UNIVERZITET CRNE GORE

Prirodno-matematički fakultet Podgorica

Dragoje Božović

Vizuelno-kriptografski algoritmi dijeljenja tajne

Specijalistički rad

Podgorica, 2014.

UNIVERZITET CRNE GORE

Prirodno-matematički fakultet Podgorica

Vizuelno-kriptografski algoritmi dijeljenja tajne

Specijalistički rad

Kriptografija

Mentor: dr Vladimir Božović

Dragoje Božović

Matematika i računarske nauke

Podgorica, septembar 2014.

Apstrakt

Vizuelna kriptografija je kriptografski metod kojim se vrši "skrivanje" (kriptovanje) podataka u nekom skupu slika, a rekonstrukcija zaštićenih, odnosno kriptovanih podataka se vrši direktnim, vizuelnim pregledom, bez računarskog izračunavanja. Algoritmi vizuelne kriptografije rješavaju problem dijeljenja tajne i karakteriše ih visok stepen sigurnosti i otpornosti. U drugom poglavlju je dat pregled osnovnih kriptografskih tehnika koje se bave sigurnom razmjenom podataka. U trećem poglavlju predstavljen je problem dijeljenja tajne i riješen korišćenjem Šamirovog algoritma dijeljenja tajne. Matematička pozadina i računarski model slike koji se koristi za predstavljanje podataka sa kojima rade vizuelno-kriptografski algoritmi data je u četvrtom poglavlju. Rješenja problema dijeljenja vizuelne tajne interpretirane kao binarna slika dato je kroz Naor-Šamirov algoritam i Kafri-Kerenov algoritam u petom poglavlju. Osim predstavljanja pokazana je složenost i korektnost ovih algoritama i data su neka moguća proširenja u odnosu na njihove originalne koncepte.

Abstract

Visual cryptography is cryptography method which is used for data encryption of some image set, where decryption of encrypted data is possible directly by human vision without computer usage. Visual cryptografic algorithms are capable to solve secret sharing scheme problem and distinguished by high level of security and resistance. Second chapter gives us an overview of basic cryptographic techniques which are primarily used for securing data exchange. In third chapter secret sharing problem is presented and solved with Shamir's secret sharing scheme. Mathematical background of visual cryptography and computer model of picture are sections in chapter four. Main algorithms for visual cryptography presented in fift chapter are Naor-Shamir and Kafri-Keren's visual cryptography scheme followed by Chen-Tsao's algorithm for binary images and image channeling scheme for picture in colors.

Sadržaj

Glava 1

Uvod

Snažan razvoj informacionih tehnologija omogućava nam razmjenu i obradu velike količine podataka. Zbog širenja računarskih mreža i razvoja tehnika napada, svaki računar koji je priključen na njih potencijalno je ugrožen, a prevashodno su ugroženi podaci. Kriptografija nam između ostalog služi kako bismo zaštitili podatke od neovlašćenog pristupa i izmjene. Da bismo podatak zaštitili od neželjenog uništenja, potrebno je da napravimo njegove kopije na što više lokacija. Proporcionalno sa povećanjem broja lokacija, povećava se rizik od neovlašćenog pristupa i ovakav koncept je suprotan konceptu sigurnosti u kome broj kopija treba da bude što manji. Šeme *dijeljenja tajne* prevazilaze ovaj problem, tako što na lokacijama ne ostavljaju kompletan podatak, već njegove djelove. Algoritmi vizuelne kriptografije poželjni su kao alati za rješavanje problema dijeljenja tajne jer spajaju ideje visoke sigurnosti,a imaju veliku sličnost sa bajt simetričnom enkripcijom i vrlo jednostavne mehanizme za dekriptovanje tajne. Nedostaci vizuelno-kriptografskih algoritama se ogledaju u tome da oni uvećavaju ili degradiraju kvalitet podataka, pa se konstantno traže načini da se smanji robusnost kriptovanih, odnosno poveća tačnost dekriptovanih podataka, ali ne na uštrb sigurnosti i autentičnosti.

U radu je napravljen pregled kriptografskih tehnika koje služe da bi se nekom podatku koji treba da razmijenimo ili sačuvamo obezbijedila sigurnost. Detaljno je objašnjen problem dijeljenja tajne, kroz analizu osnovnog algoritma - Šamirov algoritam za dijeljenje tajne. Iz matematičkog modela slike koji je dat kao funkcija konstruisan je osnovni model slike koji se koristi u računarstvu, matrica. Svi koraci ovog postupka su objašnjeni, kao i produkt svakog koraka. Objašnjen je i kodni model zapisa raznih tipova slike. Uveden je pojam algebarske strukture Bulova algebra, koja služi za matematički korektno predstavljanje operacija koje se vrše sa slikom. Sa gore napisanim poglavljima stekli su se uslovi da svaki pojam koji se koristi u vizuelnoj kriptografiji bude čitaocu prethodno uveden i korektno definisan. Naor-Šamirov i Kefri-Kerenov algoritam izabrani su kao osnovni algoritmi vizuelne kriptografije kojima se rad bavi. Osim objašnjenja ovih algoritama, urađeno je izračunavanje komplekstnosti i njihova analiza. Čen-Tsaov algoritam vizuelne kriptografije i pretvaranje binarnih slika u slike u boji dati su kao dopuna osnovnih algoritama, odnosno kao njihovo proširenje. Ovakvim pristupom napravljena je dobra osnova da čitalac koji nema specijalističkih znanja iz oblasti vizuelne kriptografije može razumjeti obrađivane teme, odnosno i ako ne postoje specijalni preduslovi za njihovo razumijevanje. Pošto su dati i pseudokodovi za određene obrađivane algoritme, put do njihove implementacije na računaru je skraćen.

Glava 2

Uvod u kriptografiju

Kriptografija je naučna disciplina koja se bavi proučavanjem metoda, uglavnom matematičkih, za zaštitu i očuvanje tajnosti podataka. *Kriptoanaliza* je, nasuprot kriptografiji, disciplina koja istražuje načine za probijanje i zaobilaženje kriptografskih metoda. Kriptografija i kriptoanaliza zajedno čine nauku zvanu *kriptologija*. Elementi kriptografije počeli su da se pojavljuju čim je nastala potreba za tajnošću pisanih informacija, pa zato i korijeni kriptografije sežu daleko u istoriju, a prvi očuvani tragovi kriptovanja podataka datiraju iz 5. vijeka p.n.e. *Kriptovanje* je primjena određenog kriptografskog algoritma (metoda, šeme) na željeni podatak. Podatak koji želimo kriptovati može biti različitog tipa: pisani podatak, podatak na računaru u određenom formatu, slika itd. i ovo početno čitljivo stanje podatka naziva se *otvoreni tekst (eng. plain text)*. Kod kriptovanja razlikujemo dvije transformacije: enkripciju i dekripciju. Postupkom *enkripcije* ili *šifrovanja* otvoreni tekst se transformiše u *kriptovani tekst (kriptovanu poruku)*. Kriptovana poruka se postupkom *dekripcije* ili *dešifrovanja* vraća u otvoreni tekst.

$$\text{otvoreni tekst} \xrightarrow{enkripcija} \text{kriptovani tekst} \xrightarrow{dekripcija} \text{otvoreni tekst}$$

Da bismo osigurali tajnost podatka tokom razmjene, kriptovanjem ga transformišemo koristeći unaprijed zadati *ključ*. *Ključ* nam omogućava da prilikom ekripcije i dekripcije vršimo transformaciju otvorenog teksta u kriptovani tekst i obratno. Podatak

koji kriptujemo S sastoji se od znakova. Skup svih znakova koje koristimo za pisanje poruke čini *Azbuku* A za kreiranje podatka, koja nije obavezno sastavljena samo od slova. Matematički govoreći, enkripcija je preslikavanje E, gdje se pomoću ključa enkripcije K dati podatak S preslikava u kriptovani podatak S'. Dekripcija je preslikavanje D , koja podatak S' pomoću ključa dekripcije K' preslikava u S, pa funkcija E mora biti invertibilna.

$$E\colon\ K \times S \to S'$$

$$D\colon\ K \times S' \to S$$

za koje važi da za svako $k \in K$, postoji $k' \in K'$ tako da

$$D(k', E(k, S)) = S$$

za svako S koje se može generisati koristeći A.

