UNIVERZITET U NOVOM SADU

FAKULTET TEHNIČKIH NAUKA

Trostruki DES kriptografski algoritam

Projektni zadatak iz

Sigurnost i bezbednost elektroenergetskog softvera

Školska 2016. / 2017. godina

Profesor

Docent dr Lendak Imre

Studenti

Vladislav Simić

Zoltan Babinski

1. Teorijske osnove

Simetrični kriptografski algoritmi se mogu podeliti u dve kategorije:

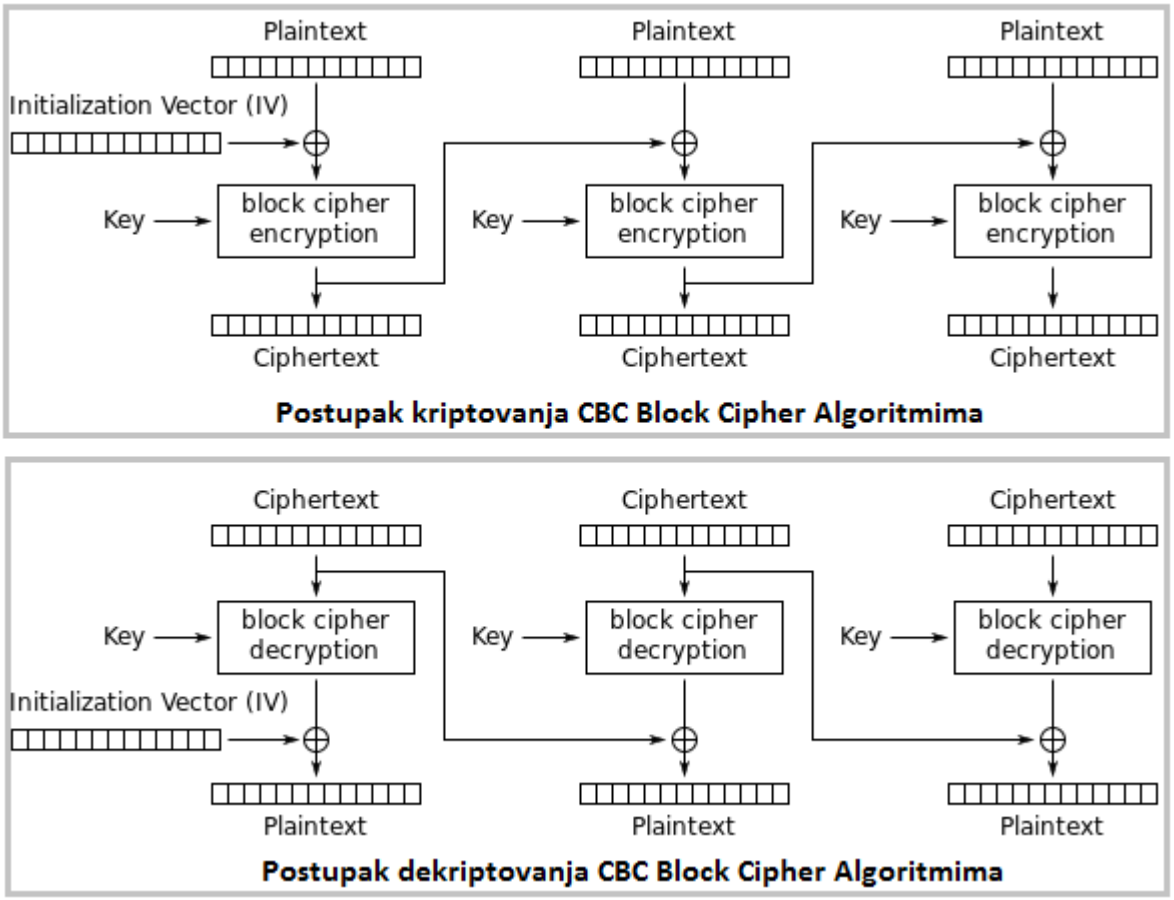
- Stream cipher algoritmi – koji rade nad bitovima podataka
- Block cipher algoritmi – koji rade nad blokovima podataka fiksne dužine (npr. DES, 3DES,AES algoritmi). Poruka se deli na n-bitne blokove, a ukoliko je poslednji blok manji od n, dopunjava se do n bita. Nad svakim blokom se primenjuje algoritam koji predstavlja kombinovane operacije zamene i permutacije bitova. Algoritam se primenjuje u iterativnim postupcima, odnosno rundama. Rezultat je takođe blok podataka iste dužine kao i ulazni blok. Dva tipična moda block cipher algoritama su Electronic Codebook (ECB) mod i Cipher Block Chaining (CBC) mod.

