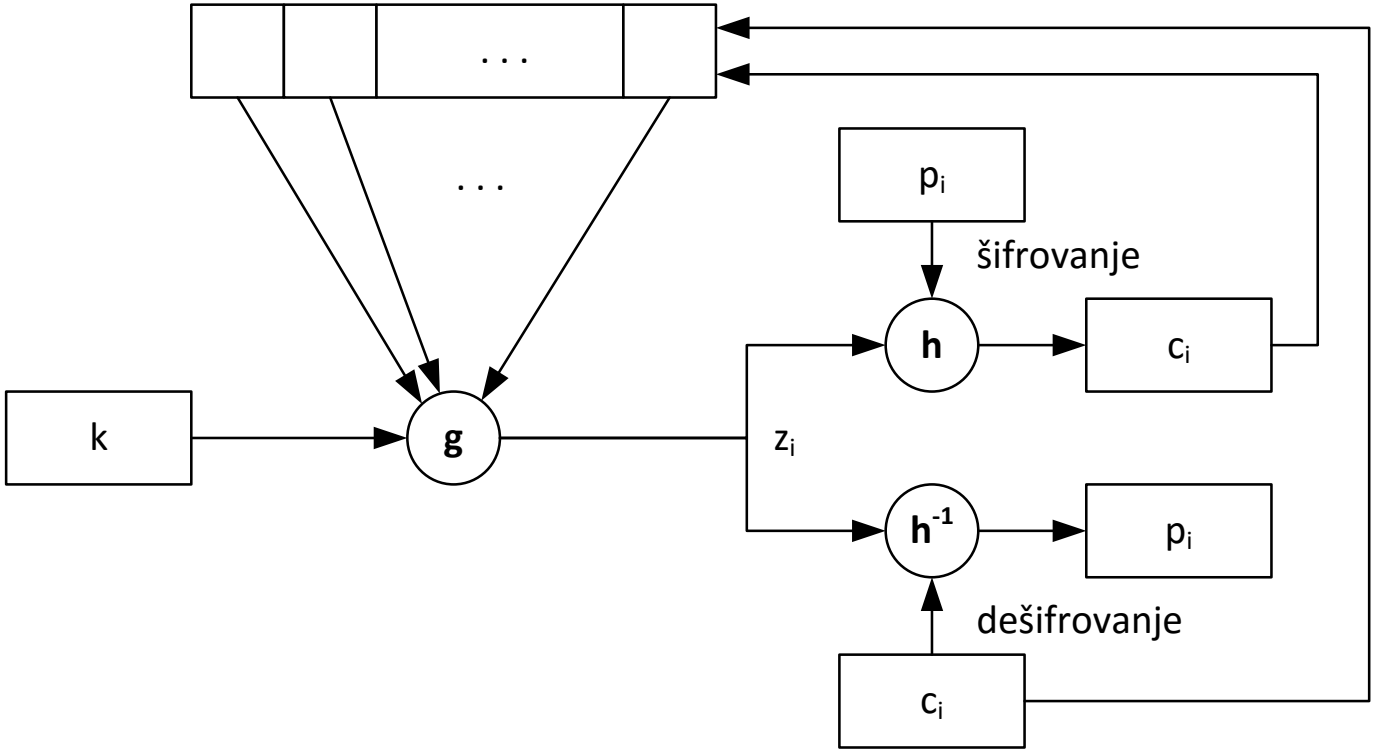
Sinhroni sekvencijalni kriptosistemi



Asinhroni sekvencijalni kriptosistemi

- U slučaju asinhronih (samosinhronišućih) kriptosistema ključna sekvenca se generiše na osnovu ključa i fiksnog broja prethodnih bitova šifrata.
- Za asinhroni kriptosistem važi:
  - $\sigma_i = f(\sigma_{i-t}, \sigma_{i-t+1}, \sigma_{i-t+2}, \ldots \sigma_{i-1})$
  - $z_i = g(\sigma_i, k)$
  - $c_i = h(z_i, m_i)$
- Nedostatak: greška u jednom bitu se propagira duž šifrata.
  - Ako *n* bitova šifrata utiče na generisanje ključne sekvence i napadač modifikuje jedan bit šifrata, *n* sledećih bitova će se neispravno dešifrovati.
- Dobra osobina: gubitak sinhronizacije nije od značaja (sistem će se sam resinhronizovati).

