Enkripcija podataka na računalu

Davidović, Ivan

VELEUČILIŠTE SUVREMENIH INFORMACIJSKIH TEHNOLOGIJA

STRUČNI PRIJEDIPLOMSKI STUDIJ INFORMACIJSKIH TEHNOLOGIJA

Ivan Davidović

ZAVRŠNI RAD

ENKRIPCIJA PODATAKA NA RAČUNALU

Zagreb, listopada 2024.

Veleučilište suvremenih informacijskih tehnologija
10000 Zagreb, Ulica Vjekoslava Klaića 7

Studij:	Stručni prijediplomski studij informacijskih tehnologija smjer programiranje
Student:	**Ivan Davidović**
Matični broj:	2021053

Zadatak završnog rada

Predmet:	Primijenjena i numerička matematika
Naslov:	**Enkripcija podataka na računalu**
Zadatak:	Definirati i objasniti osnovne enkripcijske pojmove. Opisati simetrične i asimetrične algoritme i neke metode zažtite podataka u ovisnosti o vrsti mreže. Na primjeru, prikazati postupak kriptiranja i dekriptiranja primjenom raspoloživog softvera.
Mentor:	Marijan Čančarević, v. pred.
Zadatak uručen kandidatu:	1.7.2024.
Rok za predaju rada:	31.10.2024.
Rad predan:	_____

Povjerenstvo:

Dragana Čulina, pred.	član predsjednik	_____
Marijan Čančarević, v. pred.	mentor	_____
Edmond Krusha, v. pred.	član	_____

SADRŽAJ 3

POPIS SLIKA

POPIS KODOVA

1. UVOD

Digitalizacija podataka i informacija stvara problem zaštite podataka na računalima. Povjerljivi podatci, informacije i dokumenti u digitalnom obliku moraju biti zaštićeni. Ako neovlaštene osobe dobiju pristup informacijama kojima nisu smjele imati pristup to može biti veliki problem. Da bi se takve stvari izbjegle i kako bi takvi podatci bili zaštićeni koristi se enkripcija podataka.

U ovom dokumentu objasnit će se što je to enkripcija podataka, kakav utjecaj ima na digitalni svijet te bitnost i potrebu enkripcije digitaliziranih dokumenata i informacija. Navest će se neki algoritmi koji se koriste za enkripciju podataka, njihove važne karakteristike i zašto su oni bitni. Također, bit će istaknuto kako se uspješno zaštititi od različitih opasnosti koje vrebaju na internetu, bilo da se radi o podacima koji se primaju ili šalju preko interneta.

Opisane su situacije i postupci u slučaju da nepozvane osobe pokušaju pristupiti kriptiranim podacima.

Proći će se kroz neke alate za enkripciju koji se mogu koristiti za kriptiranje podataka i koje načine enkripcije koriste kako bi zaštitili podatke. Napomenut će se i jedna od zaštita na internetu koja se često koristi i kako štiti podatke. Nakon toga obrazložit će se sigurnosne mjere koje su napravljene da spriječe neovlaštene osobe od pristupa podatcima i što se događa ako nema takvih mjera da zaštite podatke te što ako netko neovlašten uspije dobiti pristup kriptiranim podatcima. Idealan sustav za enkripciju podataka i sve potrebno da takav sustav uspije.

Bit će prikazan postupak korištenja alata za enkripciju pod nazivom BitLocker. Način pronalaska BitLocker-a na Windows-ima, izbori koji se prikazuju tijekom korištenja alata za enkripciju i prikaz u slučaju da se kriptira driver preko kojeg se pokreću Windowsi.

Praktični dio prikazuje pojednostavljen kod koji objašnjava kako se podatci kriptiraju. Kod je isto tako prikaz simetričnog kripto sustava za razmjenu kriptiranih poruka. Prolaskom kroz kod pobliže je objašnjeno na koji način se ključ za kriptiranje stvara i korištenje ključa kako se kriptirana poruka može dekriptirati.

Na kraju može se vidjeti postupak korištenja koda da se poruke kriptiraju i dekriptiraju. Problem u slučaju da se unese krivi broj ili znamenka koja nije broj riješen je na jednostavan način koji osigurava da korisnik može unijeti samo ispravan broj za odabir kriptiranja, dekriptiranja podataka i izlazak iz petlje pa pri tome i izlazak iz programa.