Electronic Codebook (ECB) mod je najjednostavnija metoda block cipher algoritama. Svaki blok se kriptuje potpuno nezavisno od ostalih blokova, a isti ključ se koristi za svaki blok. Upravo ove osobine čine ECB metodu nesigurnom, jer isti blokovi podataka imaju iste šifrovane vrednosti.

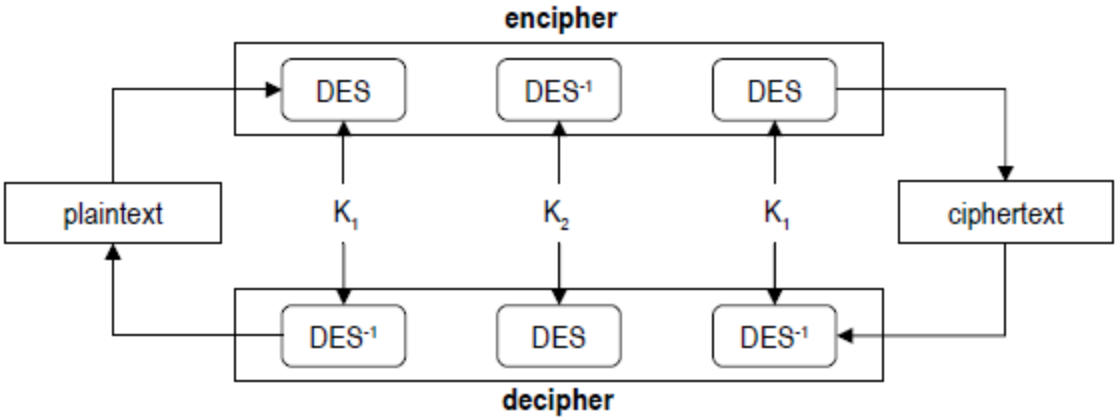
Cipher Block Chaining (CBC) mod definisan je kako bi prevazišao problem identičnih poruka ECB moda. Mod se zasniva na randomizaciji ulazne poruke (Slika 1.) tako da isti blok podataka neće dati istu šifrovanu poruku. Uvodi se inicijalni vektor (IV) tako da se nad prvim blokom podataka pre kriptovanja primenjuje XOR operacija sa vrednošću IV. Random vrednost kojom se XOR-uje svaki sledeći blok biće izlaz iz prethodnog bloka. Inicijalni vektor je random vrednost koja ne mora da se čuva u tajnosti, bitno je samo da bude nepredvidiva i da se svaki put koristi drugačija vrednost. Da bi poruka bila dekriptovana, potrebno je da strana koja dekriptuje, osim algoritma i tajnog ključa, zna i IV vrednost.

DES algoritam je block cipher algoritam, odnosno algoritam zasnovan na kriptovanju podataka po blokovima fiksne dužine. Otvoreni tekst se deli u blokove od 64 bita i DES algoritam se primenjuje korišćenjem 64-bitnog ključa (od kojih se efektivno koristi 56 bita, dok se nižih 8 koristi za proveru pariteta). Block cipher algoritmi se zasnivaju na primeni iterativnih postupaka zamene i permutacije. DES algoritam se sastoji iz 16 Feistel rundi, a rezultat kriptovanja je šifrat iste dužine kao plaintext. Suštinska slabost ovog algoritma je pre svega dužina ključa. U periodu kada je DES algoritam objavljen, 56 bita je bila dovoljna dužina da ključ ne može biti otkriven brute-force napadom.

