Protočna šifra RC4

Krnjak, Jelena

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Jelena Krnjak

PROTOČNA ŠIFRA RC4

Diplomski rad

Voditelj rada:
prof. dr. sc. Andrej Dujella

Zagreb, 2016.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Mami i tati…
Hvala što ste vjerovali u mene.

Sadržaj

Uvod

Od samih početaka civilizacije ljudi su bili fascinirani tajnama. Osim što je tajne trebalo očuvati, pojavila se potreba za tajnim komuniciranjem. Ovdje na scenu stupa kriptografija, znanstvena disciplina specijalizirana za metode sigurne (tajne) komunikacije. Smatra se da začetke kriptografije možemo pronaći već kod starih Egipćana (oko 2000. g. pr. Kr.). Od tog je vremena kriptografija mnogo napredovala, razvijajući se s trenutnim potrebama čovječanstva. Danas je jedna od najvažnijih primjena kriptografije u očuvanju računalne sigurnosti.

U ovom radu obrađena je sinkronizirana protočna šifra RC4. Za razliku od blokovnih šifri koje obrađuju jedan po jedan blok otvorenog teksta, protočne šifre obrađuju jedan po jedan element otvorenog teksta koristeći pseudoslučajno generirani niz ključeva. RC4 je jedna od najpopularnijih i najboljih suvremenih protočnih šifri te ima mnoge praktične primjene. Algoritam RC4 sastoji se od dva dijela: KSA i PRGA-e. Sam algoritam izuzetno je jednostavan te je zbog svoje jednostavnosti zainteresirao mnoge kriptografe te postoje mnogi radovi koji se bave analizom upravo ove šifre. Iako je od njenog kreiranja prošlo gotovo trideset godina, RC4 ostaje u fokusu proučavanja mnogih. Ovaj rad donosi pregled dosadašnjih rezultata u analizi KSA i PRGA-e.

Poglavlje 1

Osnovni pojmovi

1.1 Osnovni pojmovi

Kriptografija je znanstvena disciplina koja proučava metode slanja poruka u takvom obliku da ih samo onaj kome su namijenjene može pročitati. Riječ kriptografija dolazi od grčkih riječi *kriptos* što znači tajna i *grafein* što znači pisati pa je zato doslovno možemo prevesti kao „tajnopis".

Glavni cilj kriptografije je omogućiti neometano komuniciranje osobe A (pošiljaoca, koji se u literaturi obično naziva Alice) i osobe B (primaoca, koji se u literaturi obično naziva Bob) putem nesigurnog komunikacijskog kanala tako da treća osoba C (protivnik, koji se u literaturi obično naziva Eva, Oskar ili Trudy) ne može razumijeti njihove poruke. Poruka koju osoba A želi poslati osobi B naziva se **otvoreni tekst** (eng. *plaintext*). Osoba A prvo transformira, tj. šifrira otvoreni tekst koristeći unaprijed dogovoreni **ključ** (eng. *key*) te tako dobiva **šifrat** (eng. *ciphertext*). Zatim Alice šalje šifrat nesigurnim komunikacijskim kanalom, primjerice telefonskom linijom ili računalnom mrežom. Eva presreće ovako poslan šifrat, no ne može odrediti otvoreni tekst jer ne zna ključ. Primijetimo da razmatramo samo slučaj u kojem Eva može presresti šifrat, ali ga ne može na bilo koji način izmijeniti ili spriječiti njegov daljnji prolaz komunikacijskim kanalom. Nakon što Bob primi šifrat i pomoću ključa ga **dešifrira**, dobiva otvoreni tekst. Prilikom (de)šifriranja može se koristi jedan, niti jedan ili više od jednog ključa.

Kriptoanaliza ili **dekriptiranje** je znanstvena disciplina koja proučava postupke otkrivanja otvorenog teksta bez poznavanja ključa. Kriptoanaliza uključuje i otkrivanje ključa uz poznavanje otvorenog teksta i/ili šifrata.

Kriptologija obuhvaća kriptoanalizu i kriptografiju. Kriptologija se bavi proučavanjem i definiranjem metoda zaštite informacija te pronalaskom metoda za otkrivanje šifriranih

podataka.

Šifra ili **kriptografski algoritam** je matematička funkcija koja se koristi za šifriranje i dešifriranje. Riječ je o dvije funkcije od kojih je jedna za šifriranje, a druga za dešifriranje. Funkcija šifriranja (u oznaci e_K) ima dva argumenta: element(e) otvorenog teksta (koji su najčešće slova, bitovi ili grupe slova ili bitova) i ključ. Konačan skup svih mogućih elemenata otvorenog teksta označavamo s $\mathcal{P}$. Funkcija dešifriranja (u oznaci d_K) kao argumente ima šifrat i ključ. Konačan skup svih mogućih osnovnih elemenata šifrata označavamo s C. Skup svih mogućih vrijednosti ključeva naziva se **prostor ključeva** (eng. *keyspace*). Konačan skup svih mogućih ključeva označavamo s $\mathcal{K}$.

Kriptosustav se sastoji od šifre, skupa svih mogućih otvorenih tekstova, skupa svih mogućih šifrata, prostora ključeva te funkcija šifriranja i dešifriranja.
Označimo s $\mathcal{E}$ skup svih funkcija šifriranja, a s $\mathcal{D}$ skup svih funkcija dešifriranja.
Dakle, kriptosustav možemo definirati kao uređenu petorku ($\mathcal{P}$, C,$\mathcal{K}$, $\mathcal{E}$, $\mathcal{D}$). Tada za svaki $K \in \mathcal{K}$ postoji funkcija šifriranja $e_K \in \mathcal{E}$, $e_K : \mathcal{P} \to C$ koja otvorenom tekstu pridružuje odgovarajući šifrat te funkcija dešifriranja $d_K \in \mathcal{D}$, $d_K : C \to \mathcal{P}$ koja šifratu dobivenom djelovanjem funkcije e_K pridružuje polazni otvoreni tekst. Dakle, vrijedi

$$d_K(e_K(x)) = x, \text{ za svaki } x \in \mathcal{P}.$$

Zaključujemo da je funkcija e_K injekcija. U suprotnom, postojala bi mogućnost da e_K dvama različitim otvorenim tekstovima a i b pridruži jednaki šifrat c. Primaoc tada ne bi znao koji od otvorenih tekstova (a ili b) treba dobiti postupkom dešifriranja, tj. je li $d_K(c) = a$ ili $d_K(c) = b$. Dakle, tada $d_K(c)$ nije definirano.
Ako vrijedi $\mathcal{P} = C$, onda je funkcija e_K permutacija.

Definicija 1.1.1. *Neka je zadan skup S. Svaku bijekciju $p : S \to S$ nazivamo permutacija skupa S.*

Ako skup S ima n elemenata, onda, bez smanjenja općenitosti, umjesto $S = \{a_1, a_2, \ldots, a_n\}$ možemo pisati $S = \{1, 2, \ldots, n\}$.
Uobičajeno je permutaciju S pisati matrično

$$S = \begin{pmatrix} 1 & 2 & 3 & \ldots & n-1 & n \\ p(1) & p(2) & p(3) & \ldots & p(n-1) & p(n) \end{pmatrix}$$

1.2 Klasifikacija kriptosustava

Kriptosustave obično klasificiramo s obzirom na tri kriterija:

1. S obzirom na **tip operacija koje se koriste pri šifriranju** kriptosustave dijelimo na supstitucijske šifre u kojima se svaki element otvorenog teksta zamjenjuje s nekim drugim elementom te transpozicijske šifre u kojima se elementi otvorenog teksta permutiraju.

2. S obzirom na **način na koji se obrađuje otvoreni tekst** razlikujemo blokovne šifre kod kojih se obrađuje jedan po jedan blok elemenata otvorenog teksta koristeći isti ključ K te protočne šifre (eng. *stream cipher*) koje elemente otvorenog teksta obrađuju jedan po jedan koristeći paralelno generirani niz ključeva (eng. *keystream*).

3. S obzirom na **tajnost i javnost ključeva**, kriptosustave dijelimo na dva tipa. Simetrični ili konvencionalni kriptosustavi su oni kriptosustavi kod kojih se ključ za dešifriranje može izračunati poznavajući ključ za šifriranje, i obratno.
   Najčešće su ključevi za šifriranje i dešifriranje jednaki.
   Sigurnost ovakvih kriptosustava je u tajnosti ključa pa se zato nazivaju kriptosustavi s tajnim ključem.
   Kriptosustavi s javnim ključem ili asimetrični kriptosustavi su konstruirani tako da se ključ za dešifriranje ne može u razumnom vremenu izračunati iz ključa za šifriranje. Ključ za šifriranje je u ovakvim kriptosustavima javno dostupan pa se zato naziva javni ključ. Svatko može šifrirati poruku koristeći javni ključ, no jedino osoba koja posjeduje odgovarajući tajni ključ za dešifriranje može dešifrirati poruku.

1.3 Kriptoanalitički napadi

Alice i Bob komuniciraju šifriranim porukama jer ne žele da netko drugi sazna pojedinosti njihove komunikacije, a pretpostavljaju da je Eva sposobna presresti njihove poruke. Eva koristi kriptoanalizu da bi otkrila sadržaj poruka. Možemo pretpostaviti da Eva zna koji kriptosustav koriste Alice i Bob pri komunikaciji (čak i ako to nije istina, nakon provjere nekoliko kriptosustava Eva će sigurno saznati koji kriptosustav se koristi, a to joj neće predstavljati prevelik problem). Pretpostavka da kriptoanalitičar zna koji kriptosustav se koristi naziva se *Kerckhoffsovo*[1] *načelo*.

[1] Auguste Kerckhoffs (1835. – 1903.) je nizozemski kriptoanalitičar i lingvist. Najpoznatije djelo mu je skup eseja objavljenih u časopisu le Journal des Sciences Militaires pod naslovom Vojna kriptografija. U njima je dao pregled (tadašnjih) najboljih suvremenih šifri te je iznio mnoge praktične savjete za poboljšanje francuskog korištenja šifri, kao i poznatih šest principa za praktično dizajniranje kriptosustava.

Kriptoanalitičke napade razlikujemo na četiri osnovne razine.

1. Kriptoanalitičar posjeduje **samo šifrate** nekoliko poruka šifriranih pomoću istog algoritma. Kriptoanalitičarov je zadatak otkriti otvoreni tekst što većeg broja poruka ili ključ kojim su poruke šifrirane.

2. Kriptoanalitičar posjeduje šifrat poruke, ali mu je **poznat** i njemu odgovarajući **otvoreni tekst**. Sada je zadatak kriptoanalitičara otkrivanje ključa ili algoritma za dešifriranje poruka šifriranih zadanim ključem.

3. Kriptoanalitičar može **odabrati otvoreni tekst** koji će biti šifriran te dobiti njegov šifrat. Ovakva vrsta napada jača je od prethodnoga, ali je i manje realistična.

4. Kriptoanalitičar ima pristup alatu za dešifriranje te može **odabrati šifrat** i dobiti pripadni otvoreni tekst. Kriptoanalitičarov je zadatak otkriti tajni ključ za dešifriranje. Ovaj napad je tipičan kod kriptosustava s javnim ključem.

5. Iako ne pripadaju u čisto kriptoanalitičke napade, napadi pomoću **potkupljivanja, ucjene, krađe** i ostalih sličnih (uglavnom ilegalnih) metoda česti su i efikasni u slučaju primjene s nekim od prethodno navedenih napada.

Dakle, Alice i Bob znaju da će njihov komunikacijski kanal biti ugrožen pa je logično da žele odabrati što sigurniji kriptosustav. Stoga se postavlja pitanje što je uopće siguran kriptosustav. Razlikujemo nekoliko ideja sigurnog kriptosustava.

1. **Savršena sigurnost**
   Kriptosustav se smatra savršeno sigurnim ako ga je nemoguće razbiti, čak ni uz pretpostavku da kriptoanalitičar na raspolaganju ima neograničene resurse (vrijeme i računalnu tehnologiju).

2. **Računalna sigurnost**
   Za kriptosustav kažemo da je računalno siguran ako je najboljem poznatom algoritmu za njegovo razbijanje potrebno barem N operacija, gdje je N unaprijed zadani, velik broj.

3. **Dokaziva sigurnost**
   Kriptosustav je dokazivo siguran ako je jednako težak za razbijanje kao i neki poznati problem koji se smatra teškim. Važno je napomenuti da kriptosustav baziran na teško rješivom problemu ne garantira sigurnost.
   Primjerice, može se dogoditi da je složenost problema u najgorem slučaju eksponencijalna, no složenost u nekom dijelu problema ili u prosječnoj situaciji je polinomijalna.

Poglavlje 2

Protočne šifre

2.1 Savršeno sigurni kriptosustavi

Pojam savršene sigurnosti uveo je Claude Shannon[1] 1949. godine. Savršeno siguran kriptosustav je onaj u kojem šifrat ne daje nikakvu informaciju o otvorenom Neka je vjerojatnost pojavljivanja otvorenog teksta $x \in \mathcal{P}$ jednaka $P(x) = p_x$. Tada je, za svaki fiksni $x \in \mathcal{P}$ i za sve $y \in C$, vjerojatnost da je x otvoreni tekst ako znamo da je $y \in C$ šifrat jednaka $P(x|y) = p_x$.