Asinhroni sekvencijalni kriptosistemi

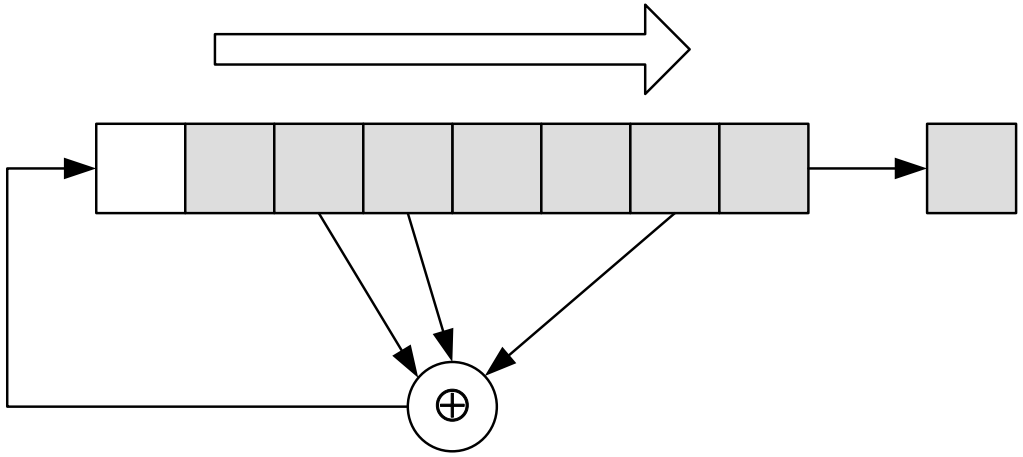


Linearni pomerački registar sa povratnom spregom

- **Pomerački registar sa povratnom spregom** (engl. *feedback shift register, FSR*) je registar kod koga se prelazak u sledeće stanje ostvaruje pomoću sledeće dve operacije:
  - Kružni pomeraj, tj. rotacija bitova registra za 1 bit udesno.
  - Generisanje najznačajnijeg bita (engl. *most significant bit*) na osnovu funkcije povratne sprege čiji su argumenti ostali bitovi registra.
- Najmanje značajan bit (engl. *least significant bit*) je izlaz iz registra.
  - Na taj način se uzastopnim prelaskom u sledeće stanje može generisati sekvenca.
- Perioda FSR je broj bitova nakon kojih sekvenca počinje da se ponavlja.
- Sekvencijalni kriptosistemi se mogu lako implementirati u hardveru ukoliko se koriste ovi registri.

Linearni pomerački registar sa povratnom spregom

- **Linearni FSR** (engl. *linear feedback shift register*, *LFSR*) je pomerački registar kod kog se povratna sprega realizuje pomoću operacije ekskluzivno ILI nad određenim bitovima.
  - Ovi bitovi su poznati pod imenom *tap sequence*.



Linearni pomeračka registar sa povratnom spregom

- Primer:
  - 4-bitni *LFSR*, tap sekvencu čine prvi i četvrti bit, inicijalno stanje 1111.
  - *LFSR* prolazi kroz sledeća stanja: 1111, 0111, 1011, 0101, 1010, 1101, 0110, 0011, 1001, 0100, 0010, 0001, 1000, 1100, 1110, 1111.
  - Izlazna sekvenca *LFSR* je: 111101011001000, a perioda registra 15.
- Idealan *n*-bitni LFSR može se naći u 2^n-1 internih stanja.
  - Teoretski, LFSR može da generiše pseudoslučajnu sekvencu dužine 2^n-1 bitova.
  - Nakon toga se sekvenca ponavlja.
- Pseudoslučajnu sekvencu dužine 2^n nemoguće je generisati.
  - Ukoliko se LFSR napuni nulama na početku, na izlazu se dobija beskonačan niz nula.
- Realni LFSR generiše *m*-sekvencu, odnosno sekvencu dužine *m* bitova, pri čemu je $m<2^n$-1.

RC4

- Tvorac RC4 algoritma je Ron Rivest pa se pretpostavlja da je RC skraćenica od „Rivest Cipher“ ili „Ron's Code“.
- Koristi se u protokolima kao što su SSL, WEP i drugi.
- RC4 je jednostavan sekvencijalni šifarski algoritam koji koristi tabelu stanja.
  - Reč je o *lookup* tabeli koja sadrži permutacije bajtova, tj. brojeve od 0 do 255.
  - Tabeli su pridružena dva pokazivača *i* i *j*, koji takođe mogu da imaju jednu od 256 vrednosti.
- RC4 algoritam obuhvata dve faze:
  - Faza inicijalizacije. Tabela stanja se inicijalizuje primenom odgovarajućeg ključa.
  - Fazu generisanja *keystream*-a.
- Nakon toga sledi šifrovanje primenom XOR funkcije nad originalnom porukom i *keystream*-om.