Triple-DES (trostruki DES) je poboljšana verzija DES algoritma koja podrazumeva kriptovanje podataka DES algoritmom u tri prolaza sa dva različita ključa od 64 bita (Slika 2). Uvođenjem dodatnog 64-bitnog ključa dobija se ukupna dužina ključa 128 bita (odnosno 112 bita efektivno), čime je prevaziđen problem otkrivanja ključa.



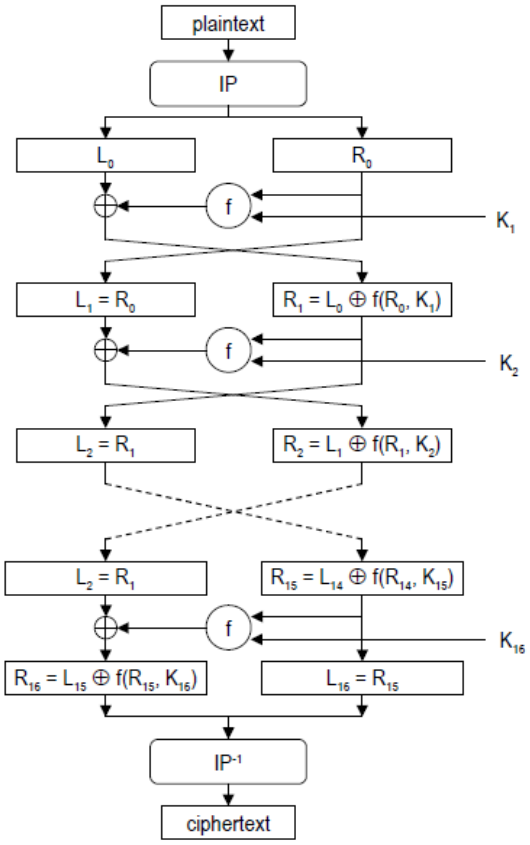
Slika 1. Cipher Block Chaining (CBC) mod



Slika 2. Triple DES Algoritam

2. Dizajn sistema

DES algoritam predstavljen na slici 3 se sastoji od 16 Feistel rundi u kojima se vrši kombinovana zamena i permutacija bitova.



Slika 3. DES algoritam

1. Pre izvršavanja Feistel rundi obavlja se inicijalna permutacija bitova nad blokom podataka. Permutacija se vrši na osnovu matrice inicijalnih permutacija čije vrednost predstavljaju pozicije bita nakon permutacije.
2. Nakon inicijalne permutacije vrši se podela bloka na levu (32 bita) i desnu (32 bita) stranu.
3. Za svaku rundu generišu se podključevi, čiji postupak generisanja je opisan u odeljku 2.1.
4. Mozak svake Feistel runde predstavlja tzv. Feistel funkcija (odeljak 2.2). Izlaz Feistel funkcije se XOR-uje sa levim (32-bitnim) blokom podataka i on postaje desni blok podataka koji se koristi kao ulaz za Feistel funkciju u narednoj rundi. U poslednjoj rundi ne vrši se zamena desne i leve strane bita.
5. Nakon završetka 16 rundi, leva i desna strana blokova podataka se spajaju.

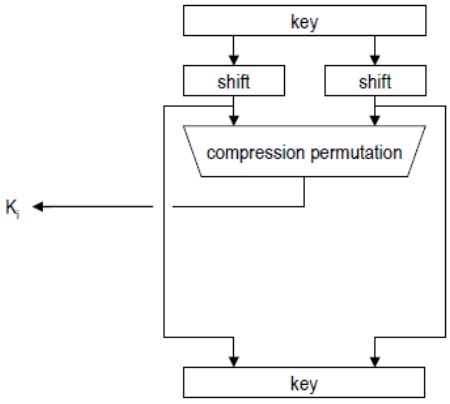
6. Vrši se inverzna permutacija bita na osnovu inverzne vrednosti tabele inicijalnih permutacija.