Teorem 2.1.1. *(**Bayesov teorem**) Neka je $H_1, H_2, H_3, \ldots, H_i, \ldots, H_n$ potpun sustav događaja i neka vrijedi $P(H_1) \neq 0, P(H_1) \neq 0, \ldots, P(H_i) \neq 0, \ldots, P(H_n) \neq 0$. Neka je $A \in \mathcal{F}$ i $P(A) \neq 0$. Tada vrijedi:*

$$P(H_i|A) = \frac{P(A|H_i) \cdot P(H_i)}{\sum_{k=1}^{n} P(A|H_k) \cdot P(H_k)}.$$

Koristeći Bayesov teorem, dobiva se da je $P(y|x) = \frac{p_x \cdot p_y}{p_x \cdot p_y + p_x \cdot (1-p_y)} = p_y$.
Bez smanjenja općenitosti možemo pretpostaviti da je $P(y) > 0$, za svaki $y \in C$ (inače izbacimo y iz C). Dakle, y je šifrat nekog otvorenog teksta. Zato postoji $K \in \mathcal{K}$ takav da je $e_K(x) = y$. Dakle, ključeva ima barem onoliko koliko ima šifrata. Zato zaključujemo da u savršeno sigurnom kriptosustavu vrijedi $|\mathcal{K}| \geq |C|$. Svaka funkcija šifriranja je injekcija pa zato sigurno vrijedi $|C| \geq |\mathcal{P}|$.
Za $|\mathcal{P}| = |C| = |\mathcal{K}|$ kriptosustav je savršeno siguran ako i samo ako je svaki ključ korišten s istom vjerojatnošću te ako za svaki $x \in \mathcal{P}, y \in C$ postoji jedinstveni $K \in \mathcal{K}$ takav da vrijedi $e_K(x) = y$.

[1] Claude Elwood Shannon (1916. – 2001.) je američki matematičar, inženjer elektrotehnike i kriptograf. Utemeljitelj je teorije informacija kao znanstvene discipline.

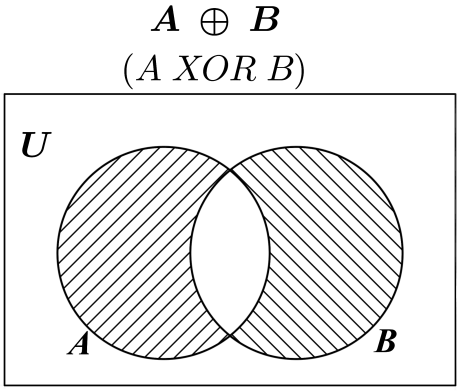
2.2 Jednokratna bilježnica

Vjerojatno najpoznatija realizacija savršeno sigurnog kriptosustava je tzv. **jednokratna bilježnica** (eng. *one-time pad*). Opišimo način djelovanja ove šifre.

Alice želi Bobu poslati poruku koristeći jednokratnu bilježnicu. Alice prvo odabrani otvoreni tekst P zapiše u binarnom obliku (tj. kao niz bitova (nula i jedinica)). Zatim `slučajnim izborom` odabire ključ K u binarnom zapisu koji je jednako dugačak kao i otvoreni tekst. Alice računa šifrat kao $C = P \oplus K$, gdje je $\oplus$ oznaka za operaciju "ekskluzivno ili"(XOR). Bob zna tajni ključ K te, nakon što primi Alicein šifrat, računa otvoreni tekst kao $P = C \oplus K = (P \oplus K) \oplus K = P$.

XOR

"**Ekskluzivno ili**"(XOR) logička je operacija koja kao rezultat daje istinu ako su joj ulazni podatci različiti (jedan je istinit, a drugi lažan). Ako su ulazi jednaki, rezultat je lažan. Na slici 2.1 se nalazi grafički prikaz operacije XOR pomoću Vennovih dijagrama.



Slika 2.1: Grafički prikaz operacije XOR

Djelovanje funkcije XOR dano je tablicom 2.1.

$\oplus$	**0**	**1**
0	0	1
1	1	0

Tablica 2.1: Tablica istinitosti operacije XOR

Formalna definicija ovog kriptosustava glasi:

Definicija 2.2.1. ***(Jednokratna bilježnica)*** *Neka je n prirodan broj i* $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_2^n$. *Za* $K = (k_1, k_2, \ldots, k_{n-1}, k_n) \in \mathcal{K}$ *definiramo*

$$e_K(x_1, x_2, \ldots, x_{n-1}, x_n) = (x_1 +_2 k_1, x_2 +_2 k_2, \ldots, x_{n-1} +_2 k_{n-1}, x_n +_2 k_n),$$
$$d_K(y_1, y_2, \ldots, y_{n-1}, y_n) = (y_1 +_2 k_1, y_2 +_2 k_2, \ldots, y_{n-1} +_2 k_{n-1}, y_n +_2 k_n).$$

Napomena: Primijetimo da su operacije XOR i $+_2$(zbrajanje modulo 2) ekvivalentne (vrijedi $0 +_2 0 = 0, 1 +_2 1 = 0, 0 +_2 1 = 1$ i $1 +_2 0 = 1$).

Primjerice, ako Alice želi poslati poruku $P = 01100010$ i pritom koristi ključ $K = 10101011$, ona računa $C = P \oplus K = 01100010 \oplus 10101011 = 11001001$ te dobiveni šifrat šalje Bobu, koji zna ključ K pa računa $C \oplus K = 11001001 \oplus 10101011 = 01100010$ i dobiva otvoreni tekst.
Zamislimo da je $\mathcal{P} = \mathcal{C} = \mathcal{K}$ i da je skup svih elementata otvorenog teksta dan tablicom

Slovo	R	A	B	V	I	S	C	O
Binarni zapis	000	001	010	100	011	101	110	111

Neka Alice želi poslati poruku `RAB` i neka je ključ $K = 110100001$.Pripadni šifrat je $C = 110101111$, tj. `CSO`. Eva presretne šifrat i pokuša pogoditi ključ, npr. $\check{K} = 010110010$. Eva računa $\check{P} = C \oplus \check{K} = 110101111 \oplus 010110010 = 100011101$, što odgovara riječi `VIS`. S obzirom na šifrat i pretpostavljeni otvoreni tekst, Eva nikako ne može zaključiti je li poruka `VIS` vjerojatnija od poruke `RAB`, ili, štoviše, od bilo koje riječi sastavljene od tri slova odabrane abecede. Dešifriranjem poruke C bilo kojim od mogućih $8^3 = 512$ ključeva dobiva se jedan od 512 mogućih otvorenih tekstova, a sam šifrat ne odaje nikakvu informaciju o tome koji od otvorenih tekstova je traženi. Dakle, jednokratna bilježnica imuna je na napade tipa *poznat samo šifrat* zato što šifrat ne daje nikakve informacije o otvorenom tekstu osim njegove duljine.

Mana ovog kriptosustava je što se sigurnost kriptosustava smanjuje što više koristimo jedan te isti ključ. Primjerice, neka su otvoreni tekstovi P_1 i P_2 šifrirani istim ključem K te su tako dobiveni šifrati C_1 i C_2. Vrijedi $C_1 \oplus C_2 = (P_1 \oplus K) \oplus (P_2 \oplus K) = P_1 \oplus P_2$. Dakle, ako imamo dva šifrata, možemo izračunati XOR pripadnih otvorenih tekstova. U ovakvom slučaju Eva je u poziciji da, uz malo truda, razdvoji otvorene tekstove. U praksi, Eva može koristiti jednu od poruka kao kontrolu pri dešifriranju druge poruke (ili ključa). Dakle, šifrat u ovakvoj situaciji odaje informacije o početnim otvorenim tekstovima i savršena sigurnost više nije zajamčena. Situacija se pogoršava što više puta koristimo jedan te isti ključ. Zato se ovaj kriptosustav naziva *jednokratna* bilježnica. Jedan od načina za eliminaciju ovog problema je korištenje tzv. **inicijalizirajućeg vektora** (IV), slučajno odabranog ulaznog podatka unaprijed zadane duljine. Još jedna mana ovog kriptosustava je činjenica

da je za njegovu upotrebu potrebno nasumično generirati ključeve jednake duljine kao otvoreni tekst te ih sigurnim kanalom razmijeniti između pošiljatelja i primalaca.

Ali, ako Alice i Bob mogu sigurno razmijeniti ključeve, zašto ne bi odmah tim istim putem razmijenili poruku?[2]. Upravo iz ovog razloga ovaj kriptosustav nije često korišten u realnom svijetu [3].

Jednokratna bilježnica kod koje se uvijek koristi jedan te isti ključ naziva se **Vernamova** [4] **šifra**.

2.3 Protočne šifre

Vidjeli smo da pri korištenju jednokratne bilježnice problem predstavlja činjenica da je ključ jednake duljine kao i poruka koju želimo poslati. Zanima nas kako bismo mogli riješiti ovaj problem.

Protočne šifre su simetrični kriptosustavi koje možemo shvatiti kao generalizaciju jednokratne bilježnice. Za razliku od jednokratne bilježnice, koja koristi ključ jednake duljine kao sam otvoreni tekst, protočna šifra relativno kratki ključ "rastegne" u niz ključeva koji se potom koristi baš kao i ključ jednokratne bilježnice. Protočne šifre imaju manje potencijalnih kandidata za ključ nego mogućih nizova ključeva. Zato protočne šifre nisu savršeno sigurne. Opišimo rad tipične protočne šifre.

Alice prvo pomoću ključa κ algoritmom protočne šifre generira niz ključeva K_i. Zatim šifrira otvoreni tekst P_i pomoću operacije XOR: $C_i = P_i \oplus K_i$. Nakon što Bob primi šifrat, uzima isti ključ κ te algoritmom protočne šifre generira isti niz ključeva K_i. Potom Bob dešifrira poruku i dobiva polazni otvoreni tekst primjenjujući na šifratu i nizu ključeva operaciju XOR: $C_i \oplus K_i = (P_i \oplus K_i) \oplus K_i = P_i$. Primijetimo da uvijek vrijedi $K_i \oplus K_i = 0$.

[2]Primjerice, sovjetski špijuni su, u vrijeme Hladnog rata, pri ulasku u SAD imali bilježnice koje su na svakoj stranici imale zapisan jedan ključ (odakle naziv jednokratna *bilježnica*). Neki špijuni su imali ključeve zapisane na malenim papirićima koje su sakrili u orahovu ljusku.

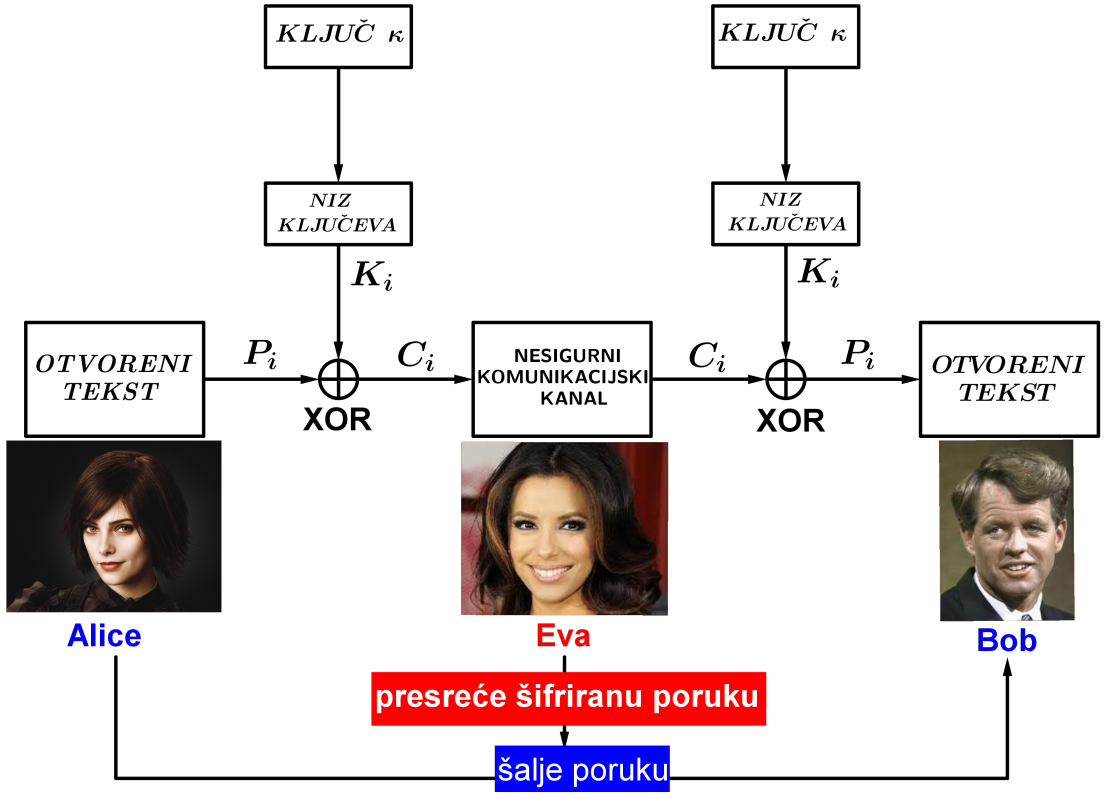
[3]Ovaj kriptosustav se, navodno, koristio pri komunikaciji između Moskve i Washingtona nakon Kubanske krize. Pri međusobnoj razmjeni ključeva posredovali su visokopouzdani kuriri.

[4]Gilbert Sandford Vernam (1890.– 1960.) diplomirani je inženjer Politehničkog instituta Worcester. 1917. godine osmislio je polialfabetsku protočnu šifru poznatu pod nazivom jednokratna bilježnica. Kasnije je osmislio automatiziranu verziju jednokratne bilježnice koja je realizirana pomoću teleprintera (elektromehaničkog pisaćg stroja) u koji se umetnula papirnata traka na kojoj je bio ispisan prethodno odabrani ključ. Teleprinter je zatim, element po element, kombinirao otvoreni tekst i ključ te generirao šifrat. Da bi dešifrirao dobiveni šifrat, primalac je trebao ponoviti jednak postupak kao i pošiljatelj poruke koristeći teleprinter i jednak ključ kao i pošiljatelj. Takvim postupkom dobiven je poslani otvoreni tekst.