2.1 Podjela kriptografskih algoritama

Jedan od osnovnih zadataka u kriptografiji, koji se tiče kriptovanja podatka, jeste omogućiti dvijema osobama (Alisa i Bob) razmjenu podatka na takav način da osoba koja bi mogla da dođe u posjed kriptovanog podatka u toku njegovog transporta (Eva) ne može razumjeti njegovu sadržinu. Transport nam predstavlja komunikacioni put poruke: za podatke na računaru to je računarska mreža između dva računara koji razmjenjuju poruke, a za slanje pisma, mreža pošta. Pretpostavka je da je ovaj komunikacioni kanal nesiguran, te da nam potencijalni napadač Eva uvijek može presresti poruke i čitati njihov sadržaj. Osim kriptovanja podataka različitim kriptografskim tehnikama se obezbjeđuje autentičnost pošiljaoca i poruke. Zavisno od kriptografske metode koja se koristi za kriptovanje podatka i načina na koji se primjenjuje, postoje

podjele kriptografskih algoritama. Jedna od podjela odnosi se na vrstu ključa koji se koristi, pa razlikujemo algoritme *asimetrične* kriptografije i algoritme *simetrične* kriptografije.

Kod asimetričnih algoritama, za razliku od simetričnih, ključ kojim se otvoreni tekst transformiše u kriptovani i ključ kojim se iz kriptovanog dobija otvoreni tekst nisu isti. U razmjeni poruka u kojoj je tajnost podatka osigurana primjenom asimetričnih algoritama, Alisa i Bob imaju po par ključeva, *javni* i *privatni*. Prije početka komunikacije Alisa i Bob treba da razmijene javne ključeve. Za enkripciju nad otvorenim tekstom Alisa koristi Bobov javni ključ ($k_j(B)$) i tako kriptovanu poruku šalje. Bob prima kriptovanu poruku, a da bi došao do originalnog sadržaja poruke koju mu je Alisa namijenila, koristi svoj privatni ključ ($k_p(B)$) da primljenu poruku dekriptuje. Suprotan smjer slanja poruka, od Boba ka Alisi, identičan je proces, s tim što se za enkripciju koristi Alisin javni ključ ($k_j(A)$), a za dekripciju Alisin ($k_j(A)$) privatni ključ.

$$\text{A} \underset{D(k_t(A),S')=S}{\overset{S}{\longleftrightarrow}} \underset{S'}{\overset{E(k_j(B),S)=S'}{\longleftrightarrow}} \underset{\text{transport}}{\overset{\text{transport}}{\longleftrightarrow}} \underset{E(k_j(A),S)=S'}{\overset{S'}{\longleftrightarrow}} \underset{S}{\overset{D(k_p(B),S')=S}{\longleftrightarrow}} \text{B}$$

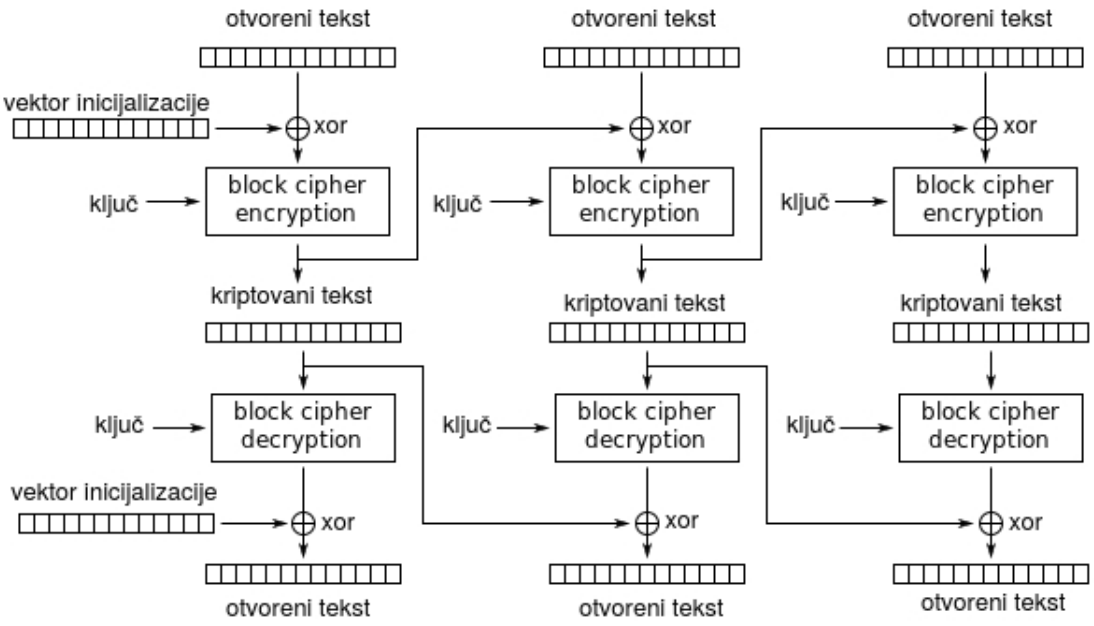
Za razliku od asimetrične kriptografije, algoritmi simetrične kriptografije koriste isti ključ sa enkripciju i dekripciju poruke. Ova različitost algoritama u smislu tipa ključa koji se koristi, definiše i njihove osnovne primjene. Za razmjenu poruka najčešće se koriste algoritmi asimetrične kriptografije, jer se za komunikaciju sa više osoba može koristiti ista kombinacija javnog-privatnog ključa, a privatni ključ se ne mora razmjenjivati sa potencijalnim pošiljaocem da bi se obezbijedilo kriptovanje. Algoritmi simetrične kriptografije se takođe mogu koristiti za razmjenu poruka, ali se u tom slučaju mora obezbijediti sigurna razmjena ključa, za šta se najčešće koristi asimetrični algoritam. Da bi se podigao stepen sigurnosti, razmjena poruka sa svakom

stranom kod simetričnih algoritama podrazumijava različite ključeve, što proces čini složenijim i stvara problem čuvanja velikog broja ključeva. Zato najčešću primjenu algoritama simetrične kriptografije nalazimo u kriptovanju podataka koji su snimljeni na računaru. Primjenom nekog od simetričnih alogritama pravimo kriptovani podatak. Ovako dobijen kriptovani podatak ne moramo čuvati na posebno 'tajnom' mjestu da bismo mu obezbijedili sigurnost, što nije bio slučaj dok je podatak bio u otvorenom tekstu tj. čitljiv. Kriptovanje nam omogućava da čuvamo više kopija podatka na različitim mjestima, tj. u različitim računarima i na taj način povećamo otpornost na njihov gubitak, a istovremeno se zaštitimo da njihov sadržaj ne dođe u posjed neželjenih čitalaca.

Prema načinu na koji se primjenjuju simetrični kriptografski algoritam na otvoreni tekst pravi se podjela na *bajt* i *blokovske* algoritme.

Blokovski simetrični algoritmi primjenjuju se na djelove otvorenog teksta koje nazivamo *blokovi*. Prije enkripcije vršimo podjelu otvorenog teksta na blokove iste dužine, koji su traženi da bi se obezbijedio ispravan rad algoritma, npr. DES algoritam radi sa blokovima dužine 64 bita. U slučaju da nam je posljednji blok manji od tražene dužine popunićemo ga slučajno generisanim vrijednostima iz Azbuke. Na ovaj način smo obezbijedili odgovarajući ulazni podatak koji enkriptujemo. Ako enkripciju vršimo tako što ćemo svaki blok podatka enkriptovati posebno i posmatrati svaki blok poruke kao posebnu poruku, algoritam radi u ECB *(eng. Electronic Code Blocker)* režimu rada. Na ovaj način se za dva bloka koji sadrže isti otvoreni tekst enkripcijom dobija identičan kriptovani tekst. Osim ECB-a, postoji i režim rada CBC *(eng. Chiper Block Chaining)*. Kod CBC režima samo se prvi blok otvorenog teksta enkriptuje ključem XOR-ovanim vektorom inicijalizacije i tako dobija enkriptovani blok, a svaki sljedeći blok za enkripciju se dobija primjenom operacije XOR (ekskluzivno ili) na trenutni blok otvorenog teksta i prethodno enkriptovani blok. Na ovaj

način se postiže da svaki enkriptovani blok zavisi od svih prethodno enkriptovanih blokova, pa se identični blokovi otvorenog teksta poslije enkripcije razlikuju.



Slika 2.1: Blok simetrični algoritam u CBC modu, enkripcija i dekripcija

Kod *bajt simetričnih algoritama* enkripciju vršimo tako što svaki bajt kriptovanog teksta dobijamo korišćenjem XOR operacije između jednog bajta otvorenog teksta i jednog bajta ključa. Za ovakvu vrstu algoritama ključ mora biti iste dužine kao i tekst koji enkriptujemo, pa se za generisanje ključa koriste algoritmi koji generišu slučajne nizove znakova iz Azbuke. Najčešće se za generisanje ključa koriste mašine na strani primaoca i pošiljaoca. Sigurnost simetričnih bajt algoritama zavisi od ključa i načina njihovog generisanja, pa se kriptoanaliza ovih algoritama svodi na izučavanje i testiranje algoritama za generisanje nizova slučajnih brojeva.