Isti algoritam se koristi i za dekriptovanje, a suštinska razlika je u generisanju podključeva za svaku rundu.

2.1. Generisanje podključeva

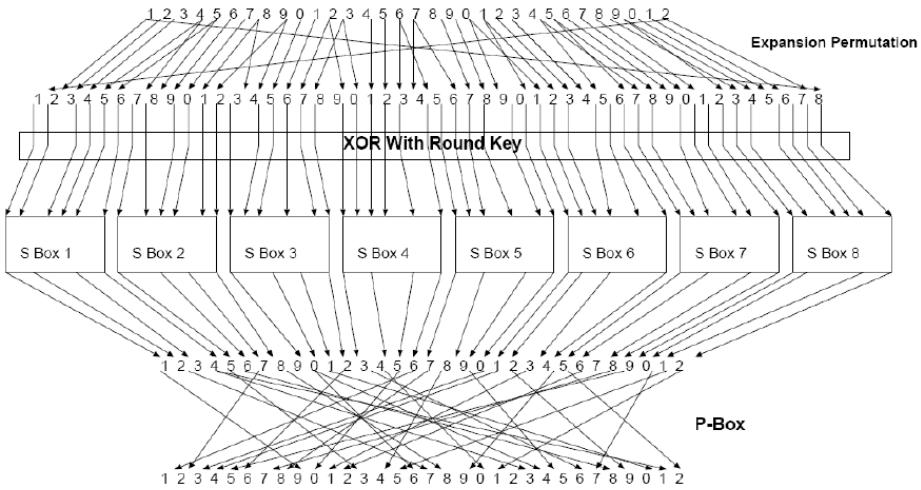
64-bitni ključ se redukuje u 56-bitni ignorisanjem svakog osmog bita. Dalje se od 56-bitnog ključa generiše 48-bitni ključ za svaku rundu.

1. ključ se podeli u dve 28-bitne polovine
2. polovine se cirkularno pomeraju u levo za 1 ili 2 mesta, zavisno od kruga
3. izdvaja se 48 bita pomoću sažimajuće permutacije, koja je predstavljena u permutacionoj matrici
4. ključ za naredni krug čine spojene polovine 56-bitnog ključa



2.2. Feistel funkcija

Postupak permutacije i zamene bitova u Feistel funkciji je prikazan na slici ispod. Na ulazu Feistel funkcije su 32 bita podataka i 48 bita podključa. Na početku izvršavanja 32 bita podataka se proširuju na 48 bita, pomoću odgovarajuće matrice permutacija. Izlaz permutovanja se XOR-uje sa podključem za datu rundu. Nakon XOR-ovanja, 48 bita se prosleđuje na 8 Sbox-ova gde u svaki ulazi 6 bita, a na izlazu se dobijaju 4 bita. Svaki Sbox je predstavljen matricom, čije ćelije predstavljaju izlaznu vrednost SBox-a u binarnoj vrednosti. Izlazi SBox-a se spajaju, a zatim permutuju prema permutacionoj tablici i takva permutovana vrednost predstavlja izlaz Feistel funkcije.