P_i	K_i	$C_i = P_i \oplus K_i$	$P_i = C_i \oplus K_i$	$K_i \oplus K_i$	$P_i \oplus C_i$
0	0	0	0	0	0
0	1	1	0	0	1
1	0	1	1	0	0
1	1	0	1	0	1

Tablica 2.2: Tablica istinitosti - protočna šifra

Shema rada protočne šifre nalazi se na slici 2.2.



Slika 2.2: Shema rada protočne šifre

Alice i Bob koriste (protočnu) šifru jer očekuju da će Eva pokušati presresti poruke koje izmjenjuju. Možemo pretpostaviti da Eva zna (ili, u najgorem slučaju, može pretpostaviti) barem neki dio otvorenog teksta.

Ako Eva zna otvoreni tekst i pripadni šifrat, odmah može otkriti dio niza ključeva (šesti stupac tablice 2.2). Ako Eva može, na temelju male količine poznatih podataka, otkriti velik dio niza ključeva, onda je protočna šifra nesigurna. Dakle, sigurnost protočne šifre je u svojstvu generiranog niza ključeva.
Zanima nas koja svojstva treba imati niz ključeva da bi protočna šifra bila sigurna.

Da bi protočna šifra bila sigurna, generirani niz ključeva treba biti nepredvidiv, tj. slučajan. Među matematičarima su se kroz povijest vodile mnoge rasprave o tome što je slučajnost. Danas prevladavaju nekoliko pristupa slučajnosti, a protočne šifre generiraju slučajan niz ključeva koji je u skladu s pristupom koji smatra da je slučajna distribucija svaka distribucija koju nije moguće razlikovati od uniformne distribucije. Šifre rade s prirodnim brojevima (diskretnim skupom vrijednosti) pa zato promatramo diskretnu uniformnu distribuciju.

Definicija 2.3.1. ***(Diskretna uniformna distribucija)***
Za slučajnu varijablu X kažemo da ima diskretnu uniformnu distribuciju ako je njezin skup svih mogućih vrijednosti konačan podskup skupa realnih brojeva, tj.
$\mathcal{R}(x) = \{x_1, x_2, \ldots, x_{n-1}, x_n\} \subseteq \mathbb{R}$*, a pripadni je niz vjerojatnosti definiran kao*

$$p_i = P\{X = x_i\} = \frac{1}{n}, \text{ za sve } i = 1, 2, \ldots, n-1, n.$$

Takve distribucije koriste se ako slučajna varijabla X može poprimiti samo vrijednosti iz skupa $\mathcal{R}(x) = \{x_1, x_2, \ldots, x_{n-1}, x_n\} \subseteq \mathbb{R}$ i to tako da je vjerojatnost realizacije svakog pojedinog ishoda jednaka.

Kada govorimo o ovakvoj ideji slučajnosti, često se koristi pojam tzv. *pseudoslučajnosti*. Tako govorimo o **pseudoslučajno generiranom nizu ključeva** protočne šifre. Zato što je niz ključeva pseudoslučajan, a ne slučajan, protočne šifre, za razliku od jednokratnih bilježnica nisu savršeno sigurne. Vidimo da je u protočnim šiframa dsavršena sigurnost žrtvovana zbog praktičnosti i brzine. Dapače, može se dogoditi da je neka protočna šifra potpuno nesigurna.

Odabir ključa

Pseudoslučajno generirani niz ključeva protočne šifre potpuno je određen odabirom tajnog ključa. Zato je potrebno izbjegavati tzv. **slabe ključeve**. Niz ključeva dobiven upotrebom ovakvih ključeva odaje informacije o samome ključu. Analizom niza ključeva Eva je tada u poziciji otkriti važne informacije o ključu. Slabi ključevi zadovoljavaju neki od sljedećih uvjeta:

1. Poželjno je da promjena jednog bita ključa mijenja svaki od bitova niza ključa s vjerojatnošću $\frac{1}{2}$. U suprotnom, tajni ključ je moguće naći u vremenu manjem od vremena potrebnog za napad grubom silom (sustavnom provjerom skupa svih mogućih ključeva.

2. Poznati odnos između ključeva ne bi trebao rezultirati poznatim odnosom između generiranih nizova ključeva. Ovakva situacija može dovesti do napada tipa *poznati odnos među ključevima*.

3. Neispravno korišteni inicijalizirajući vektori mogu uzrokovati pojavljivanje tajnih informacija o ključu u generiranom nizu ključeva.

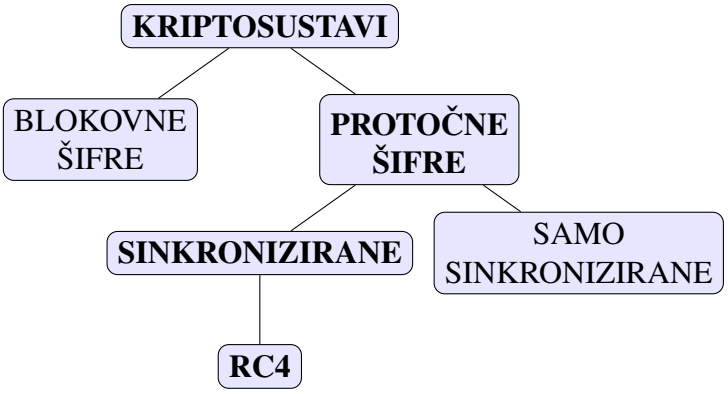
Vrste protočnih šifri

Protočne šifre generiraju pseudoslučajni niz ključeva koristeći trenutno tzv. **interno stanje** (eng. *internal state*) protočne šifre. Interno stanje je tajno i njegov sadržaj se razlikuje od šifre do šifre.
S obzirom na to mijenja li se interno stanje šifre u ovisnosti o otvorenom tekstu ili šifratu, protočne šifre dijelimo na:

1. U **sinkroniziranim protočnim šiframa** pseudoslučajni niz ključeva generiran je neovisno o otvorenom tekstu i šifratu. Da bi šifriranje proteklo uspješno, pošiljatelj i primatelj moraju biti usklađeni (sinkronizirani). Ako se dogodi da se iz poruke ispuste (ili u nju dodaju) elementi, šifriranje se ne može uspješno provesti. Da bi pošiljatelj i primatelj mogli provesti postupak razmjene poruke, potrebno je sustavno isprobati velik broj mogućih zamjenskih elemenata. Druga opcija je da, pri šifriranju, na jednako udaljenim kontrolnim mjestima šifrata uvodimo oznake da bismo lakše kontrolirali postupak šifriranja (u slučaju da se dogodi ispuštanje elemenata).

2. Protočne šifre koje za generiranje pseudoslučajnog niza ključeva koriste prethodnih N elemenata šifrata nazivaju se **samo - sinkronizirane protočne šifre** ili šifrati s autoključem (eng. *ciphertext autokey*). Ideja samo - sinkronizirane protočne šifre patentirana je 1946. godine.
   Prednost ovakvih šifri je u tome što je, u slučaju mijenjanja šifrata, primatelju poruke dovoljno samo N elemenata šifrata da bi se sinkronizirao s generatorom niza ključeva. Ako je samo jedan element šifrata izmjenjen, on može utjecati na najviše N elemenata otvorenog teksta.

Protočna šifra RC4 je primjer sinkronizirane protočne šifre.



Protočne šifre mogu biti namijenjene softverskoj ili hardverskoj primjeni, a RC4 je primjer softverske protočne sifre.

Poglavlje 3

RC4

Protočne šifre namijenjene softverskoj upotrebi obično se sastoje od dva dijela.
Prvi dio za ulaz ima tajni ključ (a neke šifre i dodatni IV) koji se zatim razvije u interno stanje. Ovaj postupak naziva se **algoritam za pripremu ključa** (eng. *key scheduling algorithm* - **KSA**). Drugi dio generira pseudoslučajni niz ključeva koristeći interno stanjeiz prvog dijela. S obzirom na dizajn pojedine šifre, izlaz ovog dijela algoritma je niz ključeva duljine bit, bajt ili riječ.
Protočnu šifru RC4 osmislio je Ronald Rivest [1] 1987. godine.

Slika 3.1: Ronald L. Rivest

[1]Ronald Linn Rivest (1947.–) američki je kriptograf i sveučilišni profesor na MIT–ju. Jedan je od tvoraca poznatog kriptosustava RSA, javnosti objavljenog 1977. godine. Osnivač je tvrtke RSA Security koja se bavi računalnom i mrežnom sigurnosti. Tvorac je simetričnih kriptosustava RC2, RC4, RC5, i jedan od tvoraca RC6 (RC3 je razbijen tijekom razvoja u RSA Security, a RC1 iz sličnih razloga nikada nije objavljen).

RC u RC4 znači *Rivest Cipher* (Rivestova šifra) ili, kako mnogi govore, *Ron's Code* (Ronov kod).

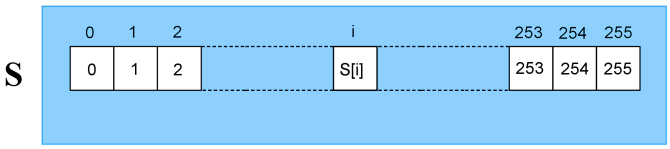
3.1 Algoritam RC4

Algoritam RC4 bio je tajna sve do 1994. godine, kada je objavljen na internetu. RSA Security nikad službeno nije potvrdila algoritam RC4. Trenutno je RC4 jedna od ponajboljih protočnih šifri s mnogim primjenama u realnom svijetu. Kao i kod ostalih protočnih šifri, algoritam RC4 sastoji se od dva dijela: algoritma za pripremu ključa (**KSA**) i algoritma koji generira pseudoslučajni niz ključeva (eng. *Pseudo - Random Generation Alghorithm*-**PRGA**).
RC4 se sastoji od permutacije S, tzv. S-kutije

$$S = (\begin{smallmatrix} 0 & 1 & 2 & 3 & \dots & N-2 & N-1 \\ S[0] & S[1] & S[2] & S[3] & \dots & S[N-2] & S[N-1] \end{smallmatrix}),$$

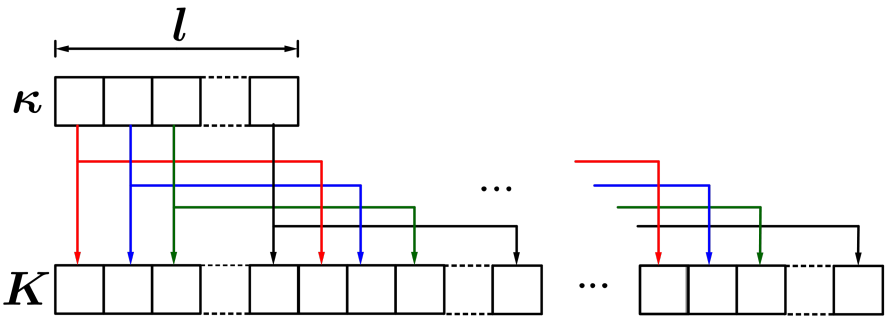
gdje je $N = 2^n$, a svaki od $S[i]$ n-bitni prirodni broj. Obično je $n = 8$ i $N = 2^8 = 256$.
Prvo S inicijaliziramo na identitetu (slika 3.1), tj. vrijedi

$$S[i] = i \text{ , za sve } i \in \{1, 2, \dots, n-1\}$$



Slika 3.2: RC4 inicijalizacija

Zatim se odabire tajni ključ κ, duljine l bajtova (obično je $5 \leq l \leq 16$, što odgovara duljini $40 \leq l_{bit} \leq 128$ u bitovima). Ključem κ punimo niz $K = K[0], K[1], K[2], K[3], \dots, K[N-2], K[N-1])$, gdje je $K[i], i \in \{1, 2, \dots, n-1\}$ n-bitni prirodni broj. Punjenje se odvija tako da prvih l bajtova niza K napunimo s l bajtova ključa κ, zatim idućih l bajtova niza K napunimo s l bajtova ključa κ te postupak analogno nastavimo sve dok ne popunimo čitav K (slika 3.1).

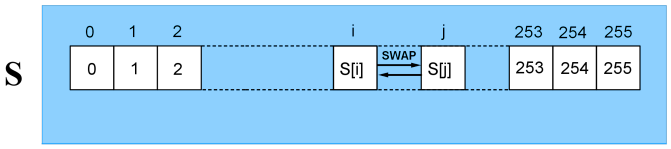


Slika 3.3: Generiranje niza K

Primjerice, ako je duljina ključa 5 bajtova (40 bitova), onda cijeli ključ u potpunosti popuni $K[0], K[1], K[2], K[3]$ i $K[4]$ te se postupak nastavlja dok se ne popuni čitav K.

RC4 KSA

Prvo se izvodi KSA algoritam, koji koristi dva indeksa, i i j, koji su na početku inicijalizirani na nulu. Potom, u svakom od n koraka, j je formiran kao zbroj i-tog elementa niza K, i-tog elementa permutacije S i prethodne vrijednosti j. Zatim vrijednosti $S[i]$ i $S[j]$ permutacije S zamijene mjesta (eng. *swap*) (slika 3.1).
Ovime završava KSA algoritam.
Dakle, KSA ima zadatak promiješati (eng. *scramble*) elemente polazne permutacije.

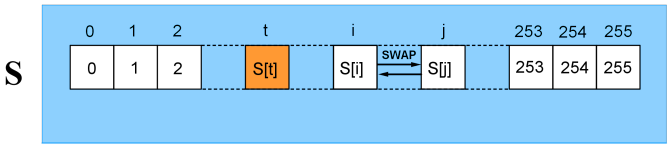


Slika 3.4: RC4 KSA SWAP

RC4 PRGA

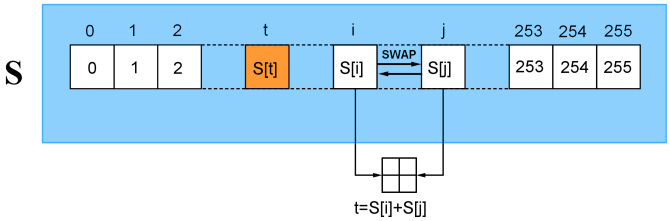
Nakon što je KSA algoritam promiješao elemente od S, slijedi algoritam koji generira pseudoslučajni niz ključeva.
Indeksi i i j su ponovno inicijalizirani na nulu. U svakom koraku algoritma indeks i se povećava za 1 , a indeks j se dobiva kao suma i-te vrijednosti permutacije S i vrijednosti j

iz prethodnog koraka.
Zatim vrijednosti permutacije S na i-tom i j-tom mjestu zamijene mjesta (slika 3.1).