0	22
1	7
2	134
…	…
i →	77
…	…
j →	101
245	2
255	45

RC4

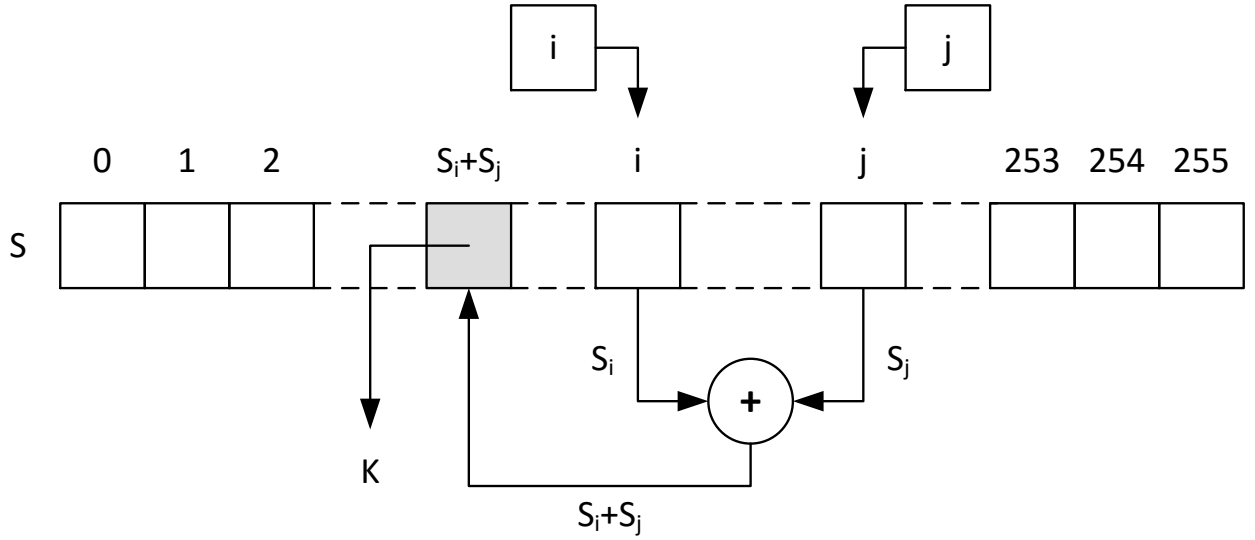
- Neka je zadat ključ key koga čini niz od N bajtova.
- Inicijalizacija tabele stanja *S* odvija se u dve etape:
  - Najpre se *S* popuni redom bajtovima S_0=0, S_1=1, S_2=2, ..., S_{255}=255.
  - Zatim se niz od 256 bajtova K_0, K_1,..., K_{255} puni ciklički ključem. Zatim se sadržaj *S* modifikuje u skladu sa zadatim ključem.

```
for i = 0 to 255
  S[i] = i
  K[i] = key[i (mod N)]
next i
j = 0
for i = 0 to 255
  j = (j + S[i] + K[i]) (mod 256)
  swap(S[i],S[j])
next i
i = j = 0
```

RC4

- Bajt *keystream*-a se generiše tako što:
- Najpre dva elementa tabele S međusobno zamene mesta.
- Zatim se sam bajt *keystream*-a pročita sa odgovarajuće pozicije u tabeli.
- Dobijeni *keystream* se koristi za šifrovanje originalne poruke operacijom XOR.

```
i = (i + 1) (mod 256)
j = (j + S[i]) (mod 256)
swap(S[i], S[j])
t = (S[i] + S[j]) (mod 256)
keystreamByte = S[t]
```



Matematičke osnove kriptografije
(izvod)

Kombinatorika

- **Faktorijel** broja n definiše se na sledeći način:
  $0! = 1! = 1$
  $n! = n \cdot (n\text{-}1) \cdot (n\text{-}2) \cdot \ldots \cdot 2 \cdot 1$
- Neka je $S = \{a_1, a_2, \ldots, a_n\}$.
- **Varijacija bez ponavljanja** klase k ($0 \leq k \leq n$) od n elemenata $V(n,k)$ je bilo koja uređena k-torka različitih elemenata skupa S. Takvih različitih k-torki ima:
  $V(n,k) = n! / (n\text{-}k)!$
- Primer:
  - Dat je skup $S = \{1, 2, 3\}$.
  - Varijacije bez ponavljanja druge klase čine skup {12, 13, 23, 21, 31, 32}.
  - Ima ih ukupno 3!/(3-2)!=6.
- Varijacija klase n od n elemenata (dakle, slučaj kada je $k=n$) je **permutacija** $P(n)$ tih elemenata. Takvih n-torki ukupno ima $n!$:
  $P(n) = V(n,n) = n!$

Kombinatorika

- **Varijacija sa ponavljanjem** klase k ($0 \leq k \leq n$) od n elemenata $V_p(n,k)$ je bilo koja uređena k-torka elemenata skupa S. Elementi u svakoj k-torki mogu se ponavljati, što znači različitih k-torki ima:
  $V_p(n,k) = n^k$
- Primer:
  - Dat je skup $S = \{1, 2, 3\}$.
  - Varijacije s ponavljanjem druge klase čine skup {11, 12, 13, 21, 22, 23, 31, 32, 33}.
  - Ima ih ukupno $3^2=9$.
- **Kombinacija bez ponavljanja** klase k ($0 \leq k \leq n$) od n elemenata $C(n,k)$ je neuređena k-torka različitih elemenata skupa S. Za razliku od varijacija nije dozvoljeno da se ponove dve mogućnosti koje imaju iste elemente, a različit raspored. Različitih k-torki ima ukupno:
  $C(n,k) = n! / ((n-k)! \cdot k!)$
- Primer:
  - Dat je skup $S = \{1, 2, 3\}$.
  - Kombinacije druge klase čine skup {12, 13, 23}.
  - Ima ih ukupno $3!/(2!\cdot 1!)=3$.