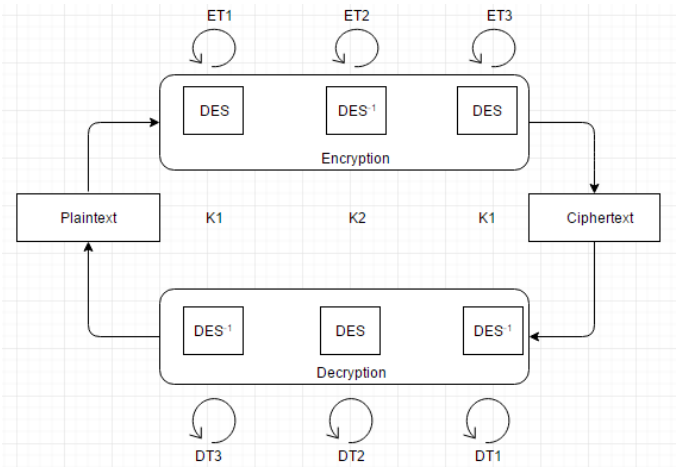
3. Korišćene strukture podataka

U toku realizacije algoritma uglavnom je korišćen **niz bitova** (*BitArray*). Algoritam je razvijan u .NET okruženju, a u njemu je realizovana klasa koja omogućava čuvanje niza bitova i operacije nad istim bitovima. Obzirom da se ceo algoritam bazira nad zamenama i permutacijama nad bitovima podataka, pokazalo se da je najpogodnije korišćenje upravo ove strukture podataka za čuvanje i obradu bitova.

4. Paralelizacija sistema

CBC mod unosi lančanu zavisnost između blokova poruka tako što izlaz iz svakog prethodnog bloka utiče na sledeći blok, odnosno da bi se mogao izvršavati DES algoritam na sledećem bloku, potrebno je da izvrši DES algoritam na prethodnom bloku. Ovo svakako utiče na brzinu obrade i performanse samog sistema.

Paralelizacija u ovom rešenju realizovana je kreiranjem thread-ova za svaki DES algoritam (i za enkripciju i za dekripciju, Slika 3). Tokom enkripcije prvi se aktivira thread ET1 koji vrši enkripciju. Thread ET2 čeka signaliziranje semafora da je završena obrada na thread-u ET1, i nakon toga kreće njegova obrada. Thread ET3 čeka da se završi obrada na ET2, odnosno da se izvrši dekripcija nad blokom podataka. Analogno ovome, isti logika je iskorištena i kod dekripcije. Postupak enkripcije tokom iteracija je prezentovan u tabeli.



Slika 3. Paralelizacija sistema

Rbr. iteracije	blok #1	blok #2	blok #3	blok #4	...
1.	DES				
2.	DES^{-1}	DES			
3.	DES	DES^{-1}	DES		
4.		DES	DES^{-1}	DES	
5.			DES	DES^{-1}	
6.				DES	
7.					
8.					
...					

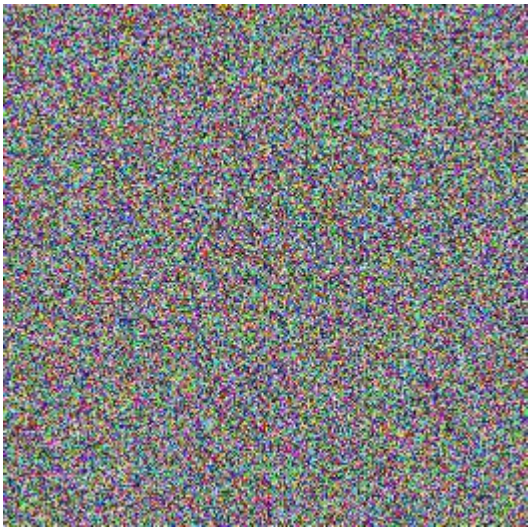
5. Testiranje rešenja

Testiranje i verifikovanje rešenja algoritma izvršeno je enkriptovanjem i dekriptovanjem:

- fotografije .jpg formata i dimenzija 262 x 261 px

1. Izgled fotografije pre enkriptovanja

2. Izgled enkriptovane fotografije



3. Izgled fotografije nakon dekripcije

- teksta zapisanog u .txt fajlu veličine 222 KB

1. Tekst pre enkriptovanja

THE WELL OF ETERNITY
ONE
The tall, forbidding palace perched atop the very edge of the mountainous cliff, overlooking so precariously the vast, black body of water below that it appeared almost ready to plummet into the latter’s dark depths. When first the vast, walled edifice had been constructed, using magic that melded both stone and forest into a single, cohesive form, it had been a wonder to touch the heart of any who saw it. Its towers were trees strengthened by rock, with jutting spires and high, open windows. The walls were volcanic stone raised up, then bound tightly by draping vines and giant roots. The main palace at the center had originally been created by the mystical binding of more than a hundred giant, ancient trees. Bent in together, they had formed the skeleton of the rounded center, over which the stone and vines had been set.
A wonder to touch the hearts of all when first it had been built, now it touched the fears of some. An unsettling aura enshrouded it, one heightened this stormy night. The few who peered at the ancient edifice now quickly averted their gaze.
Those who looked instead to the waters below it found no peace, either. The ebony lake was now in violent, unnatural turmoil. Churning waves as high as the palace rose and fell in the distance, crashing with a roar. Lightning played over its vast body, lightning gold, crimson, or the green of decay. Thunder rumbled like a thousand dragons and those who lived around its shores huddled close, uncertain as to what sort of storm might be unleashed.

2. Tekst nakon enkriptovanja

3. Tekst nakon dekriptovanja

THE WELL OF ETERNITY

ONE

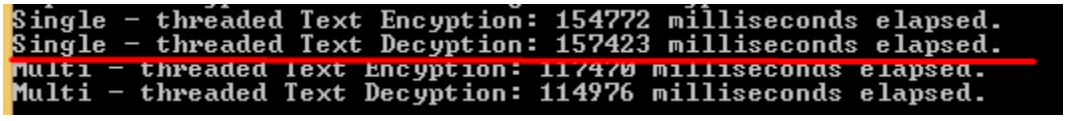
The tall, forbidding palace perched atop the very edge of the mountainous cliff, overlooking so precariously the vast, black body of water below that it appeared almost ready to plummet into the latter's dark depths. When first the vast, walled edifice had been constructed, using magic that melded both stone and forest into a single, cohesive form, it had been a wonder to touch the heart of any who saw it. Its towers were trees strengthened by rock, with jutting spires and high, open windows. The walls were volcanic stone raised up, then bound tightly by draping vines and giant roots. The main palace at the center had originally been created by the mystical binding of more than a hundred giant, ancient trees. Bent in together, they had formed the skeleton of the rounded center, over which the stone and vines had been set.

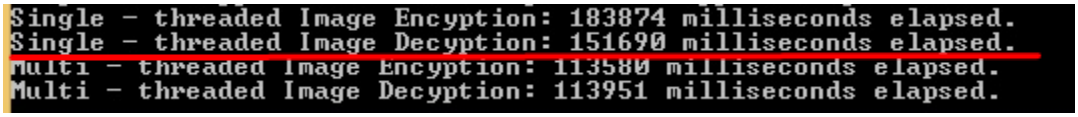
A wonder to touch the hearts of all when first it had been built, now it touched the fears of some. An unsettling aura enshrouded it, one heightened this stormy night. The few who peered at the ancient edifice now quickly averted their gaze.

Those who looked instead to the waters below it found no peace, either. The ebony lake was now in violent, unnatural turmoil. Churning waves as high as the palace rose and fell in the distance, crashing with a roar. Lightning played over its vast body, lightning gold, crimson, or the green of decay. Thunder rumbled like a thousand dragons and those who lived around its shores huddled close, uncertain as to what sort of storm might be unleashed.

5.1. Upoređivanje vremena izvršavanja

Na slikama ispod prikazana je razlika u vremenu izvršavanja u single-threaded režimu i vremena nakon paralelizacije obrade podataka. Očekivano vreme nakon paralelizacije bi trebalo biti 3 puta manje od vremena u single-threaded režimu, ali lančana zavisnost između obrade podataka utiče na vreme izvršavanja kako je pomenuto ranije u tekstu.





5.2. Opterećenje procesora

Za praćenje opterećenja procesora iskorišten je programski paket Performance Monitor. Na slici ispod prikazan je dijagram zauzeća procesora. Na mestu označenom strelicom pokrenuta je obrada u multi-threaded režimu pa se uočava veća iskorišćenost jezgara procesora, pošto je u tom trenutku pokrenuto više thread-ova. Prosečno zauzeće procesora iznosi 86%.