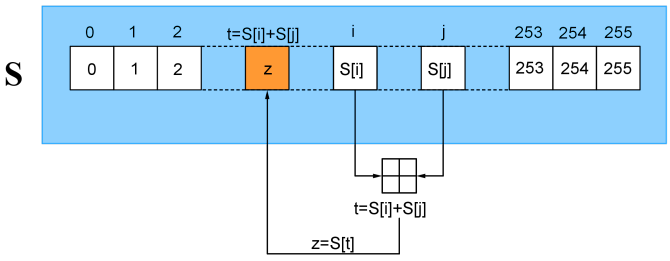
Slika 3.5: RC4 PRGA SWAP

Potom se određuje vrijednost t kao zbroj $S[i]$ i $S[j]$ (slika 3.1).



Slika 3.6: RC4 PRGA, $t = S[i] + S[j]$

Izlazna vrijednost ovog algoritma je vrijednost permutacije S na poziciji t, u oznaci z (slika 3.1).



Slika 3.7: RC4 PRGA, $z = S[t]$

Nakon što odredimo vrijednost z, primijenimo operaciju XOR na z i n bitova otvorenog teksta te tako dobivamo dio šifrata duljine n bitova. Algoritam provodimo tako dugo dok ne šifriramo čitavu poruku. Pri dešifriranju poruke na z i dio šifrata primijenimo operaciju XOR te tako dobivamo dio otvorenog teksta (postupak ponavljamo tako dugo dok ne

dešifriramo cijelu poruku).

Napomena: Sve operacije zbrajanja u algoritmima KSA i PRGA odvijaju se modulo N.

Algoritam RC4 KSA

1: INICIJALIZACIJA ▷ S je identiteta
2: **for** $i = 0$ **to** $N - 1$ **do**
3: $S[i] \leftarrow i$
4: SCRAMBLING ▷ Mijenjanje permutacije S
5: $j \leftarrow 0$
6: **for** $i = 0$ **to** $N - 1$ **do**
7: $j \leftarrow j + S[i] + K[i \bmod l] \pmod{N}$
8: SWAP $(S[i], S[j])$

Algoritam RC4 PRGA

1: INICIJALIZACIJA
2: $i \leftarrow 0$
3: $j \leftarrow 0$
4: PETLJA ZA GENERIRANJE NIZA KLJUČEVA
5: *Provodi petlju sve dok cijela poruka nije šifrirana /dešifrirana:*
6: $i \leftarrow i + 1 \pmod{N}$
7: $j \leftarrow j + S[i] \pmod{N}$
8: SWAP $(S[i], S[j])$
9: $t \leftarrow S[i] + S[j]$
10: $z \leftarrow S[t]$
11: **return** *XOR z-a i sljedećeg dijela ulaznog podatka*

Primjer 3.1.1. *Neka je zadan ključ* $\kappa = 00001111\ 01011010\ 11011001$. *Pretpostavimo da na raspolaganju imamo slova engleskog alfabeta te da želimo šifrirati poruku RONOVA SIFRA pomoću RC4. Prvo otvoreni tekst zapišemo u binarnom obliku (koristeći ASCII kod). Tada poruka* $(R, O, N, O, V, A, S, I, F, R, A)$ *poprima oblik*

$$(01010010,\ 01001111,\ 01001110,\ 01010011,\ 01010110,$$
$$01000001, 01010011, 01001001, 01000110, 01010010, 01000001).$$

Zatim se pokretanjem algoritma RC4 dobiva niz ključeva

$$(11101111,\ 01001010,\ 00110100,\ 11011111,\ 10011111,$$
$$11100101, 10111111, 01010011, 10100000, 10100010, 10000000).$$

Primjenom operacije XOR na elemente otvorenog teksta i niza ključeva dobiva se šifrat

$$(10111101,\ 00000101,\ 01111010,\ 10001100,\ 11001001,$$
$$10100100, 11101100, 00011010, 11100110, 11110000, 11000001).$$

Kada primaoc dobije poruku, pomoću ključa κ *generira isti niz ključeva te na elemente šifrata i niza ključeva primijeni operaciju XOR i tako dobiva polazni otvoreni tekst.*

otvoreni tekst	R	O	N	O	V	A	S	I	F	R	A
ASCII (binarno)	01010010	01001111	01001110	01010011	01010110	01000001	01010011	01001001	01000110	01010010	01000001
					$\oplus$						
niz ključeva	11101111	01001010	00110100	11011111	10011111	11100101	10111111	01010011	10100000	10100010	10000000
					=						
šifrat	10111101	00000101	01111010	10001100	11001001	10100100	11101100	00011010	11100110	11110000	11000001
					$\oplus$						
niz ključeva	11101111	01001010	00110100	11011111	10011111	11100101	10111111	01010011	10100000	10100010	10000000
					=						
ASCII (binarno)	01010010	01001111	01001110	01010011	01010110	01000001	01010011	01001001	01000110	01010010	01000001
otvoreni tekst	R	O	N	O	V	A	S	I	F	R	A

Tablica 3.1: RC4-primjer šifriranja i dešifriranja poruke „RONOVA SIFRA”

Notacija

U idućim poglavljima koristit će se sljedeća notacija:

- S, i i j u KSA
- S_r, i_r i j_r su vrijednosti u r-toj rundi KSA
- S_r^G, i_r^G i j_r^G su vrijednosti u r-toj rundi PRGA-e.

Poglavlje 4

KSA

Algoritam RC4 je jako jednostavan. No, zanima nas je li ovako osmišljena šifra kriptografski sigurna, tj. koliko su nasumično raspoređeni elementi permutacije S. S kriptografskog stajališta, bilo koji događaj u nizu ključeva za kojeg možemo računski pokazati da nije slučajan smatra se nepoželjnim. Zato je potrebno izvršiti analizu RC4 da bi se utvrdilo postoje li takve manjkavosti u dizajnu ove šifre. U sljedeća dva poglavlja bavit ćemo se analizom KSA i PRGA-e.

4.1 O slučajno generiranim permutacijama

Cilj KSA je elemente identitete S promiješati tako da S ima nasumično poredane elemente. Postavlja se pitanje kako generirati permutaciju sa slučajno poredanim elementima.
Neka je $T = (\tau_0, \tau_1, \ldots, \tau_n)$ permutacija n elemenata. Tada je primjenom $n-1$ transpozicija moguće generirati slučajnu permutaciju.
Neka je $n \longmapsto rand(0, n)$ funkcija koja svakom prirodnom broju pridružuje slučajno odabrani prirodni broj između 0 i n, pri čemu su vjerojatnosti odabira svakog od broja međusobno jednake.
Slučajno generiranu permutaciju dobivamo tako da svaki element permutacije, počevši od posljednjeg zamijenimo s nasumično odabranim elementom te permutacije.

Algoritam za generiranje nasumične permutacije (SP)

1: **for** $i = n - 1$ **to** 1 **do**
2: $\quad \pi_i \longleftrightarrow \pi_{rand(0,i)}$

Dokazano je (u [5]) da je ovako dobivena permutacija doista slučajna.
Razlike između algoritma SP i KSA dane su tablicom 4.1.

RC4 KSA vs SP	
Indeks i povećava se od 0 do $N-1$ (imamo N transpozicija)	Indeks i smanjuje se od $N-1$ do 0 (imamo N-1 transpoziciju)
Vrijednost j mijenja se u ovisnosti o elementima niza K (koji ovisi o tajnom ključu).	Indeks j može poprimiti bilo koju vrijednost između 0 i $N-1$, tj. u i-tom koraku j može poprimiti samo vrijednosti od 0 do i.

Tablica 4.1: Razlike između algoritma za generiranje slučajne permutacije i KSA

Prirodno se nameće sljedeća ideja: bismo li mogli koristiti SP umjesto KSA koristeći pritom elemente od K umjesto $rand(0, n)$? Uspostavlja se da ne, zato što bi u tom slučaju permutacija nakon završenog KSA u potpunosti otkrila elemente od K (a samim time i tajni ključ). Primjerice, τ_{n-1} će sadržavati prvi element niza ključeva kojem je algoritam pristupio, τ_{n-2} drugi element ... Zaključujemo da je potrebno osmisliti KSA sa dobrim kriptografskim svojstvima.

Predznak permutacije S

Definicija 4.1.1. *Predznak permutacije $T = (\tau_0, \tau_1, \ldots, \tau_n)$ definira se kao $sgn(T) = (-1)^{I_T}$, gdje je $I_T = \{(u, v) : u < v \text{ i } \tau_u > \tau_v\}$ skup **inverzija**.*

Dokazano je da vrijedi sljedeći rezultat:

Teorem 4.1.2. *Neka su, u svakoj od rundi KSA, vrijednosti indeksa j nezavisno i slučajno odabrani, uniformno distribuirani elementi od $\mathbb{Z}_N$. Tada predznak permutacije S nakon završetka KSA može poprimiti vrijednosti 1 i -1 i to sa ovim vjerojatnostima:*

$$P(sgn(S_N) = (-1)^N) \approx \frac{1}{2}(1 + e^{-2}) \text{ i } P(sgn(S_N) = (-1)^{N-1}) \approx \frac{1}{2}(1 - e^{-2}).$$

Iz prethodnog deorema zaključujemo da se predznak permutacije nakon završetka KSA može predvidjeti s vjerojatnoču koja se od slučajne razlikuje za otprilike 6.7%.

Analizom KSA uspostavlja se da vrijednosti permutacije nakon KSA nisu slučajne. Iako vrijednosti permutacije ne ovise o tajnom ključu, možemo ih razlikovati od nasumično odabrane permutacije bez ikakve pretpostavke na tajni ključ.

4.2 Analiza KSA

Prve rezultate o analizi KSA objavio je Andrew Roos.

Teorem 4.2.1. *Najvjerojatnija vrijednost y-tog elementa (za male vrijednosti y) permutacije S nakon završetka KSA je* $S_N[y] = f_y$*, gdje je* $f_y = \frac{y\cdot(y+1)}{2} + \sum_{x=0}^{y} K[x]$.

Dakle, za dovoljno male vrijednosti y elemente permutacije možemo prikazati u obliku linearne kombinacije koja ovisi o ključu. Roos je pokazao da je, za dovoljno male y ($y < l$), velika vjerojatnost da niti jedan od elemenata $S[i]$ i $S[j]$ nije sudjelovao u niti jednoj od prijašnjih zamjeni. Zato možemo pretpostaviti da je $S_y[y] = y$ prije zamjene u $(y+1)$-oj rundi. Vjerojatnost da odabrani element od S ne sudjeluje u jednoj zamjeni je $\frac{N-1}{N}$, a vjerojatnost da taj element ne sudjeluje niti u jednoj od N zamjena je $(\frac{N-1}{N})^N$. Prema tome, vrijedi sljedeći rezultat:

Teorem 4.2.2. *Najvjerojatnije vrijednosti prvog i drugog elementa permutacije S, nakon završtka KSA su*

1. $P(S_N[1] = K[0] + K[1] + 1) \approx \left(\frac{N-1}{N}\right)^N i$
2. $P(S_N[2] = K[0] + K[1] + K[2] + 3) \approx \left(\frac{N-1}{N}\right)^N$.

Pokazuje se da su vjerojatnosti da element permutacije poprimi određenu vrijednost u vezi s elementima od K. Dapače, može se pokazati kako se te vjerojatnosti mijenjaju kao funkcija od y. Da bismo to vidjeli, koristimo nekoliko pomoćnih činjenica.

Teorem 4.2.3. ***(Teorem potpune vjerojatnosti)*** *Neka je* $(\Omega, \mathcal{F}, P)$ *vjerojatnosni prostor i* $\{H_i : i \in I\}$, $I \subseteq \mathbb{N}$ *potpun sustav događaja na njemu. Tada za proizvoljan dogaąj* $A \in \mathcal{F}$ *vrijedi*

$$P(A) = \sum_{i\in I} P(A|H_i) \cdot P(H_i)$$

.

Lema 4.2.4. *Neka su, u svakoj od rundi KSA, vrijednosti indeksa j nezavisno i slučajno odabrani, uniformno distribuirani elementi od* $\mathbb{Z}_N$*. Tada, za sve* $0 \leq y \leq N-1$ *vrijedi*

$$P(j_{y+1} = \sum_{x=0}^{y} S_0[x] + \sum_{x=0}^{y} K[x]) \approx \left(\frac{N-1}{N}\right)^{1+\frac{y\cdot(y+1)}{2}}.$$

Dokaz. Za $y \geq 0$, neka E_y označava događaj

$$j_{y+1} = \sum_{x=0}^{y} S_0[x] + \sum_{x=0}^{y} K[x].$$

Neka A_y označava događaj

$S_0[x] = S_x[x]$, za sve $x \in [0, y]$, a $\overline{A_y}$ njemu komplementaran događaj.

Po teoremu 4.2.3 vrijedi

$$P(E_y) = P(E_y|A_y) \cdot P(A_y) + P(E_y|\overline{A_y}) \cdot P(\overline{A_y}).$$

Također, vrijedi $P(E_y|A_y) = 1$ i $P(E_y|\overline{A_y}) \approx \frac{1}{N}$.

Sada ćemo metodom matematičke indukcije pokazati da vrijedi $P(A_y) = \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}$.
Baza indukcije Trivijalno vrijedi $P(A_0) = P(S_0[0] = S_0[0]) = 1$.