Glava 3

Problem dijeljenja tajne

U problemu *dijeljenja tajne* (eng. secret sharing (SS)), podatak (tajna) označen sa S je podijeljen na određen broj djelova n. Djelovi podatka označeni su sa D_i i=1...n i nazivaju se *ključevi*. Spajanjem dovoljnog broja ključeva rekonstruišemo tajni podatak S. Broj ključeva k koji je potreban da bi se tajna rekonstruisala zove se *prag*. Dio podatka označen kao *ključ* u *SS* šemi mora biti generisan na takav način da ne smije sadržati informaciju koja bi otkrila nešto o samoj tajni S ili nekom drugom ključu.

Definicija 1. *Zadatak dijeljenja tajnog podatka S na n djelova $D_1, D_2 \dots D_n$ treba da zadovolji sljedeće:*

1. *Sastavljanje bilo kojih k, $k \leq n$, ili više D_i djelova rekonstruiše početni podatak S.*

2. *Sastavljanje bilo kojih k - 1 ili manje D_i djelova ne smije da otkrije početni podatak S.*

Ovako definisan problem naziva se dijeljenje tajne sa pragom *k od n* . Ako je k=n, onda su potrebni svi djelovi da bi se S rekonstruisao, pa je prag jednak broju ključeva. SS šemu odlikuje visok stepen sigurnosti, otpornost na krađu podataka, jer

i sa *k - 1* ključeva ne smije se saznati S. Visok stepen pouzdanosti ogleda se u tome što i sa uništavanjem n - k ključeva možemo generisati S.

U osnovnoj, gore definisanoj SS šemi, svaki ključ ima isti prioritet. To znači da bilo koji podskup od k elemenata iz skupa ključeva n može biti kvalifikovan da otkrije tajnu S. Samim tim svi ključevi su ravnopravni i isto vrijedni. Kad postoji potreba da različito vrednujemo ključeve, vršimo modifikaciju osnovne SS šeme u šemu u kojoj ključevi imaju različit prioritet.

Primjer 1. *Tajnu S, $S \in N$, treba da podijelimo između četiri osobe A, B, C i D, tako da spajanje dva ključa osoba $\{A, B\}$, $\{B, C\}$ ili $\{C, D\}$ omogućava rekonstruisanje tajne, a da osobe $\{A, D\}$ spajanjem svojih ključeve ne mogu rekonstruisati tajnu.*

Nasumično biramo ključeve $\{D_1, D_1', D_2, D_2', D_3, D_3'\} \in N$ tako da zadovoljavaju uslov $D_i + D_i' = S$, $i = 1, 2, 3$. Ključ D_1 daćemo osobi A, ključeve D_1' i D_2 osobi B, ključeve D_2' i D_3 osobi C, a ključ D_3' osobi D. Ovakva podjela ključeva zadovoljava traženi uslov.

Familije skupova koji rekonstruišu tajnu, kao i proširenja tih skupova sa ostalim ključevima nazivaju se *kvalifikovane familije*. Ovakva struktura Γ koja ima pristup (eng. access structure), sastavljena je od podskupova $\{1, ..., n\}$. Tada su ključevi D_i podijeljeni po prioritetima i ovo se radi kad postoji potreba za dijeljenjem ključeva osobama na različitim hijerarhijskim nivoima ili potreba dijeljenja podataka na nepouzdanim lokacijama.

3.1 Šamirov algoritam za dijeljenje tajne

Prije nego što pređemo na vizuelne algoritme za dijeljenje tajne, predstavićemo osnovni algoritam dijeljenja tajne. Rješenje problema dijeljenja tajne 1979. predložili su nezavisno jedan od drugog Adi Shamir [7] i G Blakley. Bleklijev algoritam baziran

je na geomeriji hiperravni, a ideja je da se učesnicima algoritma podijele informacije o ravnima koje nijesu paralelne, dok je tajni podatak tačka njihovog presjeka. Godine 1983. Asmuth i Bloom predložili su algoritam koji se bazira na kineskoj teoremi o ostacima.

Šamirov algoritam je algoritam dijeljenja tajne *n od k*. Ovaj algoritam se bazira na interpolaciji polinoma. Neka su u tačkama x_i, $i = 0, 1, \ldots, n$ koje su poređane u rastućem redosljedu zadate vrijednosti neke funkcije $y_i = f(x_i)$, $i = 0, 1, 2, \ldots, n$. Zadatak interpolacije je sljedeći: Treba naći polinom $P_n(x)$ takav da

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n$$

koji aproksimira funkciju $f(x)$ ($f(x) = P_n(x)$) na intervalu $x \in [x_0, x_n]$, tako da u tačkama x_i $i = 0, 1, \ldots, n$ ima jednake vrijednosti sa njom $P_n(x_i) = f(x_i)$. Tačke x_i nazivaju se čvorovi interpolacije i važi tvrdnja da ako su čvorovi x_i različiti među sobom, tada postoji jedan i samo jedan polinom stepena ne većeg od n koji zadovoljava uslov $P_n(x_i) = f(x_i)$

Na osnovu gore navedenog jasno je da nam je k tačaka dovoljno da definišu polinom stepena $k - 1$. Pretpostavimo da je za okrivanje tajne dovoljno *k od n* učesnika koji su dobili po dio tajnog podatka S. Bez gubljenja osnovne pretpostavke naš tajni podatak će biti element konačnog polja F veličine P, za koje važi : $0 < k \leq n < P$; $S < P$ i P je (velik) prost broj. Biramo nasumično $k - 1$ broj iz skupa $a_1, \ldots, a_{k-1}$ prirodnih brojeva $a_i < P$ pri čemu je $a_0 = S$. Konstruišimo polinom $k - 1$ stepena

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + ... + a_{k-1} x^{k-1}$$

Napravićemo skup od n tačaka tako što ćemo izračunati za bilo koje i $i = 1, \ldots, n$

$(i, f(i))$. Svaki od djelova D_i u SS predstavljaće nam koordinate jedne tačke zadate sa $(i, f(i))$.

Na ovaj način smo postigli da sa svakim $k - podskupom$ skupa svih D_i korišćenjem interpolacije dobijemo sve koeficijente polinoma, pa i našu tajnu a_0. U 2 od 2 problemu dijeljena tajne jedan od učesnika dobija tačku sa koordinatama $(a_1, f(a_1))$, a drugi $(a_2, f(a_2))$. Ove tačke računamo za proizvoljnu funkciju $y = a_0 + ax$, na već opisan način. Znajući koordinate samo svoje tačke, tj. dijela tajne i pod pretpostavkom da je broj pravih koje prolaze kroz jednu tačku neograničen, niko od učesnika sa sigurnošću ne može tvrditi koja je od njih potrebna za rješenje SS, što nam je i cilj. Spajanjem informacija o objema koordinatama podijeljenih tačaka na liniji x_1,x_2 možemo izračunati njen položaj u R^2 ravni, a samim tim i a_0.

Traženje početnog koeficijenata a_0 na osnovu zadatih tačaka rješava se Lagranžovom interpolacijom. Za dati skup tačaka $x_i, y_i \ i = 0, 1, 2, \ldots, k-1$ Lagranžov interpolacioni polinom se može konstruisati korišćenjem sljedeće formule:

$$P(x) = \sum_{i=0}^{k-1} y_i \prod_{i \neq j} \frac{x - x_i}{x_{\mathrm{j}} - x_{\mathrm{i}}}$$

Glava 4

Matematička pozadina

Matematika primijenjena u vizuelnoj kriptografiji usko je povezana sa algebarskim strukturama. Algebarska struktura sastoji se od skupa i operacija koji su definisani nad tim skupom. Ove operacije zadovoljavaju jedno ili više svojstava (aksioma). U vizuelnoj kriptografiji, skup je obično skup piksela, dok je pridružena binarna operacija modelirana kao *'OR'* operacija. Predstavljanje monohromatske slike i operacija koje se vrše nad njom je modelirano preko algebarske strukture koja se naziva *Bulova algebra.*

4.1 Bulova algebra

Dat je skup S sa najmanje dva elementa, u oznaci 0 i 1, na kome su definisane dvije binarne operacije i jedna unarna operacija.