Pojam verovatnoće

- Bacamo kocku čije su strane numerisane brojevima od 1 do 6.
- Pojavljivanje jednog od tih brojeva nazivamo ishodom ili **elementarnim događajem** *A*.
- Količnik povoljnih i svih mogućih ishoda za događaj *A* nazivamo **verovatnoćom događaja** *A*: *P(A)*.
- Pošto je broj povoljnih događaja podskup skupa svih mogućih događaja važi $0 \leq P(A) \leq 1$.
- Primer:
  - Koja je verovatnoća da će nakon jednog bacanja kocke pasti broj manji od 5?
  - U ovom slučaju je $S = \{1, 2, 3, 4, 5, 6\}$.
  - Događaj „kocka pokazuje broj manji od 5“ je $A = \{1, 2, 3, 4\}$.
  - Verovatnoća događaja *A* je $4/6 = 2/3$.
- **Siguran događaj** je onaj koji omogućuje da ako jednom bacamo novčić, padne ili pismo ili glava.
  - Verovatnoća sigurnog događaja je 1.
- **Nemoguć događaj** je onaj pri kojem ako jednom bacamo novčić padne i pismo i glava.
  - Verovatnoća nemogućeg događaja je 0.

Pojam verovatnoće

- **Unija događaja** A i B ($A \cup B$) je skup svih ishoda koji pripadaju bar jednom od događaja A i B.
- **Presek događaja** A i B ($A \cap B$) je skup svih ishoda koji pripadaju i događaju A i događaju B.
- **Komplement** događaja A ($\bar{A}$) u odnosu na skup S jeste skup svih ishoda koji ne pripadaju A.
- Dakle, A se ostvaruje onda kada se $\bar{A}$ ne ostvaruje, odnosno A i $\bar{A}$ su suprotni događaji.
- Primeri:
  - Ako je A događaj da je prilikom bacanja novčića palo pismo, $\bar{A}$ je događaj da nije palo pismo.
  - Ako je A događaj da je na kocki pao broj veći od 3 i ako je B događaj da je broj paran:
    - $A \cap B$ obuhvata ishode {4, 6}.
    - $A \cup B$ obuhvata ishode {2, 4, 5, 6}.
- Važe sledeće jednakosti
  - $P(A) + P(\bar{A}) = 1$
  - $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
  - Ako se A i B međusobno onda je verovatnoća unije događaja $P(A \cup B) = P(A) + P(B)$.

Primer: rođendanski paradoks

- Pitanje: koliko učenika treba da bude u odeljenju da bi bilo verovatnije da su bar dvoje rođeni istog datuma nego da nisu?
- Neka ja P(A) verovatnoća da su svi rođeni različitog datuma.
- Verovatnoća koju tražimo je: $P(\bar{A})=1-P(A)$.
  - $P(A) = V(365,k) \,/\, V_p(365,k)$
  - $P(\bar{A}) = 1 - P(A)$
- Tražimo vrednost k za koju je $P(\bar{A})\geq 0{,}5$.
- Za k=23, $P(\bar{A})=0.507>0{,}5$.
- Pomalo iznenađujuće, odgovor je 23 učenika!

Uslovna verovatnoća

- Neka su A i B događaji nekog eksperimenta.
- Ako ostvarivanje jednog od njh ne utiče na ostvarivanje drugog, kažemo da su događaji **nezavisni**. U suprotnom, događaji su zavisni.
- Verovatnoća događaja B pod uslovom da se ostvario događaj A je **uslovna verovatnoća**:
  $P(B|A) = P(A \cap B)/P(A)$
- Primer:
  - Neka je A događaj da je na kocki pao broj veći od 3 i neka je B događaj da je pao paran broj.
  - $P(B|A)$ je verovatnoća da je pao paran broj pod uslovom da je pao broj veći od 3.
  - $P(A \cap B)=2/6$
  - $P(A)=3/6$
  - $P(B|A) = P(A \cap B)/P(A) = 2/3$
- Ako su A i B nezavisni događaji tj. $P(B|A) = P(B)$, onda je verovatnoća preseka događaja:
  $P(A \cap B) = P(A) \cdot P(B)$