Pretpostavka indukcije (p.i.) Pretpostavimo da za neki $n \in \mathbb{N}$ vrijedi $P(A_y) = \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}$.
Korak indukcije
Tada je

$$\begin{aligned} \mathrm{P}(\mathrm{A}_{y+1}) &= P(A_y \wedge S_{y+1}[y+1] = S_0[y+1]) \\ &\approx P(A_y) \cdot P(S_{y+1}[y+1] = S_0[y+1]) \\ &\overset{P.I.}{=} \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}} \cdot P(S_{y+1}[y+1] = S_0[y+1]). \end{aligned} \tag{4.1}$$

Događaj $S_{y+1}[y+1] = S_0[y+1]$ će nastupiti ako je svaki od indeksa $j_1, j_2, \ldots j_{y+1}$ različiti od $y+1$, a čija je vjerojatnost $\left(\frac{N-1}{N}\right)^{y+1}$. Uvrštavanjem u 4.1 dobiva se $P(A_{y+1}) = \left(\frac{N-1}{N}\right)^{\frac{(y+1)\cdot(y+2)}{2}}$ pa tvrdnja vrijedi po aksiomu matematičke indukcije.

Uvrštavanjem u 4.2 dobiva se $P(E_y) \approx \left(\frac{N-1}{N}\right)^{1+\frac{y\cdot(y+1)}{2}}$, što je i trebalo dokazati. □

Lema 4.2.5. *Neka su, u svakoj od rundi KSA, vrijednosti indeksa j nezavisno i slučajno odabrani, uniformno distribuirani elementi od $\mathbb{Z}_N$. Tada, za sve $0 \leq y \leq r-1$, $1 \leq r \leq N$ vrijedi*

$$P(S_r[y] = S_0[j_{y+1}]) \approx \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{r-1}.$$

Dokaz. Vrijednosti $S_y[j_{y+1}]$ i $S_{y+1}[y]$ u $(y+1)$-oj rundi zamijene mjesta. Indeks j_{y+1} nije sudjelovao niti u jednoj od prethodnih y zamjena ako je $j_{y+1} \neq j_i$, $i \in \{1, 2, \ldots y\}$ i ako je $j_{y+1} \neq t$ $t \in \{0, 1, \ldots y-1\}$. Vjerojatnost prvog događaja je $\left(\frac{N-y}{N}\right)$, a vjerojatnost drugog događaja je $\left(\frac{N-1}{N}\right)^y$. Dakle,

$$P(S_{y+1}[y] = S_0[j_{y+1}]) \approx \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^y.$$

Nakon $(y+1)$-e runde, vjerojatnost da bilo koji od preostalih $r-1-y$ j indeksa poprimi vrijednost y je $\left(\frac{N-1}{N}\right)^{r-1-y}$. Zato je

$$\mathrm{P}(\mathrm{S}_r[y] = S_0[j_{y+1}]) \approx \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^y \cdot \left(\frac{N-1}{N}\right)^{r-1-y} = \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{r-1}. \tag{4.2}$$

□

Teorem 4.2.6. *Neka su, u svakoj od rundi KSA, vrijednosti indeksa j nezavisno i slučajno odabrani, uniformno distribuirani elementi od* $\mathbb{Z}_N$*. Tada, za sve* $0 \leq y \leq r-1$, $1 \leq r \leq N$ *vrijedi*

$$P(S_r[y] = f_y) = \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{r+\frac{y\cdot(y+1)}{2}} + \frac{1}{N},$$

gdje je $f_y = S_0[\sum_{x=0}^{y} S_0[x] + \sum_{x=0}^{y} K[x]]$.

Dokaz. Neka je A_y događaj iz leme 4.2.4. Događaj $S_r[y] = f_y$ može nastupiti u jednom od sljedeća dva međusobno disjunktna slučaja:

1. slučaj

Dogodili su se i A_y i $S_r[y] = S_0[j_{y+1}]$. Prema lemama 4.2.4 i 4.2.5 dobiva se

$$P(A_y)\cdot P(S_r[y] = S_0[j_{y+1}]) = \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}} \cdot \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{r-1} = \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}+r-1}.$$

2. slučaj

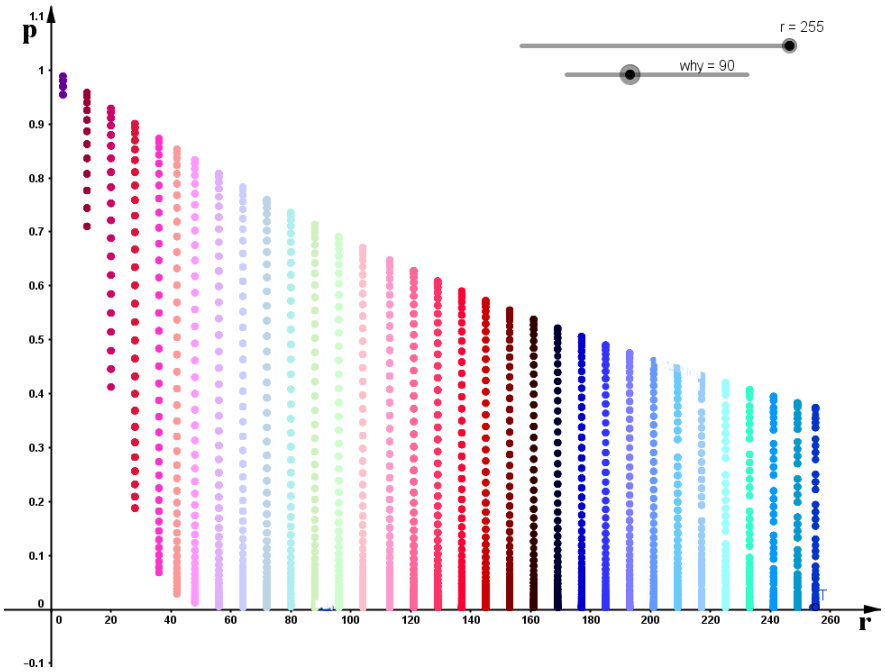
Niti jedan od događaja iz prvog slučaja se nije dogodio (vjerojatnost ovog događaja je $\frac{1}{N}$), ali vrijedi $S_r[y] = S_0[\sum_{x=0}^{y} S_0[x] + \sum_{x=0}^{y} K[x]]$, čija je vjerojatnost $(1 - \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}+r-1})$.

Zbrajanjem vjerojatnosti svakog od slučajeva dobiva se

$$\begin{aligned} P(S_r[y] = f_y) &= \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}+r-1} + (1 - \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}+r-1}) \cdot \frac{1}{N} \\ &= (1 - \frac{1}{N}) \cdot \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}+r-1} + \frac{1}{N} \\ &= \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{r+\frac{y\cdot(y+1)}{2}} + \frac{1}{N}. \end{aligned}$$

□

Vidimo da se s povećanjem broja runde r, vjerojatnosti $P(S_r[y] = f_y)$ smanjuju (slika 4.2).



Slika 4.1: Teorem 4.2.6

Uvrštavanjem $r = N$ u 4.2.6 dobiva se ovaj rezultat:

Korolar 4.2.7. *Nakon posljednje runde KSA vrijedi*

$$P(S_N[y] = f_y) = \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{N+\frac{y\cdot(y+1)}{2}} + \frac{1}{N}, \text{ za sve } 0 \leq y \leq N-1.$$

Dokazano je da prethodno prezentirani rezultati vrijede za bilo koju početnu permutaciju S (S ne mora biti identiteta).
Za $y \geq 48$, i teoretski i eksperimentalno dobivene vrijednosti teže k $\frac{1}{N}$ (tj. k 0.0039 što je $\approx 0.00390625 = \frac{1}{256}$, za $N = 256$ (prema [5]).

Nakon što smo utvrdili u kakvoj su vezi odabir elementa permutacije S s elementima od K, postavlja se pitanje postoji li veza između elemenata od K i elemenata od S kojima se iterirano pristupa, tj. elementima $S_r[S_r[y]]$,$S_r[S_r[S_r[y]]]\ldots$
Sljedeći rezultati govore o tome kakva je situacija nakon druge runde KSA.

Lema 4.2.8. $P(S_2[S_2[1]] = K[0] + K[1] + 1) = \frac{3}{N} - \frac{4}{N^2} + \frac{2}{N^3}$.
Nadalje, $P(S_2[S_2[1]] = K[0] + K[1] + 1 \wedge S_2[1] \leq 1) \approx \frac{2}{N}$.

Dokaz. Razlikujemo tri slučaja:
1. slučaj Neka je $K[0] \neq 0$ i $K[1] = N - 1$.

Vjerojatnost prvog događaja je $\frac{N-1}{N}$, a drugog $\frac{1}{N}$.
Nadalje,

$$S_2[1] = S_1[K[0]+K[1]+1] = S_1[K[0]+N-1+1] = S_1[K[0]+N] = S_1[K[0]] = S_0[0] = 0.$$

(vrijedi $S_2[1] \leq 1$) pa je

$$S_2[S_2[1]] = S_2[0] = S_1[0] = K[0] = K[0] + N = K[0] + N - 1 + 1 = K[0] + K[1] + 1.$$

2. slučaj Neka je $K[0] + K[1] = 0$ i $K[0] \neq 1$, tj. $K[1] \neq N - 1$
Vjerojatnost prvog događaja je $\frac{N-1}{N}$, a drugog $\frac{1}{N}$.
Nadalje,

$$S_2[1] = S_1[K[0] + K[1] + 1] = S_1[1] = S_0[1] = 1$$

(vrijedi $S_2[1] \leq 1$) pa je

$$S_2[S_2[1]] = S_2[1] = 1 = 0 + 1 = K[0] + K[1] + 1.$$

3. slučaj
Događaj $S_2[S_2[1]] = K[0] + K[1] + 1$ je slučajan (ne uzimajući u obzir prethodna dva slučaja). Dakle, njegova je vjerojatnost $(1 - 2 \cdot \frac{N-1}{N} \cdot \frac{1}{N}) \cdot \frac{1}{N}$. Tada je vjerojatnost događaja $S_2[1] \leq 1$ jednaka $\frac{2}{N}$.
Zato je

$$\begin{aligned} \mathrm{P}(\mathrm{S}_2[S_2[1]] = K[0] + K[1] + 1) &= 2 \cdot \frac{N-1}{N^2} + (1 - 2 \cdot \frac{N-1}{N} \cdot \frac{1}{N}) \cdot \frac{1}{N} \\ &= \frac{3}{N} - \frac{4}{N^2} + \frac{2}{N^3}, \end{aligned}$$

$$\begin{aligned} \mathrm{P}(\mathrm{S}_2[S_2[1]] = K[0] + K[1] + 1 \wedge S_2[1] \leq 1) &= 2 \cdot \frac{N-1}{N^2} + \frac{2}{N} \cdot (1 - 2 \cdot \frac{N-1}{N} \cdot \frac{1}{N}) \cdot \frac{1}{N} \\ &= \frac{2}{N} - 4 \cdot \frac{N-1}{N^4} \approx \frac{2}{N}. \end{aligned}$$

□

Dakle, nakon druge runde KSA, dogaaj $S_2[S_2[1]] = K[0] + K[1] + 1$ nije slučajan (njegova vjerojatnost je različita od $\frac{1}{N}$).
Promotrimo sada što se događa u preostalim rundama KSA.

Neka je

$$p_r = P((S_{r-1}[S_{r-1}[1]] = K[0] + K[1] + 1) \wedge (S_r[1] \leq r - 1)), \; za \; r \geq 2.$$

Tada vrijede sljedeći rezultati (za dokaze pogledati [5]).

Lema 4.2.9. *Za* $r \geq 3$ *vrijedi* $p_r = (\frac{N-2}{N}) \cdot p_{r-1} + \frac{1}{N} \cdot (\frac{N-2}{N}) \cdot (\frac{N-1}{N})^{2(r-2).}$

Teorem 4.2.10. $P(S_N[S_N[1]] = K[0] + K[1] + 1) \approx (\frac{N-1}{N})^{2N}$.

Željeli bismo generalizirati dobivene rezultate, tj. postavlja se pitanje je li $S_r[S_r[y]] = f_y$ slučajan događaj. U nastavku donosimo odgovor na ovo pitanje u terminima malih vrijednosti y.

Lema 4.2.11. *Za* $0 \leq y \leq 31$ *vrijedi*

$$P((S_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_{y+1}[y] \leq y)) \approx \left(\frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}\right) \cdot \left(y\left(\frac{N-2}{N}\right)^{y-1}\right) + \left(\frac{N-1}{N}\right)^{y}.$$

Dokaz. Uvjet $S_{y+1}[y] \leq y$ podrazumijeva da $S_{y+1}[y]$ može poprimiti bilo koju od $y+1$ vrijednosti iz skupa $\{0, 1, 2, \ldots y\}$.
Zato razlikujemo dva slučaja:
1. slučaj $S_{y+1}[y] < y$ i **2. slučaj** $S_{y+1}[y] = y$.