Definicija 2. *Na skupu S je definisana Bulova algebra ako za svako $x, y, z \in S$ važe sljedeće aksiome:*

1. $x + y = y + x$; $x * y = y * x$ *(Komutativnost)*

2. $(x + y) + z = x + (y + z)$; $(x * y) * z = x * (y * z)$ *(Asocijativnost)*

3. $x * (y + z) = (x * y) + (x * z)$; $x + (y * z) = (x + y) * (x + z)$ *(Distributivnost)*

4. $x + 0 = x$ *;* $x * 1 = x$ *(Neutralni element)*

5. $x + \bar{x} = 1$ *;* $x * \bar{x} = 0$ *(Inverzni element)*

Bulovu algebru na skupu S sa operacijama $+$ *,* $*$ *,*$^{-}$ *kraće označavamo kao četvorku (S,* $+$*,* $*$*,*$^{-}$*).*

4.1.1 Osnovni model Bulove algebre

Dat je skup $L = \{0, 1\}$. Uvedimo na skupu L binarne operacije $+$ (disjunkcija), $*$ (konjukcija) i unarnu operaciju^{-}na sljedeći način:

1. $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, $1 + 1 = 1$

2. $0 * 0 = 0$, $0 * 1 = 0$, $1 * 0 = 0$, $1 * 1 = 1$

3. $\bar{0} = 1$, $\bar{1} = 0$

Ovako definisane operacije na skupu L zadovoljavaju aksiome Bulove algebre iz *definicije 2.*

4.1.2 Matrični model Bulove algebre

Matrice $A = [a_{ij}]$, $i = 1, ..., n$ $j = 1, ..., m$, $B = [b_{ij}]$, $i = 1, ..., n$ $j = 1, ..., m$ dje je $a_{ij} \in \{0, 1\}$ i $b_{ij} \in \{0, 1\}$ zovemo Bulove matrice formata $n \times m$. Neka je M skup svih Bulovih matrica formata $n \times m$. Uvedimo na skupu M binarne operacije $+$ (OR), $\times$ (AND) i unarnu operaciju^{-}, na sljedeći način:

1. $A + B \stackrel{\text{def}}{=} [a_{ij} + b_{ij}]$

2. $A \times B \stackrel{\text{def}}{=} [a_{ij} * b_{ij}]$

3. $\bar{A} \stackrel{\text{def}}{=} [\bar{a_{ij}}]$

Uvedene operacije " $+$ ", " $\times$ " i " $^-$ " zadovoljavaju aksiome Bulove algebre iz *definicije 2.* Dokaz:

1. $A + B = [a_{ij} + b_{ij}] = [b_{ij} + a_{ij}] = B + A$

   $A \times B = [a_{ij} * b_{ij}] = [b_{ij} * a_{ij}] = B \times A$

2. $(A + B) + C = [(a_{ij} + b_{ij}) + c_{ij}] = [a_{ij} + (b_{ij} + c_{ij})] = A + (B + C)$

   $(A \times B) \times C = [(a_{ij} * b_{ij}) * c_{ij}] = [a_{ij} * (b_{ij} * c_{ij})] = A \times (B \times C)$

3. $A+(B\times C) = [a_{ij}+(b_{ij}*c_{ij})] = [(a_{ij}+b_{ij})*(a_{ij}+c_{ij})] = (A+B)\times(A+C)$

   $A\times(B+C) = [a_{ij}*(b_{ij}+c_{ij})] = [(a_{ij}*b_{ij})+(a_{ij}*c_{ij})] = (A\times B)+(A\times C)$

4. Neutralni element skupa M za operaciju '+' je Bulova matrica čiji su svi elementi nule. Označimo je sa $0 = [0]$.

   $A + 0 = [a_{ij} + 0] = [a_{ij}] = A$

   Neutralni element skupa M za operaciju '×' je Bulova matrica čiji su svi elementi jedinice. Označimo je sa $1 = [1]$.

   $A \times 1 = [a_{ij} * 1] = [a_{ij}] = A$

5. $A + \bar{A} = [a_{ij} + \bar{a_{ij}}] = [1] = 1$

   $A \times \bar{A} = [a_{ij} * \bar{a_{ij}}] = [0] = 0$

Dokazali smo da operacije " $+$ ", " $\times$ " i " $^-$ " na skupu M zadovoljavaju aksiome Bulove algebre, pa algebarska struktura na skupu M predstavlja model Bulove algebre. Za vizuelno kriptografske algoritme koji rade sa slikama, a objašnjeni su u radu, dovoljno je bilo da uvedemo "siromašniju" strukturu, polugrupu. Polugrupa je uređeni par $(S, +)$ skupa S, $S = \{0, 1\}$, i binarne operacije $+$ koja je zatvorena i asocijativna: $S \times S \to S$, odnosno svakom paru elemenata iz S pridružuje element iz S. Ovakav pristup je izbjegnut jer za neke transformacije u obradi slika polugrupa nije dovoljna.

4.2 Reprezentacija slike na računaru

Matematički gledano, slika je dvodimenzionalna funkcija intenziteta svijetla u vremenu $f(x, y)$, $0 < f(x, y) < \infty$. Slike koje vidi ljudsko oko su svijetlo reflektovano od objekata i ovo svijetlo zavisi od dvije komponente: ukupno svijetlo koje dolazi do objekta $i(x, y)$ i svjetlo koje se reflektuje od objekta $r(x, y)$, pa je matematički model za predstavljanje slika dat u obliku proizvoda:

$$f(x, y) = i(x, y)r(x, y)$$

$$0 < i(x, y) < \infty,\ 0 < r(x, y) < 1$$

Direktna implementacija ovakve pretpostavke u računarima trenutno nije moguća, zbog tehničkih ograničenja, pa se za predstavljanje slike u računarstvu koristi pojednostavljeni model, tj. diskretni model. Za transformaciju analognog signala, tj. slike kakvu naše oko percipira, u digitalni format koriste se dvije transformacije: semplovanje *(eng. Sampling)* i kvantizacija *(eng. Quantization)*.

Proces semplovanja nam dijeli neprekidnu funkciju $f(x, y)$ na intervale po x-osi koje nazivamo intervalima semplovanja i na ovaj način smo izvršili diskretizaciju vremena (prostora). Što je više intervala, odnosno što je semplovanje učestalije, digitalni reprezent slike ima mogućnost da bude vjerodostojniji stvarnoj slici.

Na računaru u procesu semplovanja neprekidnu funkciju $f(x, y)$ aproksimiramo matricom dimenzija $N \times M$, a element ove matrice naziva se *piksel*.

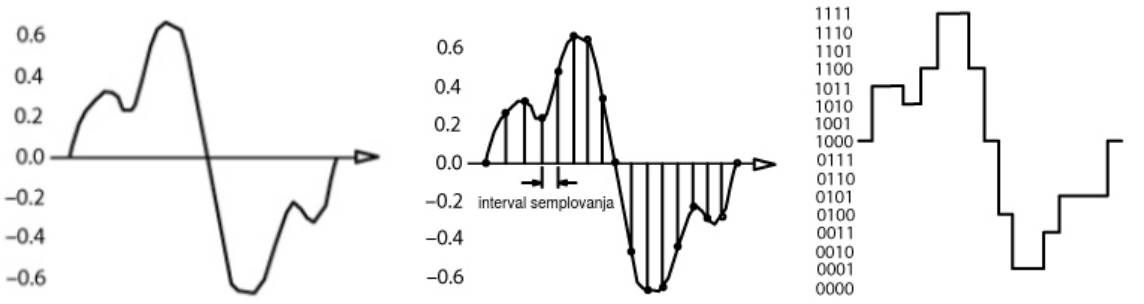
$$f(x, y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M) \\ f(1,0) & f(1,1) & \dots & f(1,M) \\ \vdots & \vdots & & \vdots \\ f(N,0) & f(N,1) & \dots & f(N,M) \end{bmatrix}$$

Konvencionalno *rezolucija slike*, apsolutna rezolucija, predstavlja broj piksela u matrici aproksimacije.

U procesu kvantizacije neprekidnu funkciju $f(x, y)$ diskretizujemo po y-osi tako što ćemo vrijednosti funkcije u intervalima semplovanja, piksele, ograničiti do nekog cjelobrojnog iznosa. Ovaj proces nam omogućava da određeni piksel u računaru predstavimo sa konačnim brojem bitova.

$$f(x, y) \approx \begin{bmatrix} f[0,0] & f[0,1] & \dots & f[0,M] \\ f[1,0] & f[1,1] & \dots & f[1,M] \\ \vdots & \vdots & & \vdots \\ f[N,0] & f[N,1] & \dots & f[N,M] \end{bmatrix}$$

Kvantizacija nam određuje *dubinu boje* slike (eng. Color Depth), pa zavisno od toga koliko memorije imamo na raspolaganju za predstavljanje jednog piksela, od 2^n ukupno, razlikujemo više tipova slika.



(a) Matematički model (b) Semplovanje (c) Kvantizacija piksela 4-bita

Slika 4.1: Digitalizacija slike

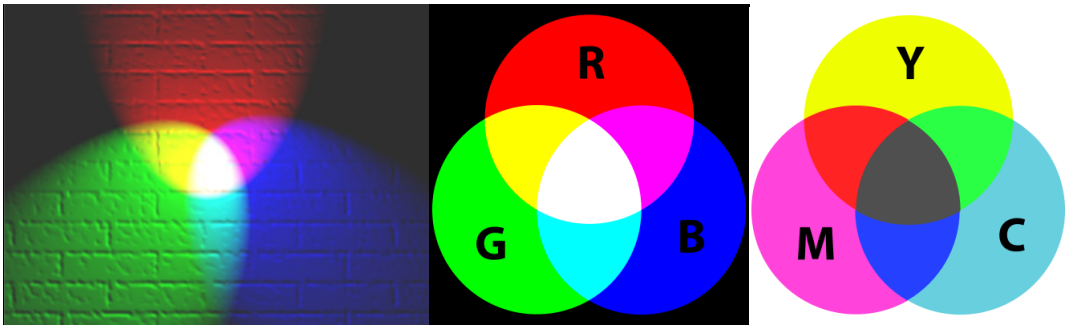
Semplovanje i kvantizacija zajedno čine *digitalizaciju (eng. Digitalization)* ili pretvaranje slike u digitalni format.