Bajesova formula

- Neka je S skup elementarnih događaja nekog eksperimenta i neka je $A_1, A_2, ..., A_n$ potpun sistem događaja tog eksperimenta kod koga se svaka dva događaja uzajamno isključuju.
  - Drugim rečima: $A_1 \cup A_2 \cup ... \cup A_n = S$.
- Za događaj $B \in S$ važi:
  - $B = S \cap B = (A_1 \cup A_2 \cup ... \cup A_n) \cap B = (A_1 \cap B) \cup (A_2 \cap B) \cup ... \cup (A_n \cap B)$
  - $P(B) = P(A_1 \cap B) + P(A_2 \cap B) + ... + P(A_n \cap B)$
- Kako se događaji $A_1, A_2, ..., A_n$ isključuju, i događaji $A_1 \cap B, A_2 \cap B, ..., A_n \cap B$ se isključuju, pa je:
  - $P(B) = P(A_1)P(B|A_1) + P(A_2)P(B|A_2) + ... + P(A_n)P(B|A_n) = \sum P(A_k)P(B|A_k)$
- Iz jednakosti $P(A_k \cap B) = P(A_k)P(B|A_k) = P(B)P(A_k|B)$ sledi:
  - $P(A_k|B) = P(A_k)P(B|A_k) / P(B)$
- Na osnovu toga dobija se **Bajesova formula** (koja svoju primenu, između ostalog, nalazi i u filtriranju neželjene elektronske pošte):
  - $P(A_k|B) = P(A_k)P(B|A_k) / \sum P(A_k)P(B|A_k)$

Slučajna promenljiva i matematičko očekivanje

- **Slučajna promenjiva** X je funkcija koja nekom događaju dodeljuje neki realan broj.
- **Raspodela slučajne promenljive** pokazuje verovatnoću sa kojom slučajna promenljiva uzima određenu vrednost.
- Primer: neka se novčić baca tri puta.
  - Svaki od ishoda {GGG, GGP, GPG, PGG, GPP, PGP, PPG, PPP} ima verovatnoću 1/8.
  - Neka je X broj palih pisama.
  - Na primer: $X(\text{GGP}) = X(\text{GPG}) = X(\text{PGG}) = 1$, dok je $X(\text{PPP}) = 3$.
  - Raspodela slučajne promenjive X data je sa:

0	1	2	3
1/8	3/8	3/8	1

- Neka je $\begin{matrix} x_1 & x_2 & \dots & x_m \\ p_1 & p_2 & \dots & p_m \end{matrix}$ raspodela slučajne promenljive X pri čemu je $p_1 + p_2 + \dots + p_m = 1$.
- **Matematičko očekivanje** slučajne promenjive X je $E(X) = x_1 \cdot p_1 + x_2 \cdot p_2 + \dots + x_m \cdot p_m$.

Pojam neodređenosti, entropije i informacije

- Ako posmatramo jedan slučajan događaj A čija je verovatnoća p = P(A) onda se njegova **neodređenost** može meriti nekom monotono opadajućom funkcijom od p.
- Ukoliko verovatnoća odstupa od „sredine“, utoliko je neodređenost manja.
  - Za takvu funkciju možemo uzeti $\log_2$ (logaritam sa osnovom 2).
  - Na taj način neodređenost događaja čija je verovatnoća 0.5 iznosi 1.
- „Razbijanje“ događaja S preko događaja A_1, A_2, ..., A_n označićemo sa A.
- Neodređenost nekog događaja A_i iz A je $\log_2 p_i$, a verovatnoća pojavljivanja te verovatnoće p_i.
- To znači da $-\log_2 p_i$ možemo posmatrati kao moguće vrednosti jedne slučajne promenljive.
- Zato za meru neodređenosti uzimamo srednju ili očekivanu vrednost: $H(A) = \sum p_i \cdot \log_2 p_i$.
- Na primer, opit koji ima dva jednakoverovatna ishoda A_1 i A_2 ima **entropiju**:
  - $H(A) = -0{,}5 \cdot \log_2 0{,}5 - 0{,}5 \cdot \log_2 0{,}5 = 1$.
- Entropija opita sa raspodelom $\begin{matrix} 0 & 1 & 2 \\ 1/4 & 1/2 & 1/4 \end{matrix}$ iznosi:
  - $H(A) = -0{,}25 \cdot \log_2 0{,}25 - 0{,}5 \cdot \log_2 0{,}5 - 0{,}25 \cdot \log_2 0{,}25 = 0{,}452$

Pojam neodređenosti, entropije i informacije

- **Uslovnu entropiju** $H(A|B_i)$ opita A u odnosu na događaj B_i definišemo tako što umesto verovatnoća $p_i = P(A_i)$ uvodimo uslovne verovatnoće $P(A_i|B_i)$, i=1,2,..,n.
  - $H(A|B_i) = -\sum P(A_i|B_i) \cdot \log_2 P(A_i|B_i)$.
- Neka je $B_1, B_2, ..., B_m$ razbijanje opita B.
- Uslovnu entropiju opita A u odnosu na B definišemo kao matematičko očekivanje:
  - $H(A|B) = -\sum P(B_i)H(A|B_i)$
- **Informacija** $I(A|B)$ koju o opitu A nosi opit B definiše se kao:
  - $I(A|B) = H(A) - H(A|B)$
- Ako su opiti A i B nezavisni, onda je informacija koju o opitu A nosi B jednaka nuli jer je tada $H(A) = H(A|B)$.
- Ako je $H(A|B)$=0, tj. opit B „iscrpljuje“ svu neodređenost opita A, tada je informacija $I(A|B)$ maksimalna i jednaka entropiji samog opita A.