1. slučaj
Neka je $S_{y+1}[y] = x$, za neki $0 \leq x \leq y-1$. Tada vrijedi $S_{y+1}[x] = f_y$ ako su nastupili svi od sljedećih (među sobno nezavisnih) događaja:
1.1. Od prve do x-te runde, j je različit od x i f_y pa je nakon x-te runde $S_x[x] = x$ i $S_x[f_y] = f_y$. Vjerojatnost ovog događaja je $\left(\frac{N-2}{N}\right)^x$.
1.2. U $x+1$-oj rundi j_{x+1} postaje jednak f_y, a nakon zamjene elemenata permutacije imamo $S_{x+1}[x] = f_y$ i $S_{x+1}[f_y] = x$. Vjerojatnost ovog događaja je $\frac{1}{N}$.
1.3. Od $x+2$-e do y-te runde j je različit od x i f_y pa nakon y-te runde imamo $S_y[x] = f_y$ i $S_y[f_y] = x$. Vjerojatnost ovog događaja je $\left(\frac{N-2}{N}\right)^{y-x-1}$.
1.4. U $y+1$-oj rundi j_{y+1} postaje jednak f_y, a nakon zamijene elemenata permutacije je $S_{y+1}[y] = S_y[f_y] = x$, a

$$S_{y+1}[S_{y+1}[y]] = S_{y+1}[x] = S_y[x] = f_y.$$

Prema lemi 4.2.4 vjerojatnost ovog događaja je $\left(\frac{N-1}{N}\right)^{1+\frac{y\cdot(y+1)}{2}} + \frac{1}{N}$, a za male vrijednosti y $(0 \leq y \leq 31)$ to je približno jednako $\left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}$.
Kada uzmemo u obzir sve slučajeve, imamo

$$\mathrm{P}((\mathrm{S}_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_{y+1}[y] = x)) = \left(\frac{N-2}{N}\right)^{x} \cdot \frac{1}{N} \cdot \left(\frac{N-2}{N}\right)^{y-x-1} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}$$
$$= \frac{1}{N} \cdot \left(\frac{N-2}{N}\right)^{y-1} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}.$$

Zbrajanjem po svim $x \in \{0, 1, \ldots, y-1\}$ (kojih je ukupno y) dobivamo

$$P((S_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_{y+1}[y] \leq y+1)) = \frac{y}{N} \cdot \left(\frac{N-2}{N}\right)^{y-1} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}.$$

2. slučaj $S_{y+1}[y] = y$.
Događaj $S_{y+1}[y] = f_y$ je moguć ako se dogodi svaki od događaja:
2.1. $f_y = y$ Vjerojatnost ovog događaja je $\frac{1}{N}$.
2.2. U prvih y rundi vrijedi $j \neq y$. Vjerojatnost ovog događaja je $(\frac{N-1}{N})^y$.
2.3. U $y+1$-oj rundi vrijedi $j_{y+1} = f_y$ i ne dolazi do zamjene elemenata permutacije. Vjerojatnost ovog događaja je $\left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}$ (analogno kao u slučaju 1.4.).

Zbrajanjem vrijednosti dobivenih u prethodna tri podslučaja, dobiva se

$$P((S_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_{y+1}[y] = y)) = \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{y} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}.$$

Zbrajanjem vrijednosti dobivenih u 1. i 2. slučaju dobiva se

$$P((S_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_{y+1}[y] \leq y)) \approx \left(\frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{\frac{y\cdot(y+1)}{2}}\right) \cdot \left(y\left(\frac{N-2}{N}\right)^{y-1}\right) + \left(\frac{N-1}{N}\right)^{y}.$$

□

Za $0 \leq y \leq N-1$, $1 \leq r \leq N$, neka je $q_r(y) = P((S_{y+1}[S_{y+1}[y]] = f_y) \wedge (S_r[y] \leq r-1))$.

Teorem 4.2.12. *Za* $0 \leq y \leq 31$, $y+2 \leq r \leq N$, *vrijedi*

$$q_r(y) = \left(\frac{N-2}{N}\right) q_{r-1}(y) + \frac{1}{N} \cdot \left(\frac{N-y}{N}\right) \cdot \left(\frac{N-y}{N}\right)^{\frac{y\cdot(y+1)}{2}+2r-3}.$$

Napomena: Dokazano je da do sada prezentirani rezultati vrijede za bilo koju polaznu permutaciju, a ne samo identitetu.

Teorem 4.2.13. *Na kraju KSA, za* $0 \leq u \leq N-1,\ 0 \leq v \leq N-1$ *vrijedi*

$$P(S_N[u] = v) = \begin{cases} \frac{1}{N}\left(\left(\frac{N-1}{N}\right)^v + \left(1 - \left(\frac{N-1}{N}\right)^v \left(\frac{N-1}{N}\right)^{N-u-1}\right)\right), za\ v \leq u \\ \frac{1}{N}\left(\left(\frac{N-1}{N}\right)^{N-u-1} + \left(\frac{N-1}{N}\right)^v\right), za\ v > u. \end{cases}$$

Teorem 4.2.14. *Za* $v \geq 0$ *vrijedi*

$$P(S_v[v] = v) = \left(\frac{N-1}{N}\right)^v.$$

Dokaz. Od prve do v-te runde, indeks i poprima vrijednosti od 0 do $v-1$.
Nakon v-te runde će vrijediti $S_v[v] = S_0[v] = v$ ako je vrijedilo $v \neq j_i,\ i = 1, 2, \ldots v$.
Vjerojatnost ovog događaja je $\left(\frac{N-1}{N}\right)^v$.
Za $v = 0$ imamo $P(S_0[0] = 0) = 1 = \left(\frac{N-1}{N}\right)^0$ pa tvrdnja vrijedi za sve $v \geq 0$. □

Teorem 4.2.15. *Za* $v \geq u+1$ *vrijedi*

$$P(S_v[u] = v) = \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{v-u-1}.$$

Teorem 4.2.16. *Za* $v \geq u+1$*, osim za slučaj* $u = 0$ *i* $v = 1$*, vrijedi*

$$P(S_{v+1}[u] = v) = \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{v-u} + \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^v - \frac{1}{N^2} \cdot \left(\frac{N-1}{N}\right)^{2v-u-1}.$$

Neka je $p_r^{u,v} = P(S_r[u] = v)$, za $1 \leq r \leq N$. Tada vrijede iduća dva rezultata:

Teorem 4.2.17. *Za* $0 \leq u \leq N-2,\ u+1 \leq v \leq N-1$ *vrijedi*

$$P(S_N[u] = v) = p_{v+1}^{u,v} \cdot \left(\frac{N-1}{N}\right)^{N-1-v} + (1 - p_{v+1}^{u,v}) \cdot \frac{1}{N} \cdot \left(\left(\frac{N-1}{N}\right)^v - \left(\frac{N-1}{N}\right)^{N-1}\right),$$

gdje je

$$p_{v+1}^{u,v} = \begin{cases} \frac{2(N-1)}{N^2}, za\ u = 0\ i\ v = 1 \\ \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{v-u} + \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^v - \frac{1}{N^2} \cdot \left(\frac{N-1}{N}\right)^{2v-u-1}, inače. \end{cases}$$

Teorem 4.2.18. *Za* $0 \leq v \leq N-2$, $v \leq u \leq N-1$ *vrijedi*

$$P(S_N[u] = v) = \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{N-1-u} + \frac{1}{N} \cdot \left(\frac{N-1}{N}\right)^{v+1} - \frac{1}{N^2} \cdot \left(\frac{N-1}{N}\right)^{N+v-u}.$$

Dokaz. Literatura u kojoj se nalaze dokazi prethodnih teorema je [5]. □

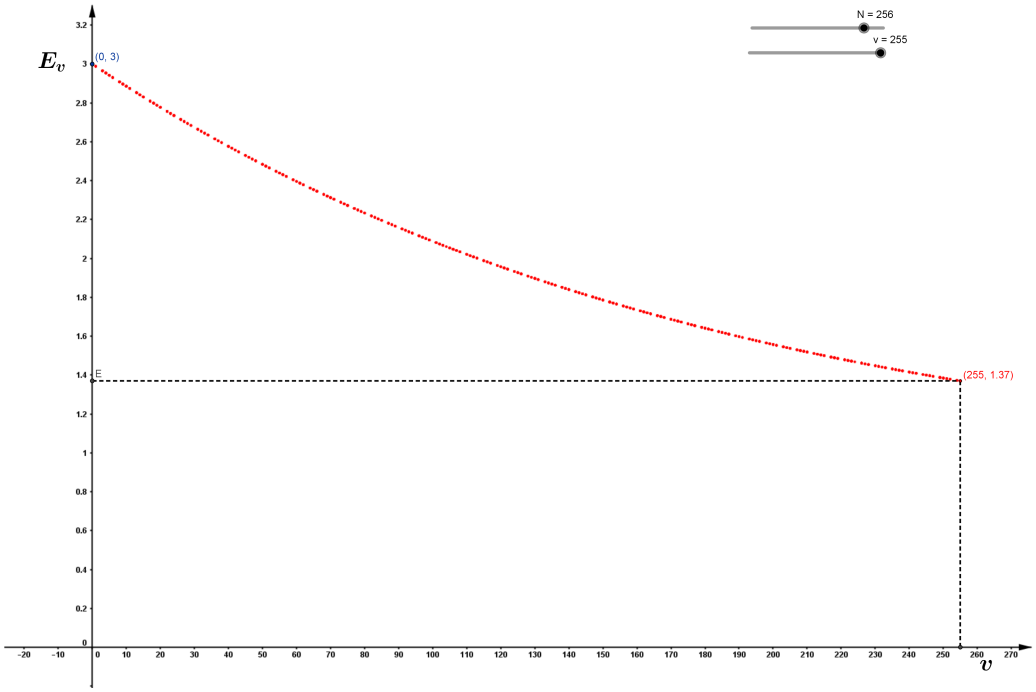
Do sada smo promatrali vjerojatnosti odabira zadanog indeksa permutacije pri pseudoslučajnom algoritmu KSA. Sada ćemo promatrati kolika je vjerojatnost da odabrana vrijednost permutacije sudjeluje u zamjeni.

Teorem 4.2.19. *Vjerojatnost da se vrijednost v točno jednom pojavljuje kao i-ti ili j-ti element permutacije u KSA jednaka je* $\frac{2v}{N} \cdot \left(\frac{N-1}{N}\right)^{N-1}$, *za* $0 \leq v \leq N-1$.

Teorem 4.2.20. *Za* $0 \leq v \leq N-1$, *očekivani broj puta kada i-ti ili j-ti element permutacije u KSA poprima vrijednost v dan je sa*

$$E_v = 1 + \left(\frac{2N-v}{N}\right) \cdot \left(\frac{N-1}{N}\right)^{v}.$$

Uočimo da se vrijednosti E_v smanjuju od 3 do 1.37 kako se v povećava od 0 do 255 (slika 4.2).



Slika 4.2: Teorem 4.2.20

Eksperiment proveden u 100 milijuna pokušaja s ključevima duljine $l = 16$ pokazuju da se teoretski i eksperimentalno dobiveni rezultati poklapaju (detaljnije informacije dostupne su u [5]).

4.3 Pronalazak ključa

Nakon analize KSA, želimo vidjeti može li se na temelju poznatih informacija o trenutnom stanju permutacije S pronaći tajni ključ, što je jedan od Evinih ciljeva. Dapače, pronalazak ključa najjači je oblik napada na bilo koju protočnu šifru. Iako bi se možda moglo pretpostaviti da iz, na prvi pogled slučajne permutacije, nije moguće saznati nikakve korisne informacije o tajnom ključu, pokazuje se da je KSA poprilično loša u ovom pogledu.

U idućem poglavlju pokazat ćemo da je PRGA reverzibilna, tj. da je poznavajući bilo koje trenutno stanje permutacije S moguće otkriti početno stanje permutacije nakon završetka KSA, a neposredno prije početka PRGA-e. Dakle, preostaje nam utvrditi postoji li način otkrivanja elemenata niza K ako su poznate vrijednosti od S_N. Postoji nekoliko metoda rješavanja ovog problema.

Metoda pronalaska ključa rješavanjem sustava jednadžbi

2007. godine Paul i Maitra prvi su dali rezultate o ovom pitanju. Njihova ideja je pronalazak ključa rješavajući sustav jednadžbi oblika $S_N[y] = f_y$. Inspiraciju za ovakav pristup autori su dobili proučavajući vezu između elemenata niza K i početnog stanja PRGA-e. Postupak se sastoji u odabiru prikladnih jednadžbi sa poznatim vrijednostima $S_N[y]$ koje se zatim rješavaju te se dobivaju nepoznanice $K[i]$. Glavni izazov ovog pristupa je odabir rješivog sustava jednadžbi. Nakon dobivanja rješenja sustava, njegova točnost lako se provjerava provođenjem algoritma KSA koji za ulazne podatke ima niz K čiji su elementi dobiveni rješavanjem sustava.

Metoda pronalaska ključa rješavanjem jednadžbi razlike

Ovaj pristup dodatno je poboljšan 2008. godine kada je predstavljena metoda tzv. jednadžbi razlike (eng. *difference equations*). Princip ove metode zasniva se na tome da se uzmu sustavi jednadžbi opisani u prethodnoj metodi te se uzimaju razlike svih mogućih parovi jednadžbi. Neka je

$$C_y = S_N[y] - \frac{y \cdot (y+1)}{2}.$$

Tada se jednadžbe oblika $S_N[y] = f_y$ mogu zapisati u obliku

$$K[0 \uplus y_1] = C_{y1},$$

$$K[0 \uplus y_2] = C_{y2},$$

gdje je $K[a \uplus b] = \sum_{x=a}^{b} K[x]$. Oduzimanjem jednadžbi u ovom obliku dobivaju se jednadžbe oblika

$$K[0 \uplus y_2] - K[0 \uplus y_1] = K[y_1 + 1 \uplus y_2] = C_{y2} - C_{y1}$$

Može se pokazati da je za ovako odabrane jednadžbe vjerojatnost da je dobivena jednadžba točna veća nego u prethodnoj metodi. Ova metoda je efikasnija od prethodne jer se pokazuje da je sustave jednadžbi razlike lakše riješiti nego sustave iz prethodne metode, a vjerojatnost da su dobiveni sustavi točni jednaka je vjerojatnosti da su sustavi iz prve metode točni.