Binarna slika (eng. 1-bit monochrome) vrijednost piksela može uzimati iz skupa $\{0, 1\}$, tako da nam je potreban 1 ili 2^0 bita za reprezentaciju. Jasno je da ovakav zapis slike zauzima najmanje memorije, jer se njime mogu prikazati samo crni i bijeli

pikseli. Slika sa sivim tonovima *eng. grayscale* obično za piksel uzima vrijednosti iz domena $0 - 255$ i tada je veličina memorijske lokacije koja je potrebna za snimanje jednog piksela 2^3 bita.

Zavisno od vrste osnovnih boja - *komponenti* koje koristimo za predstavljanje slike u boji razlikujemo dva osnovna modela: aditivni i suptraktivni.

Ideja *aditivnog* modela je da se sve boje mogu predstaviti sa tri primarne - osnovne boje svjetlosti: crvena (eng. Red), zelena (eng. Green) i plava (eng. Blue) i ove boje se nazivaju *kanali*. Paradigma aditivnog modela je u tome što se bijela svjetlost sastoji od boja osnovnih kanala. Sve ostale obojene svjetlosti se mogu dobiti kombinacijom osnovnih svjetlost, tako što ćemo podešavati njihov intenzitet. Aditivni model se obično naziva RGB model i koristi se kada se boje grade dodavanjem komponenti svjetlosti, pa se koristi kao osnovni model za skladištenje digitalnih formata koje treba prikazati na nekom uređaju (monitor, projektor...).



(a) RGB projekcija svijetla (b) RGB (c) CMYK

Slika 4.2: RGB i CMYK model

Osnovne boje *suptraktivnog* modela su boje dobijene kombinacijom svjetlosti aditivnog modela, pa razlikujemo osnovne boje za ovaj model: tirkizna (eng. Cyan), purpurna (eng. Magenta) i žuta (eng. Yellow). Za model se u kompjuterskoj terminologiji koristi izraz CMY i on se primjenjuje kada se boje grade odbijanjem svjetlosti

o površinu objekta, a ne njenim zadržavanjem, pa ima primjenu u štampi slika. Za razliku od RGB modela, gdje se bijela boja dobija projekcijom osnovnih boja, kod CMY modela ne možemo dobiti bijelu boju, pa je pretpostavka da se štampa izvodi na bijeloj podlozi. Proširenje CMY modela crnom bojom (eng. black) dobijamo poznati CMYK model.

Slika u boji u komjuterskom modelu predstavljena je matricom $N \times N$, gdje je svaki element matrice vektor [Crveno,Zeleno,Plavo] koji sadrži informacije o intenzitetu osnovne boje u tom pikselu. Kod slika u boji predstavljenim u RGB modelu, zavisno od dubine boje, pravimo razliku između više formata koji su najčešće u primjeni:

1. Veoma limitiran zapis od $8-bita$ koji nam dozvoljava prikaz ukupno 256 boja. U ovom načinu zapisa su po tri bita rezervisana za crvenu i zelenu, dok su dva bita dostupna za zapis plave boje.

2. Zapis od $16-bita$ ima mogućnost prikazivanja 65536 različitih boja. Šest bita je rezervisano za zelenu i po pet za crvenu i plavu boju.

3. $24-bita$ za prikaz boja daje osnovnim bojama RGB modela po $8-bita$, svakoj ponaosob, za prikaz jednog piksela. Ovakav prikaz je dovoljan za savršenu reprodukciju slika, jer ljudsko oko razlikuje do 10 miliona nijansi boja.

4. $32-bita$ je nadogradnja modela $24-bit$, proširenog za $8-bita$ koji predstavljaju alfa kanal. *Alfa kanal* daje informaciju o mijenjaju intenziteta kanalu u pikselu kada se dva piksela preklope.

Samim procesom digitalizacije utičemo na diskretizaciju domena i dobijamo, zavisno od toga koliko smo uticali na ograničenja, manje ili više kvalitetnu sliku. Zadnji korak digitalizacije jeste snimanje slike na računaru i na ovaj način smo dobili *nekompresovanu* sliku.

Glava 5

Vizuelno-kriptografski algoritmi monohromatskih slika

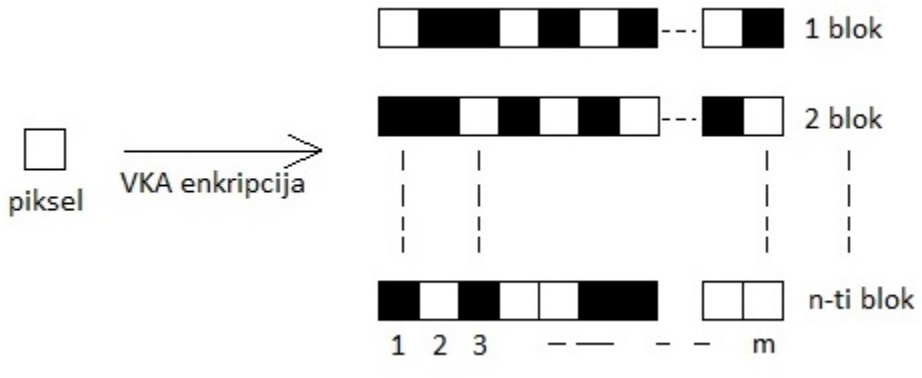
Ulazni podaci koje kriptujemo koristeći vizuelno-kriptografske algoritme VKA dati su u obliku slike, odnosno formata koji možemo vizuelno interpretirati. Zbog ove osobine, za podatak koji želimo da kriptujemo sa VKA, odnosno učinimo tajnim, koristićemo izraz *slika*. Opisani VKA rade sa monohromatskim slikama, odnosno binarnim slikama. Ovakvo ograničenje dato kao ograničenje ulaznog podatka samog algoritma, a sa određenim transformacijama slike u boji možemo pretvoriti u binarne slike i suštinski iskoristiti iste algoritme i za kriptovanje slika u boji.

U VKA SS algoritmu, sliku nećemo dijeliti na djelove, koji bi zadržali jasno čitljive podatke o početnoj slici, odnosno originalni raspored piksela. Ključ u VKA SS predstavlja *slajd* (eng. transparencie) i pri tome su svi slajdovi istog prioriteta. Svaki piksel sa slike je enkriptovan i u tom procesu su dobijeni *blokovi* od m piksela (podpiksela).

Blokova za kodiranje jednog piksela imamo koliko i slajdova, tj. VKA učesnika u SS dijeljenju. Ako je originalni piksel bijeli blok ne mora isključivo sadržati bijele podpiksele, a isto važi i za crni blok.

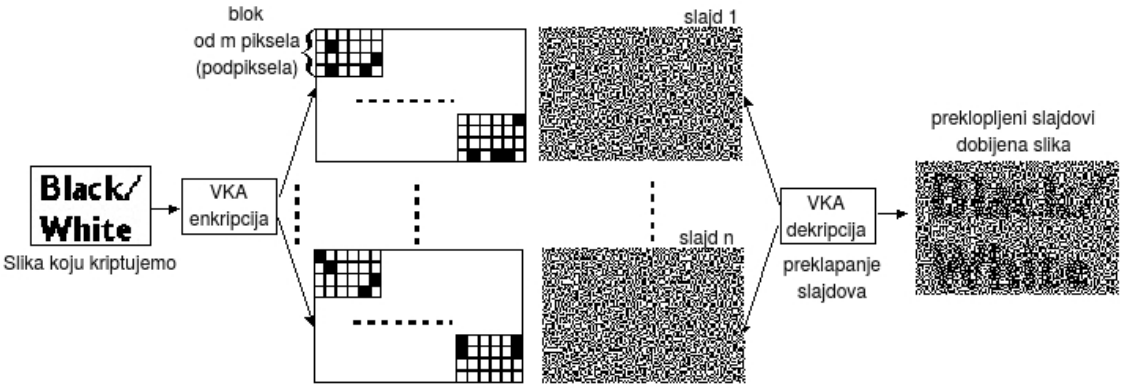
Svi VKA predlažu štampanje slajdova na providne folije, pri čemu se mora voditi računa da svi slajdovi štampaju u istoj rezoluciji i sa istim CMYK podešavanjima

intenziteta crne boje. Zbog ove specifičnosti mnogo prikladniji izraz za bijele piksele bio bi providni, jer se oni ne štampaju.