Kongruencija

- Najveći zajednički delilac NZD(a,b) brojeva a i b ($a \neq 0$, $b \neq 0$) je najveći broj koji deli a i b.
  - Primer: NZD(4,6)=2, NZD(4,8)=4, NZD(1,x)=1, gde je x bilo koji broj.
- Brojevi a i b su **uzajamno prosti** ako je NZD(a,b) = 1.
- Ostatak pri deljenju a sa b obeležićemo sa a mod b.
- Euklidov algoritam služi za računanje najmanjeg zajedničkog delioca brojeva a i b:
  - Ako je b=0, onda je NZD(a,b) = $|a|$
  - Ako je $b \neq 0$, onda je NZD(a,b) = NZD($|b|$, a mod $|b|$).
- Primer:
  - NZD(100,35) = NZD(35, 100 mod 35) =
  - = NZD(35,30) = NZD(30, 35 mod 30) =
  - = NZD(30,5) = NZD(5, 30 mod 5) =
  - = NZD(5,0) = 5

Kongruencija

- Broj a je kongruentan sa celim brojem b po modulu n ako n deli $a–b$.
- Pišemo: $a \equiv b \pmod{n}$.
- Drugim rečima, a i b su oni **kongruentni po modulu** n ako daju isti ostatak pri deljenju sa n.
- Primer:

  $24 \equiv 9 \pmod 5$ jer $24 - 9 = 3 \cdot 5$

  $-11 \equiv 17 \pmod 7$ jer $-11 - 17 = -4 \cdot 7$

Za svako celobrojno a, a_1, b, b_1, c važi:

- $a \equiv a \pmod n$
- Ako je $a \equiv b \pmod n$, onda je $b \equiv a \pmod n$
- Ako je $a \equiv b \pmod n$ i $b \equiv c \pmod n$, onda je $a \equiv c \pmod n$
- Ako je $a \equiv a_1 \pmod n$ i $b \equiv b_1 \pmod n$, onda je:
  - $a + b \equiv a_1 + b_1 \pmod n$
  - $a \cdot b \equiv a_1 \cdot b_1 \pmod n$.

Kongruencija

- **Kineska teorema ostatka**.
  - Neka su n_1, n_2, ..., n_k uzajamno prosti brojevi, tj. neka važi NZD(n_1, n_2, ..., n_k) = 1.
  - Sistem kongruencija: $x \equiv a_1 \pmod{n_1}$, $x \equiv a_2 \pmod{n_2}$, ... $x \equiv a_k \pmod{n_k}$ ima jedinstveno rešenje po modulu $n_1 \cdot n_2 \cdot \ldots \cdot n_k$.
  - Specijalno: ako je NZD(n_1,n_2)=1, par kongruencija $x \equiv a_1 \pmod{n_1}$ i $x \equiv a_2 \pmod{n_2}$ ima jedinstveno rešenje $x \equiv a \pmod{n_1 \cdot n_2}$.
  - Primer: par $x \equiv 3 \pmod 7$ i $x \equiv 7 \pmod{13}$ ima jedinstveno rešenje $x \equiv 59 \pmod{91}$.
- **Klasa ekvivalencije** u odnosu na ceo broj a predstavlja skup svih celih brojeva kongruentnih sa a po modulu n.
  - Primer: u odnosu na broj 2 postoje dve klase ekvivalencije: parni brojevi (daju ostatak 0) i neparni brojevi (daju ostatak 1).
- Svaki broj a je kongruentan po modulu n nekom broju od 1 do n–1.
- Brojevi iz tog skupa reprezentuju sve ostale brojeve i nazivaju se najmanjim ostacima broja a po modulu n.

Kongruencija

- Neka je $Z_n = \{0, 1, ..., n\text{-}1\}$.
  - Sabiranje, oduzimanje, množenje u Z_n obavlja se po modulu n.
  - Primer: u Z_{25} je $13+16 = 29 \equiv 4 \pmod{25}$.
- **Inverzni element** elementa a iz Z_n po modulu n je ceo broj x iz Z_n takav da važi $a \cdot x \equiv 1 \pmod{n}$.
  - Inverzni element elementa a iz Z_n obeležava se sa a^{-1}.
- Deljenje a i b po modulu n predstavlja proizvod a i inverznog elementa od b ako on postoji:
  $a/b \pmod{n} = a \cdot b^{-1} \pmod{n}$
- Element a iz Z_n je invertibilan, tj. ima inverzni element ako i samo ako je NZD(a,n)=1.
- Primer:
  - Invertibilni elementi u Z_9 su 1, 2, 4, 5, 7, 8.
  - Broj 7 je invertibilni element broja 4 ($7^{-1}=4$) jer je $4 \cdot 7 \equiv 1 \pmod{9}$.
- Neka je d = NZD(a,n).
  - Jednačina $a \cdot x = b \pmod{n}$ ima rešenje ako i samo ako d deli b i tada postoji tačno jedno rešenje za d između 0 i n-1.