Metoda pronalaska ključa grupiranjem elemenata od K

Prethodna metoda iste je godine poboljšana tako da se vrijednost j pokuša pogoditi razmatrajući sljedeće događaje:

- $j_{y+1} = S_N[y]$
- $j_{y+1} = S_N^{-1}[y]$
- $j_{y+1} = S_N[S_N[y]]$
- $j_{y+1} = S_N^{-1}[S_N^{-1}[y]]$
- $j_{y+1} = S_N[S_N[S_N[y]]]$
- $j_{y+1} = S_N^{-1}[S_N^{-1}[S_N^{-1}[y]]]$

Iz dvije uzastopne vrijednosti j_y i j_{y+1} dobiva se 36 kandidata za vrijednost $K[y]$. Svakom od 36 kandidata pridružuje se težina s obzirom na njegovu vjerojatnost. Metoda zatim nastavlja s rješavanjem sustava jednadžbi razlike, no umjesto direktnog rješavanja sustava metoda se sastoji u tome da se prvo pokuša pogoditi suma elemenata od K. Grupu od m elemenata s najvećim težinama smatramo pogođenima, a ostale elemente od K dobivamo rješavanjem sustava s $l - m$ nepoznanica. Ova metoda još je efikasnija od prethodne te je pronalazak elemenata od K brži nego u prethodnoj metodi, a vjerojatnost uspjeha je jednaka.

Metoda pronalaska ključa bajt po bajt

Ova metoda također rješava sustav jednadžbi razlike i razmatra sve moguće vrijednosti od $S_N[y]$ i $S_N^{-1}[y]$ te sužava izbor za moguće vrijednosti j u svakoj od rundi KSA na samo dva elementa iz $\mathbb{Z}_N$ sa konstantnom vjerojatnošću uspjeha , većom od 0.37. Procjenjene

vrijednosti za dva uzastopna j elementa ukupno daju 4 kandidata za pripadni element od K. Kako se svaki element ponavlja barem svakih $\lfloor \frac{N}{l} \rfloor$ puta , može se generirati tablica frekvencija za svaki element od K te tako dobiti mnogo kandidata za ključ.

Dvosmjerna metoda pronalaska ključa

2009. godine predstavljena je dvosmjerna metoda pronalaska ključa. Ideja ove metode je da se elementi od K $K[0], K[1], \ldots K[l-2]$ pogode korištenjem lijeve strane permutacije S, a da se preostali elementi od K ($K[l-1], \ldots$) pokušaju pogoditi istovremenim korištenjem desne strane permutacije S. Dakle, ključ pronalazimo istovremeno koristeći dvije strane od S (odakle je ova metoda i dobila naziv). Da bismo provjerili točnost ove metode, nakon što pogodimo određen broj elemenata od K, pokrenemo KSA te čekamo dok ne naiđemo na otprije poznatu vrijednost j_{y+1}. Ako se njegova vrijednost podudara s računski dobivenom vrijednošću j_{y+1}, nastavljamo s pogađanjem elemenata od K. U suprotnom, djelomično sastavljeni dio od K odbacujemo te postupak započinjemo iznova. Poznate vrijednosti j_{y+1} služe kao svojevrsni filteri za odvajanje (odbacivanje loših) dobrih i loših kandidata za elemente od K. S obzirom na to da se potraga za potencijalnim kandidatima odvija u dva smjera, za očekivati je da je ova metoda poprilično efikasna. I doista, ova metoda pronalazi elemente od K s vjerojatnošcu od 0.1409, što je gotovo dvostruko bolje od najbolje prethodno dobivene vjerojatnosti od 0.0745.

Poglavlje 5

PRGA

5.1 Reverzibilnost PRGA-e

Algoritam PRGAUnatrag

1: $i_\tau^G \leftarrow \tau mod N$
2: **for** $j_\tau^G = 0$ **to** $N-1$ **do**
3: $\quad i \leftarrow i_\tau^G$
4: $\quad j \leftarrow j_\tau^G$
5: $\quad S \leftarrow S_\tau^G$
6: $\quad r \leftarrow \tau$
7: $\quad$ Sve dok ne postane $r = 0$ ponavljaj
8: $\quad$ `SWAP` $(S[i], S[j])$
9: $\quad j \leftarrow S[i]$
10: $\quad i \leftarrow i - 1$
11: $\quad r \leftarrow r - 1$
12: $\quad$ **if** $j = 0$ **then**
13: $\quad\quad$ prijavi S kao kandidata za S_N

Pokazuje se da, ako su nam, u bilo kojem trenutku provođenja PRGA-e, poznate sljedeće informacije: sadržaj permutacije S, broj do sada generiranih elemenata niza ključeva (što je u vezi s indeksom i) te vrijednost indeksa j, moguće provesti algoritam PRGA-e unatrag i tako dobiti permutaciju S nakon završetka KSA. Nakon što smo dobili S, elemente od K dobivamo primjenom neke od metoda pronalaska ključa opisanih u prethodnom poglavlju. Pretpostavimo da nam je poznato stanje PRGA-e nakon τ rundi. Tada možemo otkriti sadržaj permutacije S_N koristeći algoritam **PRGAUnatrag**. Čak i ako nam nisu poznate vrijednosti indeksa j, možemo provesti sustavnu pretragu elemenata od $\mathbb{Z}_N$ i za svaki od njih provesti ovaj algoritam, a ako algoritmom dobijemo da vrijedi $j = 0$, ovako dobivena permutacija je kandidat za S_N. Očekuje se da ovakvom pretragom u prosjeku dobivamo

jednog kandidata za S_N. Primjetimo da nam za provedbu ovog algoritma nisu potrebni elementi tajnog ključa nego samo vrijednosti S_N i broj runde τ.

5.2 Analiza PRGA-e

Nedugo nakon što je RC4 procurio u javnost, Finney je uočio zanimljivu klasu stanja PRGA-e.

Finneyjevski ciklusi

Finney je pokazao da, ako su u proizvoljnoj rundi r PRGA-e zadovoljeni sljedeći uvjeti: $j_r^G = i_r^G + 1$ i $S_r^G[j_r^G] = 1$, onda polazne pretpostavke vrijede i u svakoj od sljedećih rundi, trenutno stanje PRGA-e je ciklus duljine $N(N-1)$, dok su izlazni elementi $z_{r+(N-1)}, z_{r+2(N-1)}, \ldots, z_{r+N(N-1)}$ pomak permutacije S_r^G. Njemu u čast, ovakvi ciklusi nazivaju se **Finneyjevski ciklusi**. S obzirom na to da u svakom trenutku ciklusa vrijede jednake relacije između promatranih elemenata, a k tome znamo i da je PRGA reverzibilna, lako se uočava da nije moguće prijeći iz stanja PRGA-e koje nije Finneyjevski ciklus u ono koje to jest. Primijetimo da inicijalizacija indeksa i i j na početku PRGA-e onemogućava nužne preduvjete za realizaciju Finneyjevskih ciklusa.
Jedan od važnijih rezultata koji ukazuje na nedostatke PRGA-e je teorem pznat pod nazivom **Jenkinsova korelacija**.

Teorem 5.2.1. *(**Jenkinsova korelacija**) Nakon r-te runde PRGA-e vrijedi*

$$P(S_r^G[j_r^G] = i_r^G - z_r) = P(S_r^G[i_r^G] = j_r^G - z_r) \approx \frac{2}{N}.$$

Dokaz. Dokaz ovog teorema može se vidjeti u [5]. □

Primijetimo da se prethodni teorem može zapisati u obliku

$$P(z_r = r - S_r^G[j_r^G]) \approx \frac{2}{N},$$

što se pokazuje izuzetno korisnim jer veza između z_r i $S_{r-1}[r]$ vodi do veze između $z - r$ i elemenata niza ključeva.

Razmotrimo sada distribuciju ulaznih i pripadnih izlaznih podataka u jednom koraku PRGA-e. U jednom koraku PRGA-e indeks i^G poprima vrijednost $i^G + 1$, dok j^G poprima vrijednost $j^G + S^G[i^G + 1]$. Uvedimo oznake $u = S^G[i^G] = S^G[i^G + 1]$ i $v = S^G[j^G] = S^G[j^G + u]$. Tada, nakon zamjene elemenata permutacije S, vrijedi $S^G[j^G + u] = u$ i

$S^G[i^G + 1] = v$. Neka $\psi(i^G, j^G, z)$ označava broj permutacija $\mathbb{Z}_N$ (kojih ukupno ima $N!$) takvih da je za zadane vrijednosti i^G i j^G prije provedbe koraka algoritma PRGA-e, izlazni podatak tog dijela algoritma upravo z. Tada vrijedi rezultat:

Teorem 5.2.2. *Neka je N paran. Tada vrijedi:*

Ako je i^G paran, onda je

$$\psi(i^G, j^G, z) = \begin{cases} (N-1)! - (N-2)!, za\ z = j^G\ i\ z = i^G + 1 - j^G \\ (N-1)! + 2(N-3)!,\ inače \end{cases}$$

Ako je i^G neparan i vrijedi $j^G = \frac{i^G+1}{2}$ ili $j^G = \frac{i^G+1+N}{2}$, onda vrijedi

$$\psi(i^G, j^G, z) = \begin{cases} 2(N-1)! - (N-2)!, za\ z = j^G \\ (N-1)! + 2(N-2)!, za\ z = j^G + \frac{N}{2} \\ (N-1)! - (N-2)! + 2(N-3)!,\ inače \end{cases}$$

Ako je i^G neparan i vrijedi $j^G \neq \frac{i^G+1}{2}$ i $j^G \neq \frac{i^G+1+N}{2}$, onda vrijedi

$$P(z_1 = v) = \begin{cases} 2(N-1)! - (N-2)!, za\ z = j^G,\ \ z = i^G + 1 - j^G,\ z = \frac{i^G+1}{2},\ z = \frac{i^G+1+N}{2} \\ (N-1)! + 4(N-3)!,\ inače \end{cases}$$

Primjetimo da je pretpostavka teorema da je N paran broj. Analogna tvrdnja vrijedi i za neparne N, a za dodatne informacije konzultirajte [5].

Korolar 5.2.3. *Neka su permuacije prije svake od rundi PRGA-e uniformno distribuirane i slučajne. Tada vrijedi:*
(1) $P(j^G = z | i^G\ je\ neparan) \geq \frac{1}{N} + \frac{1}{N^2}$
(2) $P(j^G = z | i^G\, jeparan) \leq \frac{1}{N} - \frac{1}{N^2}$
(3) $P(j^G = z | 2z = i^G + 1) = \frac{2}{N} + \frac{1}{N(N-1)}$

Navedeni rezultati govore da događaj $j^G = z$ razlikujemo od slučajnog događaja i to s vjerojatnošću od približno $\frac{1}{N^2}$. Za $N = 256$ teoretski dobivene vjerojatnosti u prethodnom korolaru su redom 0.00392151, 0.00389099 i 0.00779718, dok se eksperimentalno dobiveni rezultati (koji se nalaze u [5]) malo razlikuju. Razlog za to je činjenica da teoretski dobivene vrijednosti rezultate dobivaju s obzirom na promatranih 256! različitih permutacija, što je prevelik broj za korištenje i rezultati ne mogu biti eksperimentalno dobiveni u razumnom vremenu. Eksperimentalno dobiveni rezultati provedeni su na milijun ključeva

duljine 16 bajtova, pri čemu se u obzir uzelo milijun permutacija (što je značajno manje od 256!)

Može se pokazati da su u svakoj od rundi PRGA-e sljedeći događaji ekvivalentni: $S_r^G[i_{r+1}^G] = i_r^G - j_r^G + 2$, $j_{r+1}^G = i_r^G - 2$ i $j_{r+2}^G = 2j_{r+2}^G - j_r^G$.

Koristeći ove tvrdnje pokažimo da vrijedi rezultat:

Teorem 5.2.4. $P(j_{r+2}^G = 2 \cdot i_r^G + 4 - j_r^G) = \frac{2}{N}$.

Dokaz. Događaj $j_{r+2}^G = 2 \cdot i_r^G + 4 - j_r^G$ može nastupiti u jednom od dva međusobno disjunktna slučaja: **1. slučaj:** $S_r^G[i_{r+1}^G] = i_r^G - j_r^G + 2$ ili **2. slučaj:** Vrijedi $S_r^G[i_{r+1}^G] \neq i_r^G - j_r^G + 2$ no i dalje je slučajnim odabirom $j_{r+2}^G = 2j_{r+2}^G - j_r^G + 4$.

1. slučaj $S_r^G[i_{r+1}^G] = i_r^G - j_r^G + 2$ Zbog ekvivalencije prethodno opisanih događaja, vrijedi:

$$j_{r+2}^G = 2 \cdot j_{r+1}^G - j_r^G = 2(i_r^G + 2) - j_r^G = 2i_r^G - j_r^G + 4$$

Vjerojatnost ovog događaja je $P(S_r^G[i_{r+1}^G] = 2 \cdot i_r^G + 4 - j_r^G) = \frac{2}{N}$.

2. slučaj

Prema prethodno opisanim međusobno ekvivalentnim događajima,iz $S_r^G[i_{r+1}^G] \neq i_r^G - j_r^G + 2$ slijedi da je $j_{r+2}^G \neq 2j_{r+2}^G - j_r^G$. Ako pretpostavimo da j_{r+2}^G može poprimiti bilo koju od preostalih $N - 1$ vrijednosti s jednakom vjerojatnošću, dobivamo da je

$$P(S_r^G[i_{r+1}^G \neq i_r^G - j_r^G + 2) \cdot \frac{1}{N-1} = \left(1 - \frac{1}{N}\right) \cdot \frac{1}{N-1} = \frac{1}{N}.$$

Zbrajanjem vjerojatnosti dobivenih u ova dva slučaja dobiva se

$$P(j_{r+2}^G = 2 \cdot i_r^G + 4 - j_r^G) = \frac{1}{N} + \frac{1}{N} = \frac{2}{N}.$$

□

Andrew Roos je eksperimentalno pokazao da se s vjerojatnošću između 12% i 16% može ustvrditi da je vjerojatnost da je prvi element generiran s RC4, uz uvjet $K[0] + K[1] = 0$, jednak $K[2] + 3$. Ovaj rezultat teoretski je pokazan u [5].