Slika 5.1: Vizuelno kriptografski algoritam, enkripcija

Dekriptovanje u VKA se realizuje *preklapanjem slojeva* i na taj način generišamo početnu sliku. Proces dekriptovanja realizovan na ovaj način ne zahtijeva korišćenje kompjutera, pa je samim tim pogodan i u situacijama gdje oni nijesu dostupni.



Slika 5.2: Vizuelno-kriptografski algoritam, enkripcija i dekripcija

5.1 Naor i Šamirov VKA

Moni Naor i Adi Šamir 1994. godine [2] definisali su uslove koje treba da zadovoljava jedan siguran vizuelno-kriptografski algoritam dijeljenja tajne k od n koji radi sa binarnim slikama, kao i napravili njegovu konstrukciju za neke vrijednosti k.

Na svakom od n slajdova jedan piksel sa slike zamijenjen je blokom koji sadrži m piksela (podpiksela). Ovako dobijenu strukturu za kriptovanje jednog piksela predstavljamo Bulovom matricom S veličine $n \times m$, pri čemu je $S = [s_{ij}]$, a vrijednost $s_{ij} = 1$ akko je j-ti piksel u i-tom slajdu crn. Ljudsko oko ne može savršeno da razlikuje piksele u slici visoke rezolucije, pa se dešava da se crni i bijeli piksel jedan pored drugog vide kao jedan sivi. Nivo sive boje jednog bloka poslije dekripcije predstavlja Hamingovu težinu vektora redova V dobijenog primjenom operacije 'OR', nad blokovima.

$$S = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & \dots & a_{km} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}; V_S = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ \vee & \vee & \vee & \vee \\ \vdots & \vdots & \ddots & \vdots \\ \vee & \vee & \vee & \vee \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & \dots & h_m \end{bmatrix}$$

Ukoliko je jedan od elementa na poziciji S_{ij} crn rezultujući piksel je crn. Zbog ove osobine operacije OR moramo da uvodimo prag d i parametar α da bismo napravili očiglednu razliku između bijelih i crnih piksela. Prag $1 \leq d \leq m$ i relativna razlika $\alpha > 0$ omogućavaju očiglednu razliku između bijelih i crnih piksela, odnosno vještački ćemo bijele blokove praviti posvjetljivanjem sive boje $H(V) < d - \alpha m$, a crne blokove pokušavati da što više potamnimo $H(V) \geq d$.

Definicija 3.

Šema dijeljenja tajne slike *k od n* sadrži dva skupa C_0 i C_1 Bulovih matrica $m \times n$. Da kriptujemo bijeli piksel nasumično biramo jednu od matrica iz C_0, dok za crni piksel biramo jednu od matrica iz C_1 istom metodom.

$$C_0 = \{W_1, W_2, \ldots, W_r\} \; ; \; C_1 = \{B_1, B_2, \ldots, B_r\}$$

Izabrana matrica određuje blokove za enkripciju jednog piksela u svakom od n slajdova. Način odabira matrica za skupove C_0 i C_1 ispravan je ako:

1. Za svaku maticu W_i iz C_0, operacija OR bilo kojih *k od n* redova zadovoljava $H(V) \leq d - \alpha m$.

2. Za svaku maticu B_i iz C_1, operacija OR bilo kojih *k od n* redova zadovoljava $H(V) \geq d$.

3. Formiramo nove skupove matrica $C_0^{'}$ i $C_1^{'}$ iz skupova C_0 i C_1.

   Opisujemo postupak formiranja, na primjeru skupa $C_0^{'}$. Neka je W matrica iz C_0. Iz ove matrice formira se niz novih, veličine $q \times n$, pri čemu je $1 \leq q \leq k-1$, tako što se odaberu proizvoljnih q vrsta posmatrane matrice W. Taj postupak obavimo sa svakom matricom iz C_0 i skup svih novodobijenih matrica će činiti skup $C_0^{'}$. Na identičan način formiramo i skup $C_1^{'}$.

   $$C_0^{'} = \left\{W_1^{'}, W_2^{'}, \ldots, W_r^{'}\right\} \; ; \; C_1^{'} = \left\{B_1^{'}, B_2^{'}, \ldots, B_r^{'}\right\}$$

   Uslov tri je sadržan u zahtjevu da mora biti $C_0^{'} = C_1^{'}$.

Iz trećeg uslova, koji se zove i princip sigurnosti VKA, slijedi da analizom $k-1$ blokova ne možemo utvrditi da li je piksel koji će se pojaviti na dekriptovanoj slici bijeli ili crni. Takođe, konstruisan VKA uz poštovanje sva tri uslova definicije je

šema dijeljenja tajne *k od n*, jer se sa $k-1$ slajdom ne može otkriti koja je slika enkriptovana, dok nam k slajdova omogućava njenu dekripciju.

Za ovako postavljenu šemu sljedeći parametri su bitni:

1. m - širenje jednog originalnog piksela na podpiksele u svakom bloku. On predstavlja gubitak u rezoluciji prilikom kriptovanja između slike koju kriptujemo i dekriptovane slike. Za mali gubitak kvaliteta između enkriptovane i dekriptovane slike m treba da bude što manji.

2. α - parametar relativne razlike crnih i bijelih piksela. Ovo je parametar gubitka kontrasta slike. Za visoku čitljivost slike α parametar treba da bude što veći.

3. r - veličina kolekcija C_0 i C_1. Generalno, veličina kolekcija C_0 i C_1 ne mora da bude ista.

5.1.1 Konstrukcija Naor-Šamirovog VKA 2 od 2

Na najnižem nivou, problem vizuelne kriptografije možemo predstaviti kao specijalni slučaj dijeljenja tajne *2 od 2*. Vizuelno-kriptografski model dijeljenja tajne *2 od 2* dijeli podatak na dva slajda, $n = 2$, i za dekriptovanje su nam potrebna oba slajda $k = 2$. Zavisno od vrijednosti originalnog piksela (crno ili bijelo) biraćemo iz predloženih skupova C_0 ili C_1.

$$C_0 = \left\{ \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} \right\}, \left\{ \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \right\}$$
$$C_1 = \left\{ \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \right\}, \left\{ \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} \right\}$$

Ovakvim izborom matrica, Hamingovo rastojanje potvrđuje čitljivost, a vjerovatnoća P uniformnu raspodjelu i poštovanje principa sigurnosti VKA, pa je ovakav odabir matrica ispravan.

Piksel na slici	$P(C_t)$	Blok slajda 1	Blok slajda 2	Posle dekripcije	V(H)
□	0.5_0	□■	□■	□■	1
□	0.5_0	■□	■□	■□	1
■	0.5_1	□■	■□	■■	2
■	0.5_1	■□	□■	■■	2

Međutim, pri samom izboru matrica, mijenjali smo 1 originalni piksel za 2 podpiksela po blokovima, pa ćemo imati širenje dekriptovane slike. Da bismo sačuvali originalnu proporciju slike poslije dekripcije, za kriptovanje jednog piksela koristićemo blok $m = 4$, pa su predložene sljedeće matrice 2×2:

$$M = \left\{ \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \right\}, \left\{ \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \right\}, \left\{ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right\}$$

Elemente skupova C_0 i C_1 dobijamo tako što uzimamo po jedan element iz M i smještamo ga u svaki od skupova, a zatim za njegov par koji će se nalaziti na drugom slajdu za C_0 uzimamo isti element ili za C_1 permutaciju po kolonama istog elementa. Sljedeća tabela predstavlja odabir prvog para iz M za kriptovanje slike.

Piksel na slici	$P(C_t)$	Blok slajda 1	Blok slajda 2	Posle dekripcije	V(H)
□	0.5_0	□ ■	□ ■	□ ■	2
□	0.5_0	■ □	■ □	■ □	2
■	0.5_0	□ ■	■ □	■ ■	4
■	0.5_0	■ □	□ ■	■ ■	4

Ovakvi skupovi su takođe čitljivi, $H(V)_{C_0}$=2 $H(V)_{C_1}$=4, regularni i prihvatljiviji, a manje utiču na redukciju kvaliteta slike, konkretno joj čuvaju proporcije.

U originalnoj šemi predložen je način generisanja matrica koje rješavaju VKA 2 od n.

$$M_0 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \dots & 0 \end{bmatrix}; M_1 = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix};$$

C_0 i C_1 su sve matrice dobijene permutacijom kolona matrica M_0 i M_1. Uzimajući bilo koje blokove iz dva slajda za dekripciju slike, dobijamo $H(V)_{C_0} = 1$, $H(V)_{C_1} = 2$, zadovoljavajući čitljivost, a mnogo veću čitljivost dobijamo dodavanjem još slajdova, jer će se crni piksel sastojati od mnogo više crnih podpiksela u bloku.