Grupe, prstenovi i polja

- Skup G je **zatvoren** za neku operaciju „●“ ako za bilo koja dva elementa a i b iz G rezultat operacije $a●b$ takođe pripada G.
- Primer:
  - Skup prirodnih brojeva N je zatvoren za operaciju sabiranja.
  - Skup nije zatvoren za operaciju oduzimanja ($2 \in \mathrm{N}$ i $3 \in \mathrm{N}$, ali $2-3=-1 \notin \mathrm{N}$).
- Za operaciju „●“ kažemo da je **asocijativna** ako za bilo koje brojeve a, b, i c iz nekog skupa G važi $(a●b)●c = a●(b●c)$.
- Primer:
  - sabiranje u skupu N je asocijativno ali oduzimanje nije: $1-(2-3) \neq (1-2)-3$
- Ukoliko je data operacija „●“, element e nazivamo **neutralnim elementom** ako za svako a iz nekog skupa G važi $a●e = e●a = a$.
- Primer:
  - Ako je operacija „●“ sabiranje, neutralni element je 0 jer je $a+0 = 0+a = a$.
  - Ukoliko je „●“ množenje, neutralni element je 1 jer je $a \cdot 1 = 1 \cdot a = a$.

Grupe, prstenovi i polja

- Algebarska struktura (G,●) je **grupa** ako važi sledeće:
  - Zatvorenost za binarnu operaciju
  - Asocijativnost
  - Postojanje neutralnog elementa
  - Postojanje inverznog elementa.
- Grupa je **Abelova** ako još važi a●b=b●a (komutativnost).
- Primeri:
  - Skup celih brojeva Z i operacija sabiranja (Z,+) jeste grupa.
  - Skup celih brojeva Z i operacija množenja (Z,·) nije grupa jer postoje inverzni elementi koji ne pripadaju skupu Z (na primer, inverzni element broja 3 je 1/3 koja ne pripada Z).
- **Multiplikativnu grupu** Z_n^* čine svi brojevi od 1 do n koji su uzajamno prosti sa n.
  - $Z_n^* = \{a \in Zn \mid NZD(a,n)=1\}$.
  - Specijalno: ako je n prost broj, onda je $Z_n^* = \{a \mid 0<a<n\}$.

- **Ojlerova funkcija** $\varphi(n)$ je broj svih celih brojeva a u skupu $\{1, 2, ..., n\}$ sa osobinom NZD$(a,n) = 1$.
- **Red grupe** Z_n^* definiše se kao broj elemenata grupe: $|Z_n| = \varphi(n)$, gde je φ Ojlerova funkcija.
- Ojlerova teorema: ako je $a \in Z_n^*$, onda važi $a^{\Phi(n)} \equiv 1 \pmod{n}$.
- Takođe: ako je $r \equiv s \pmod{\varphi(n)}$, onda za svako a važi $a^r = a^s \pmod{n}$.
- Drugim rečima kad radimo po modulu n, stepen može biti redukovan po modulu $\Phi(n)$.
- Primer:
  - Pošto je 2 iz Z_5 i pošto je $\varphi(Z_5)=4$, sledi $2^4 \equiv 1 \pmod 5$. Takođe, $2^4 \equiv 2^8 \pmod 5$.
- Specijalan slučaj Ojlerove teoreme predstavlja **Mala Fermaova teorema**.
  - Ako je NZD$(a,p)=1$, tada važi: $a^{p-1} \equiv 1 \pmod{p}$.
- Red elementa $a \in Z_n$ predstavlja najmanji pozitivan broj t takav da važi: $a^{t-1} \equiv 1 \pmod{t}$.
- Ako je t red elementa $a \in Z_n^*$ i ako je $a^s = 1 \pmod{n}$, onda t deli s. Specijalno, t deli $\varphi(n)$.
- Primer:
  - $Z_{21}^* = \{1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20\}$.
  - $\varphi(21) = \varphi(7) \cdot \varphi(3) = 6 \cdot 2 = 12 = |Z_{21}^*|$.

- Grupa G je **ciklična** ako postoji element $a \in G$ takav da za svako $b \in G$ postoji celobrojno i tako da važi $b = a^i$.
  - Drugim rečima, grupa je ciklična ako se svi elementi mogu predstaviti različitim stepenima jednog istog elementa.
  - Element a generiše celu grupu i naziva se **generator** ciklične grupe G.
- Neka je $a \in Z_n^*$. Ako je red elementa a jednak $\varphi(n)$ tada a zovemo generatorom grupe Z_n^*.
- Ako grupa Z_n^* ima generator, onda kažemo da je ciklična.
- Primer:
  - Grupa Z_{21}^* nije ciklična jer ne sadrži element reda $\varphi(n)=12$.
  - Grupa Z_{25}^* jeste ciklična jer ima generator $a=2$.