Teorem 5.2.5. *Pretpostavimo da vrijedi* $K[0] + K[1] = 0$. *Tada je vjerojatnost da je vrijednost indeksa t, elementa permutacije S koji je prvi element generiran s RC4, jednaka 2 dana formulom*

$$P(t_1 = 2 | K[0] + K[1] = 0) > \left(\frac{N-1}{N}\right)^N.$$

Teorem 5.2.6. *Pretpostavimo da vrijedi* $K[0] + K[1] = 0$. *Tada vrijedi:*

$$P(z_1 = K[2] + 3 | K[0] + K[1] = 0) > \left(\frac{N-1}{N}\right)^{2N} \cdot \left(1 - \frac{1}{N} - \frac{1}{N^2}\right) + \frac{1}{N^2}.$$

Sljedeći rezultat pokazuje u kakvom su odnosu prvi element generiran PRGA-om i suma prvih triju elemenata niza ključeva.

Teorem 5.2.7. *Veza između prvog izlaznog elementa algoritma RC4 i prva tri elementa tajnog ključa, za proizvoljan tajni ključ dana je sa*

$$P(z_1 = K[0] + K[1] + K[2] + K[3]) \approx \frac{1}{N} \cdot \left(1 + \left(\frac{N-1}{N}\right)^{N} \cdot \left(1 - \frac{1}{N} - \frac{1}{N^2}\right) + \frac{1}{N^2}\right).$$

Dokaz. Literatura u kojoj se nalazi dokaz ovog teorema je [5]. □

Neka je X slučajna varijabla koja odgovara izlaznom elementu algoritma RC4, a neka je Y funkcija koja ovisi o elementima niza ključeva. Neka su vrijednosti od X i Y elementi od $\mathbb{Z}_N$. Dakle, prostor (X, Y) sadrži $N \cdot N = N^2$ točaka oblika (x, y). Zato što su i X i Y slučajne varijable, vrijedi $P(X = x, Y = y) = \frac{1}{N^2}$. Nadalje,

$$P(X = Y) = \sum_{x=0}^{N-1} P(X = x, Y = y) = \sum_{x=0}^{N-1} \frac{1}{N^2} = N \cdot \frac{1}{N^2} = \frac{1}{N}.$$

Za $N = 256$ dobiva se vrijednost 0.0039, dok su eksperimentalno dobivene vrijednosti u [5] značajno veće (približno 0.0058).

Pokazuje se da povezivanjem rezultata Jenkinsove korelacije i rezultata veze između vrijednosti elemenata permutacije i tajnog ključa dolazimo do otkrivanja informacija o z_r u kontekstu $r - f_y$.

Teorem 5.2.8. *U svakoj rundi PRGA-e vrijedi*

$$P(z_r = r - f_y) = \frac{1}{N} \cdot \left(1 + P(S_{r-1}^G[r] = f_r)\right).$$

Dokaz. Događaj $z_r = r - f_y$ može nastupiti u jednom od dva disjunktna slučaja.
1. slučaj $S_{r-1}^G[r] = f_y$ i $z_r = r - S_{r-1}^G[r]$
Prema Jenkinsovoj korelaciji, vrijedi

$$P(S_{r-1}^G[r] = f_y) \cdot P(z_r = r - S_{r-1}^G[r]) = P(S_{r-1}^G[r] = f_r) \cdot \frac{2}{N}.$$

2. slučaj $S^G_{r-1}[r] \neq f_y$ i slučajno se dogodilo $z_r = r - S^G_{r-1}[r]$.

$$P(S^G_{r-1}[r] \neq f_y) \cdot \frac{1}{N} = \left(1 - P(S^G_{r-1}[r] = f_r)\right) \cdot \frac{1}{N}.$$

Zbrajanjem vjerojatnosti dobivenih u prethodna dva slučaja dobiva se

$$P(z_r = r - f_y) = P(S^G_{r-1}[r] = f_r) \cdot \frac{2}{N} + \left(1 - P(S^G_{r-1}[r] = f_r)\right) \cdot \frac{1}{N} = \frac{1}{N} \cdot \left(1 + P(S^G_{r-1}[r] = f_r)\right).$$

□

Sljedeći rezultat u analizi generiranih elemenata niza ključeva govori o tome da je vjerojatnost da je prvi generirani element jednak nuli nešto manja od slučajne. Promotrimo prvo situaciju (dokaz nalazimo u [3]) u kojoj prvi generirani element nikada ne može biti nula.

Lema 5.2.9. *Ako je* $S^G_0[j^G_1] = 0$*, onda je* $z_1 \neq 0$.

Teorem 5.2.10. *Neka je početna permutacija PRGA-e slučajno odabrana. Tada je*

$$P(z_1 = 0) = \frac{1}{N} - \frac{1}{N^2}.$$

Dokaz. Razlikujemo dva slučaja:
1. slučaj Neka je $S^G_0[j^G_1] = 0$.
Tada, prema prethodnoj lemi, vrijedi

$$P(z_1 = 0 | S^G_0[j^G_1] = 0) = 0.$$

2. slučaj Neka je $S^G_0[j^G_1] \neq 0$.
Ako pretpostavimo da su vrijednosti z_1 uniformno distribuirane, dobivamo

$$P(z_1 = 0 | S^G_0[j^G_1] \neq 0) = \frac{1}{N}.$$

Tada je, po teoremu potpune vjerojatnosti,

$$\begin{aligned} P(z_1 = 0) &= P(zS^G_0[j^G_1] = 0) \cdot P(z_1 = 0 | S^G_0[j^G_1] = 0) + P(S^G_0[j^G_1] \neq 0) \cdot P(z_1 = 0 | S^G_0[j^G_1] \neq 0) \\ &= \frac{1}{N} \cdot 0 + \left(\frac{N-1}{N}\right) \cdot \frac{1}{N} \\ &= \frac{1}{N} - \frac{1}{N^2}. \end{aligned}$$

□

Analogno se dokazuje rezultat za $z_2 = 0$.

Teorem 5.2.11. *Neka je početna permutacija PRGA-e slučajno odabrana. Tada je*

$$P(z_1 = 0) \approx \frac{2}{N}.$$

Dakle, pokazali smo da događaj u kojem je prvi ili drugi generirani element jedank nuli nije slučajan. Dapače, rezultat se može poopćiti te je u [5] pokazano da vrijedi:

Teorem 5.2.12. *Neka je početna permutacija PRGA-e slučajno odabrana. Tada za svaku rundu r veću od 2 i manju od 256 vrijedi*

$$P(z_r = 0) \approx \frac{1}{N} + \frac{P(S_{r-1}^G[r] = r)}{N^2}.$$

Zanimljivo je da se može pokazati da je očekivani broj nula od treće do 256-e runde približno jednak 0.9906516923. Potpuna distribucija vjerojatnosti događaja $z_1 = v$ dokazana je u [3].

2013. godine Sen Gupta je dokazao (u [3]) da događaj $z_2 = 172$ nije slučajan.

Teorem 5.2.13. *Neka je početna permutacija PRGA-e slučajno odabrana. Tada je*

$$P(z_2 = 172) \approx \frac{1}{N} + \frac{0.28}{N^2}.$$

Mironov je (prema [3]) prvi eksperimentalno (bez dokaza) pokazao da događaj $z_1 = v$ ima sinusoidalnu distribuciju. U istom je radu po prvi puta u cijelosti opisana distribucija događaja $z_1 = v$.

Teorem 5.2.14. *Distribucija vjerojatnosti prvog elementa algoritma RC4 glasi*

$$P(z_1 = v) = Q_v + \sum_{X \in L} \sum_{Y \in T} P(S_0^G[1] = X \wedge S_0^G[X] = Y \wedge S_0^G[X + Y] = v),$$

gdje je $v \in \{0, 1, \ldots, N-1\}$, $L \in \{0, 1, \ldots, N-1\} \backslash \{1, v\}$, $T \in \{0, 1, \ldots, N-1\} \backslash \{0, X, 1-X, v\}$

$$Q_v = \begin{cases} P(S_0^G[1] = 1 \wedge S_0^G[2] = 0), \ za\ v = 0 \\ P(S_0^G[1] = 0 \wedge S_0^G[0] = 1), \ za\ v = 1 \\ P(S_0^G[1] = 1 \wedge S_0^G[2] = 0) + P(S_0^G[1] = v \wedge S_0^G[v] = 0) \\ +P(S_0^G[1] = 1 - v \wedge S_0^G[1 - v] = v), inače \end{cases}$$

Sen Gupta je prvi pokazao da postoji veza između duljine ključa l i vrijednosti elemenata permutacije S, tj. da je vjerojatnost događaja $S_l^G[l] = N - l$ različita od $\frac{1}{N}$. Također je pokazao da događaj $P(z_l = N - l)$ nije slučajan. Prethodni rezultat je u istom radu dodatno generaliziran te je u potpunosti prikazana distribucija događaja $P(z_{xl} = N - xl)$, gdje je $0 \leq x \leq \lfloor \frac{N}{l} \rfloor$.

Bibliografija

[1] H. Delfs, H.. Knebl, *Introduction to Cryptography. Principles and Applications*, Springer, 2002.

[2] A. Dujella, M.. Maretić, *Kriptografija*, Element, 2007.

[3] S. S. Gupta, *Analysis and Implementation of RC4 Stream Cipher*, (2013), http://souravsengupta.com/pub/phd_thesis_2013.pdf.

[4] J. Lv, B. Zhang, D. Lin i E. Weber, *Some New Weaknesses in the RC4 Stream Cipher*, (2013), http://www.springer.com/cda/content/document/cda_downloaddocument/9783319051482-c2.pdf?SGWID=0-0-45-1447514-p176593438.

[5] G. Paul, S. Maitra, *RC4 Stream Cipher and Its Variants*, CRC Press, 2012.

[6] M. Stamp, R. M. Low, *Applied Cryptanalysis. Breaking Ciphers in the Real World*, Wiley, 2007.

[7] M. Benšić, N.. Šuvak, *Uvod u vjerojatnost i statistiku*,Sveučilište J.J. Strossmayera, Odjel za matematiku, 2014.

[8] K. Horvatić, N.. Šuvak, *Linearna algebra, I. dio*,Matematički odjel PMF-a Sveučilišta u Zagrebu i Hrvatsko matematičko društvo, 1995.

[9] Jednokratna bilježnica https://en.wikipedia.org/wiki/One-time_pad.

[10] A. Kerckhoffs http://www.petitcolas.net/kerckhoffs/index.html#english.

[11] RC4-Wikipedia https://en.wikipedia.org/wiki/Stream_cipher

[12] R. Rivest http://people.csail.mit.edu/rivest/.

[13] G. Vernam https://en.wikipedia.org/wiki/Gilbert_Vernam.

[14] Operacija XOR https://en.wikipedia.org/wiki/Exclusive_or.

[15] XOR kalkulator http://www.miniwebtool.com/bitwise-calculator/.

[16] RC4 alat za šifriranje i dešifriranje http://rc4.online-domain-tools.com/

[17] RC4-Wikipedia https://en.wikipedia.org/wiki/RC4

[18] http://encyclopedia.kids.net.au/page/rc/RC4_cipher

[19] svi shematski prikazi rađeni su u alatu dinamične geometrije GeoGebra, koji se može preuzeti na linku https://www.geogebra.org/

Sažetak

RC4 je najpopularnija vrhunska suvremena protočna šifra koju je 1987. godine osmislio Ronald Rivest. Posebnost ove šifre je u njezinoj jednostavnosti te je to jedan od razloga njene izuzetne popularnosti, kako u akademskom, tako i komercijalnom svijetu. Iako je od njenog stvaranja proteklo tridesetak godina, još uvijek je u fokusu istraživanja mnogih kriptoanalitičara diljem svijeta. U ovom radu opisan je opći koncept protočne šifre te je u trećem poglavlju korak po korak predstavljen algoritam RC4, koji se sastoji od dva dijela: KSA i PRGA. U četvrtom i petom poglavlju analizirane su KSA i PRGA.

Summary

RC4 is the most popular state-of-the-art stream cipher. It was created in 1987 by Ronald Rivest. Specialty of this cipher lies in its simplicity, which is probably one of the main reasons for its popularity in academic and commercial world. Although almost thirty years had passed since its creation, RC4 remains in research focus of many cryptoanalysts. This thesis presents concept of general stream cipher and step-by-step approach of RC4 algorithm in third chapter. RC4 algorithm consists of two parts: KSA and PRGA. Chapters four and five present analysis of KSA and PRGA, respectively.

Životopis

Jelena Krnjak rođena je 24. veljače 1993. godine u Zagrebu. U istom je gradu 2008. godine završila Osnovnu školu Ivana Cankara te je po završetku osnovnoškolskog obrazovanja dobila nagradu za najbolju učenicu generacije u području matematike i informatike. Iste je godine upisala IX. gimnaziju. Za vrijeme srednjoškolskog obrazovanja redovito je sudjelovala na natjecanjima iz matematike i geografije. Zbog ostvarenih iznimnih rezultata sudjelovala je na međunarodnom seminaru „Climate changes" u gradu Aurichu u Njemačkoj, u organizaciji Europskog doma i Europskog parlamenta mladih. Maturirala je 2011. godine kao učenica generacije i učenica s najboljim postignutim rezultatima na ispitu državne mature iz matematike i hrvatskog jezika. Iste godine upisuje preddiplomski sveučilišni studij matematike nastavničkog usmjerenja na Prirodoslovno-matematičkom fakultetu Sveučilišta u Zagrebu. 2013. godine pristupa studentskoj udruzi eSTUDENT gdje je, u sklopu Tima za prakse i pripravništva, sudjelovala u organizaciji i provedbi nekoliko projekata, uključujući Starter konferenciju i Priručnik za apsolvente. 2014. godine na istome fakultetu upisuje diplomski studij matematike nastavničkog usmjerenja. 2016. godine u časopisu Matka, namijenjenom mladim matematičarima, objavljen joj je članak pod nazivom „New York, New York".