5.1.2 Kompleksnost Naor-Šamirovog algoritma 2 od 2

Da bi izvršili kriptovanje naše tajne koristeći algoritam svaki piksel se mora pročitati, a zatim blok od m piksela se mora upisati na svakom od slajdova. Pošto kriptujemo podatke na dva slajda ukupan broj piksela koji na svakom od slajdova treba da bude upisan je $2m$. Ukupan broj piksela koji upisujemo na svim slajdovima je $2(n \times m)$. Sve dok je $m < n$, algoritam ima linearnu kompleksnost $O(n)$ za kriptovanje. Dekriptovanje koje se vrši ručno, preklapanjem slajdova, ima kompleksnost $O(1)$. Ako preklapanje radimo kompjuterski, tj. ako ne štampamo slajdove, već programiramo sabiranje matrica slajdova enkripcije, potrebno nam je $O(m)$ za dekripciju. Pod uslovom da je m < n, i dekripcija ima linearnu kompleksnost.

5.1.3 Analiza Naor-Šamirovog algoritma 2 od 2

Uvedimo oznake, B za bijeli piksel, a C za piksel crne boje. Smanjićemo skupove C_0 i C_1, pa ćemo originalni piksel kriptovati sa samo dva različita bloka $BCCB$ i $CBBC$. Pretpostavimo da smo kriptovali tajnu i podijelili slajdove Alisi i Bobu. Takođe, pretpostavimo da je na neki način Eva uspjela da presretne komunikaciju i da dođe do jednog od podijeljenih slajdova. Da bi uspješno izvršila dekripciju tajne, ona treba da uzima blok po blok sa slajda i da pokušava da dođe do informacije o tome koje boje je originalni piksel. Ako je naišla na blok $BCCB$, sa vjerovatnoćom $P(B) = 0.5$ i $P(C) = 0.5$ bira boju originalnog piksela. Na isti način, ako naiđe na blok $CBBC$ sa istim vjerovatnoćama bira boje. Zbog dobrog izbora blokova za kodiranje Eva ne može da dođe do preciznije informacije o originalnim pikselima, osim da nagađa. Pošto je slika sastavljena od $n = w \times h$ piksela, vjerovatnoća da će svaki piksel pogoditi je 0.5^n. Slika standardne rezolucije 1280×800 sadrži 1024000 piksela, pa je vjerovatnoća da će Eva pogoditi šta se krije iza kriptovanog slajda $1.93{*}10^{-308255}$. Treba da naglasimo da je najbolje što Eva može da uradi Brute force napad, s tim što neće imati kriterijum za zaustavljanje jer će ona u stvari generisati mnogo slika i neće imati bliže informacije o tome koja je slika original. Teorijski algoritam izgleda savršen, dok je praktično vrlo teško napraviti algoritam za koji možemo tvrditi da je apsolutno siguran. Ranjivost algoritma leži u funkciji gdje vršimo odabir bloka za enkripciju, tj. funkciji koja će nam nasumično birati sa istom vjerovatnoćom svaku od matrica iz C_0 i C_1, odnosno blok kojim kodiramo određeni piksel.

Korektnost rutine za dekripciju leži u korektnosti binarne operacije OR nad Bulovom matricom i mogućnosti ljudskog oka da razlikuje sivo od crne. Svaki bijeli piksel kriptujemo sa $CBBC$ ili $BCCB$ blokovima na oba slajda. Crne piksele kriptujemo sa istim blokovima, s tim što ako je na prvom slajdu crni piksel kodiran blokom $CBBC$,

isti piksel na drugom slajdu mora biti kodiran blokom $BCCB$ da bi preklapanjem dobili $CCCC$. Isto važi i za kodiranje crnog piksela kombinacijom $BCCB$ na prvom i $CBBC$ na drugom slajdu. Kad se preklope dva identična bloka, rezultirajući blok se neće promijeniti, već će biti ispunjen sa samo dva crna piksela, tj. 50% od ukupnog broja piksela sa kojima kodiramo. Ljudsko oko će ovu kombinaciju detektovati kao sivu. Na isti način će 100% crnih piksela u bloku detektovati kao crno. Na ovaj način će ukupna slika poslije dekripcije biti crno-siva, umjesto crno-bijele u originalu.

Tehnike napada koje napadačima na VK šeme stoje na raspolaganju poslije dolaska u posjed određenog dijela slajdova je podmetanje, tj. napadač može iskoristiti slajdove koje je presreo i zaključiti npr. na kojim se pozicijama mogu naći crni pikseli. Sljedeći korak bi bio generisanje novog slajda koji dijeli dio podataka sa presretnutim, a ostatak polja bio bi popunjen na taj način da generiše novu tajnu. Napadač zatim novodobijenu kombinaciju, tj. 'novu sliku', proglašava originalom i na taj način vara ostale učesnike. Između ostalih tehnika koje služe za povećavanje otpornosti na podmetanje, treba pomenuti algoritme digitalnog potpisivanja (eng. Digital Watermark) koje se mogu primijeniti na slajdovima i na taj način potvrditi da su svi slajdovi generisani iz iste slike.

5.2 Kafri-Keren VKA slučajne mreže 2 od 2

Nasuprot Naor-Šamirovom algoritmu koji piksele pretvara u blokove i tako utiče na povećanje dimenzija originalne slike, Kafri i Keren predložili su algoritam koji dimenzije ostavlja originalnim [3]. Ovaj algoritam poznat je kao algoritam slučajne mreže *eng. random grid*. Algoritam počinje sa radom tako što prvo čita dimenzije $h \times w$ originalne slike I i inicijalizuje dva slajda S_1 i S_2, matrice, istih dimenzija kao original. S_1 popunjavamo nasumično crnim i bijelim pikselima, a S_2 ostavljamo

praznu. Paralelno prolazeći kroz originalnu sliku i nasumično generisan slajd čitamo piksel po piksel i generišemo sadržaj drugog slajda na sljedeći način: ako su pikseli na poziciji [x,y] identični, generisaćemo bijeli piksel, u suprotnom generišemo crni.

$I[x,y]$	$S_1[x,y]$	$P(S_1[x,y])$	$S_2[x,y]$	$S_1[x,y] + S_2[x,y]$
□	□	0.5	□	□
□	■	0.5	■	■
■	□	0.5	■	■
■	■	0.5	□	■

U gornjoj tabeli prikazane su sve kombinacije piksela koje možemo generisati na slajdovima u zavisnosti od vrijednosti piksela na ulazu. Kreiranje slajda na ovaj način odgovara operaciji XOR. Ako na slici imamo isti broj crnih i bijelih piksela, tačnu vrijednost za bijele piksele poslije njene enkripcije i dekripcije koristeći Kefri-Keren VKA imaćemo samo u 50% slučajeva, što implicira promjenu u odnosu na original i gubitak tačnosti od 25%. Ova nepravilnost se dešava zbog razlike u rezultatu kada se na isti sadržaj primjeni XOR i OR operacija. Ovu promjenu ne možemo korektno ispraviti postprocesima.

5.2.1 Kompleksnost VKA Kafri-Keren 2 od 2

Pretpostavimo da je $n = w \times h$ broj piksela na originalnoj slici. Da bismo izvršili algoritam prvo moramo da generišemo nasumičnu matricu jednog slajda tako što ćemo svakom elementu niza dužine n dodijeliti vrijednost iz skupa $\{0, 1\}$. Pošto se dodijeljene vrijednosti predstavljaju sa bitom dužine 1, za generisanje S_1 potrebno nam je linearno vrijeme $O(n)$. Da bismo generisali S_2 svaki piksel sa S i S_1 mora

biti pročitan, pa nam je i za ovo potrebno linearno vrijeme $O(n)$. Pri izvršavanju dekripcije također nam je potrebno linearno vrijeme, pa možemo zaključiti da algoritam troši linearno vrijeme za izvršavanje.