Grupe, prstenovi i polja

- **Prsten** (R,+,*) je algebarska struktura sastavljena od skupa R i operacija „+“ i „*“ (obično sabiranja i množenja) koja ima sledeće osobine:
  - (R,+) je Abelova grupa sa neutralnim elementom 0
  - operacija „*“ je asocijativna
  - za svako a iz R postoji neutralni element 1 za „*“
  - operacija „*“ je distributivna u odnosu na „+“: $a*(b+c) = (a*b)+(a*c)$
- Ako za element a prstena R postoji $b \in R$ tako da važi $a*b = 1$, onda b nazivamo **multiplikativnim inverznim elementom** elementa a.
- Prsten je **komutativan** ukoliko za svako $a,b \in R$ važi $a*b = b*a$.
- Komutativni prsten kod koga svaki element (osim nula elementa) ima multiplikativni inverzni element naziva se **polje**. Z_n je polje ako i samo ako je n prost broj.
  - Primer: (R,+, ·) je polje ali (Z,+, ·) nije jer jedino elementi 1 i -1 imaju multiplikativni inverz.
- **Konačno polje *F*** je polje koje sadrži konačan broj elemenata.
- **Red** konačnog polja F je broj elemenata polja F.

Polinomi u konačnim poljima

- Neka je $Z_p[x]$ skup svih polinoma čiji su koeficijenti redukovani po modulu p.
- Neka su $f(x)$, $g(x)$ i $q(x)$ polinomi iz skupa $Zp[x]$.
- Kažemo: $f(x)$ deli $g(x)$, tj. $f(x)|g(x)$, ako postoji $q(x)$ tako da je $g(x) = q(x) \cdot f(x)$.
- Za $f(x)$ iz $Zp[x]$ **stepen polinoma** deg(f) predstavlja eksponent člana najvišeg stepena u polinomu.
- Za polinome važu sledeće:
  - $g(x) \equiv h(x) \pmod{f(x)}$, ako $f(x)|(g(x)-h(x))$.
  - Ako je deg(f)=n i pri deljenju polinoma g(x) sa f(x) dobijemo ostatak $r(x)$, onda je stepen polinoma deg(r)<n jer je $g(x) = q(x)f(x) + r(x)$.
- Polinom $f(x)$ iz $Z_p[x]$ je **nesvodljiv** ako ne postoje polinomi $f_1(x)$, $f_2(x)$ iz $Z_p[x]$ takvi da važi sledeće: $f(x) = f_1(x) \cdot f_2(x)$, gde su deg($f_1$)>0 i deg($f_2$)>0.
  - Drugim rečima, $f(x)$ je nesvodljiv ako je deljiv samo sa sobom i sa 1.
  - Nesvodljiv polinom među polinomima isto je što i prost broj među brojevima.

Polinomi u konačnim poljima

- Primer sabiranja polinoma u GF(2^8):
  - $f(x) = (x^7+x^6+x^5+1)$
  - $g(x) = (x^6+x^5+x+1)$
  - $h(x) = f(x) + g(x) = (x^7+x)$
- Množenje u konačnom polju GF(2^8) odgovara množenju polinoma po modulu *m(x)*, gde je $m(x)$ nedeljiv polinom osmog stepena.
- Množenje definisano na ovakav način je asocijativno, a multiplikativna konstanta je 1.
- Množenje i sabiranje su distributivne operacije.
- Kao nesvodljiv polinom može se koristiti, na primer, $m(x) = x^8+x^4+x^3+x+1$.
  - Ovaj polinom se koristi u AES algoritmu.
- Primer množenja:
  - $(x^6+x^4+x^2+x+1) \cdot (x^7+x+1) \bmod (x^8+x^4+x^3+x+1) =$
  - $= (x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1) \bmod (x^8+x^4+x^3+x+1) =$
  - $= x^7+x^6+1$

Literatura

1. D. Pleskonjić, N. Maček, B. Đorđević, M. Carić (2007): Sigurnost računarskih sistema i mreža. Mikro knjiga, Beograd.
2. M. Veinović, S. Adamović (2013): Kriptologija 1. Univerzitet Singidunum, Beograd.
3. M. Milosavljević, S. Adamović (2014): Kriptologija 2. Univerzitet Singidunum, Beograd.
4. M. Stamp (2006): Information Security. John Wiley and Sons.
5. C. E. Shannon (1949): The communication theory of secrecy systems. Bell Labs Technical Journal, 28(4), pp. 656-715.
6. M. Živković (2012): Kriptografija (skripta). Matematički fakultet, Beograd.

Hvala na pažnji

Pitanja su dobrodošla.