5.2.2 Kafri-Keren VKA 2 od 2 pseudokod

Algorithm 1: Kafri Karen VKA 2 od 2

Ulaz: Slika koja se kriptuje sirina $\times$ *visina*
Izlaz: Dva slajda S_1 *i* S_2 *visina* $\times$ *sirina*
• *Setovanje konstanti* S_2
1: $WHITE \leftarrow 0$
2: $BLACK \leftarrow 1$

• *Kreiranje slajda* S_1
3: **for** $row \leftarrow 1$ **to** *visina* **do**
4: **for** $col \leftarrow 1$ **to** *sirina* **do**
5: $S_1[row, col] = RANDOM(WHITE, BLACK)$
6: **end for**
7: **end for**
• *Kreiranje slajda* S_2
8: **for** $row \leftarrow 1$ **to** *visina* **do**
9: **for** $col \leftarrow 1$ **to** *sirina* **do**
10: **if** $I[row, col] = WHITE$ **then**
11: $S_2[row, col] = S_1[row, col]$
12: **else**
13: $S_2[row, col] = 1 - S_1[row, col]$
14: **end if**
15: **end for**
16: **end for**
17: **return** S_1, S_2

5.2.3 Analiza VKA Kafri-Keren 2 od 2

Uvedimo oznake, B za bijeli piksel, a C za piksel crne boje. Pretpostavimo da smo kriptovali tajnu i podijelili slajdove Alisi i Bobu. Takođe, pretpostavimo da je na neki način Eva uspjela da presretne komunikaciju i da dođe do slajda koji nije nasumično generisan. Da bi uspješno izvršila dekripciju tajne, ona treba da uzima piksel po piksel sa slajda i da pokušava da dođe do informacije o tome koje boje je originalni piksel. Ako je naišla na crni piksel sa vjerovatnoćom $P(I[x,y] = C|S_2[x,y] = C) = 0.5$ bira boju crnu kao boju originalnog piksela. Sa istom vjerovatnoćom $P(I[x,y] = B|S_2[x,y] = B) = 0.5$ pretpostavlja da je originalni piksel bijeli. Eva ne može da dođe do preciznije informacije o originalnim pikselima, pa zaključujemo da će sa vjerovatnoćom $(\frac{1}{2})^n$ Eva uspješno pogoditi tajnu. Nema smisla analizirati situaciju u kojoj Eva dolazi u posjed slajda koji je slučajno generisan.

5.3 Čen-Tsao VKA slučajne mreže n od n

Čen i Tsao su predložili proširenje Kafri-Keren algoritma *2 od 2* za situaciju dijeljenja tajne *n od n* za bilo koju vrijednost n [4]. Ovakva ideja predloženog proširenja može se koristiti i za druge algoritme. Konkretno mogli bismo ga iskoristiti i za proširenje Naor Shamir algoritma *2 od 2* do *n od n* s tim što bismo morali uvesti međukorak koji vrši redukciju veličina generisanih slajdova.

Algoritam radi tako što počinje sa kreiranjem slajdova S_1 i S_1' iz slike I, pri čemu su S_1 i S_1' generisani korišćenjem Kafri-Kerenovog VKA 2 od 2. U svakom sljedećem koraku koristi se slajd koji nije slučajno generisan kao nova tajna koju treba kriptovati. *N-ti* slajd u algoritmu ne generišemo, nego mu dodjeljujemo vrijednost tajne proglašene u *n - 1* koraku. Dekripcija tajne se vrši preklapanjem S_i slajdova.

Dati su koraci generisanja svakog od slajdova u primjeru vizuelnog dijeljenja tajne *3 od 3*.

Korak	Slajd	Dodijeljena vrijedost
1	S_1	Random
2	S_2'	S_1 XOR I
3	S_2	Random
4	S_3'	S_2 XOR S_2'
5	S_3	S_3'

Pošto algoritam slučajnih rešetki utiče na kvalitet slike smanjujući njenu originalnost, algoritam *n od n* sa povećanjem broja učesnika dijeljenja tajne znatno utiče na degradiranje kvaliteta. Za $n \geq 5$ može se analizirati teorijski, dok se u praksi rijetko primjenjuje za kriptovanje važnih informacija, jer bitno utiče na kvalitet slike dobijene dekripcijom.

5.3.1 Čen-Tsao VKA n od n pseudokod

Ovaj program kao funkciju koristi *Kafri Keren VKA 2 od 2*, koji smo već dali.

Algorithm 2: Chen Tsao VKA n od n

Ulaz: I - slika koja se kriptuje, dimenzija sirina × visina
Ulaz: n - broj slajdova koje treba generisati
Izlaz: S_1, S_2, ... , S_n *- slajdovi, dimenzija visina × sirina*

- *Generisanje početnih slajdova*

```
1: S_1, S_1' ← KK2 − od − 2(I)
2: while n > 2 do
3:    for i ← 2 to n − 1 do
4:       S_i, S_{i+1}' ← KK2 − od − 2(S_i')
5:    end for
6:    S_n ← S_n'
7: end while
8: return S_1, ..., S_n
```

5.4 VKA binarnih slika za rad sa slikama u boji

Iako se tipovi VKA koje smo objasnili bave isključivo radom sa monohromatskim slikama, naše algoritme možemo iskoristiti i za rad sa slikama u boji. Prije nego sliku enkriptujemo uradićemo par koraka predprocesiranja. Pod raslojavanjem se podrazumijeva rastavljanje slike na više slika, tako da svaki sloj sadrži određeni dio nepromijenjenih informacija o slici iz koje je nastao. Spajanjem slojeva, tj. njihovim preklapanjem, dobijamo originalnu sliku bez gubitka informacija. Sliku u RGB kodnom modelu raslojavamo po osnovnim kanalima. Na ovaj način dobićemo tri slike, a svaka od njih sadržaće samo piksele svog kanala. U zavisnosti od dubine boja na početnoj slici 2^n ,$n = R_{bita} + G_{bita} + B_{bita}$ radićemo konačan broj raslojavanja za svaki kanal posebno i konačan produkt ovih operacija biće n slika. Po svakom od kanala dobićemo onoliko binarnih slika koliko je bilo bita potrebno za čuvanje osnovne boje u pikselu na originalnoj slici.

(a) RGB 24-bita

(b) R kanal 8-bita

(c) G kanal 8-bita

(d) B kanal 8-bita

(e) RGB 8-bita iskorišćeno 1+1+1-bit

(f) R kanal 1-bit

(g) G kanal 1-bit

(h) B kanal 1-bit

Slika 5.3: Transformacija RGB - Grayscale - Monochrome - RGB

Pristup da se slike u RGB kodnom modelu mogu raslojiti na binarne slike, omogućava da takve binarne slike iskoristimo kao ulazni podatak za VKA. Na slici 5.3

dat je primjer jedne takve transformacije slike, pri čemu nisu generisani svi potrebni slojevi, već samo dio.

Sliku 5.3(a) raslojili smo na 3 RGB kanala. Korak gdje je svaki od kanala raslojen posebno je izostavljen, pa je uzet samo po jedan sloj iz svakog od R G i B slojeva i dobijene su binarne slike 5.3(f, g i h). Ovo binarne slike su spojene i dobila se RGB slika od 8 bita, pri čemu je samo 3 bita realno iskorišćeno.

Ovakav princip je moguće primjenjivati kad osim enkripcije i dekripciju radimo na računaru. U slučaju da radimo dekripciju originalnom metodom, štampa pa preklapanje slojeva, proces iziskuje previše slajdova. Za sliku u boji od 24-bita i primjenu Naor Šamirovog VKA 2 od 2 na binarne slike dobijene iz te slike, bilo bi potrebno 48 slajdova, što proces čini previše složenim.

Glava 6

Zaključak

Naor-Šamirov algoritam je u teorijskom smislu siguran vizuelno kriptografski algoritam. Nažalost, njegova specifičnost da bez dodatnih predprocesiranja ulaznog podatka može raditi samo sa binarnom slikom, donekle mu ograničava primjenu. U koracima enkripcije povećava originalne dimenzije slike, pa nam je potrebno više memorije za čuvanje slajdova i više vremena za njihovu obradu. Takođe, za vraćanje slike u prvobitne dimenzije i popravljanje izgubljenih piksela moramo koristiti postprocesne algoritme koji povećavaju vrijeme izvršavanja. Kafri-Kerenov algoritam slučajne rešetke je takođe siguran algoritam, ali za razliku od Naor-Šamirovog algoritma ne vrši povećanje dimenzije slika, a samim tim ne utiče na povećanje potrebne memorije za njihovo čuvanje. Nasuprot Naor-Šamirovom algoritmu, manu Kafri-Kerenovog algoritma nalazimo u tome što se postprocesima dekriptovanoj slici ne može vratiti originalni raspored piksela, pa ga možemo okarakterisati kao brz ali manje pouzdan . Problemi koje treba rješavati da bi se prevazišle postojeće mane objašnjenih VKA jesu pronalasci načina da se prilikom procesa dekriptovanja operacija OR zamijeni sa operacijom XOR.

Bibliografija

[1] J. S. Lim, Two-Dimensional Signal and Image Processing, Prentice Hall, 1990.

[2] Modi Naor i Adi Shamir, Visual Cryptography, Springer-Verlag, 1998.

[3] O. Kafri and E. Keren, Encryption of pictures and shapes by random grids, Optics Letters, 1987.

[4] Tzung-Her Chen and Kai-Hsiang Tsao, Visual secret sharing by random grids, Pattern Recognition, 2009

[5] Koriolan Gilezan i Boško Latinović, Bulova algebra i primene, Matematički institut, 1977

[6] Jonathan Weir and WeiQi Yan, Visual Cryptography and Its Applications Ventus Publishing, 2012

[7] Adi Shamir, How to Share a Secret, Programming Techniques MIT, 1979