Opisano je kako bi se kod mogao modificirati da ključ za kriptiranje i dekriptiranje postane konstantan.

Unutar zaključka će se ukratko objasniti kako kvantna računala opasnost za današnju enkripciju i kako će se moći obraniti od dekriptiranja pomoću kvantnih računala.

2. ENKRIPCIJA PODATAKA

2.1. Kriptiranje

Kako bi se smanjila šansa da osjetljive informacije budu pročitane od strane neovlaštenih osoba, koristi se enkripcija podataka koja osigurava da je to nemoguće napraviti.

Razlog zašto nije moguće pročitati kriptirane podatke je taj što enkripcija pretvara jasan i otvoren tekst u nerazumljiv tekst osobama koje nemaju ključ kako bi dekriptirali podatke. Tako se zaštićuju osjetljivi podatci i sprječava curenje informacija u slučaju ako se izgubi prijenosno računalo, USB ili ako se dogodi zlonamjeran napad na neko računalo koje posjeduje takve informacije ili zlonamjerni fizički pristup neovlaštene osobe računalu koje posjeduje osjetljive podatke. Današnji operativni sustavi većinom dolaze s ugrađenim mehanizmom kriptiranja, jednostavan i lagan za korištenje, stoga je lakše zaštititi podatke.

Postoje dvije vrste enkripcije podataka koje se mogu koristiti, a to je simetrična i asimetrična enkripcija. Simetričnom kripto sustavu ključ koji se koristi za kriptiranje podataka u nerazumljiv tekst je isti kao i ključ za dekriptiranje podataka u razumljiv tekst, dok u asimetričnom kripto sustavu nije takav slučaj. Kod simetričnog kripto sustava pošiljatelj prvo mora razmijeniti ključ za kriptiranje, odnosno dekriptiranje s osobom koja prima kriptiranu poruku, stoga je ključ potrebno nekako poslati primatelju ili se osobno mora napraviti razmjena ključa enkripcije, odnosno dekripcije. Takav ključ naziva se tajni ključ jer samo primatelj i pošiljatelj imaju ključ i koriste ga za enkripciju i dekripciju. Simetrični kripto sustav, ako se implementira dobro, može biti jednako siguran ili sigurniji od asimetričnog kripto sustava. Problem kod simetričnog sustava je kako prenijeti ključ drugoj strani kako bi se uspostavila sigurna komunikacija. Protokol koji se najčešće koristi kako bi se razmijenio tajni ključ u simetričnom kripto sustavu naziva se Diffie-Hellmanov protokol.

Kako bi se u asimetričnom kripto sustavu razmijenile osjetljive poruke ili datoteke koriste se privatni i javni ključevi. Primatelj stvara svoj par ključeva od kojih se jedan naziva javni, a drugi privatni ključ. Javni ključ je dostupan javnosti i osobe koje šalju poruku koriste taj ključ kako bi kriptirali poruke koje žele poslati primatelju koji onda može dekriptirati te poruke s pomoću svog privatnog ključa. Primatelj ima svoj privatni ključ koji zadržava samo za sebe kako bi mogao samo on dekriptirati poruke koje je zaprimio. Ovako se uspostavlja kriptirana komunikacija gdje bilo tko može pristupiti javnom ključu i kriptirati poruke pomoću tog ključa te poslati kriptiranu poruku, ali samo primatelj koji ima svoj privatni ključ može dekriptirati poruku koja je poslana. Primjeri asimetričnog kripto sustava koji se najčešće

koristi je RSA, čiji su autori Ron Rivest, Adi Shimir i Len Adleman. Primjer takvih algoritama su ELGamal, NTRUEncrypt, LUC i drugi.

2.2. Algoritmi

Algoritmi kriptiranja koriste se za kriptiranje ili šifrirane, a jedina razlika je korištenje algoritama. Koliko će neki podatci biti zaštićeni, ovisi o tome koji algoritam se koristi za kriptiranje tih podatka. Generalno što više bitova algoritam koristi za kriptiranje to će podatci biti sigurniji. Ako algoritam koristi manji broj bitova dolazi do velike mogućnosti dekriptiranja podatka silom što znači da neovlaštena osoba koristi manji broj ključeva za dekriptiranje podataka. Zato je preporučeno koristiti algoritme koji koriste veći broj bitova za enkripciju da neovlaštena osoba ne može dekriptirati podatke silom koristeći relativno mali broj ključeva u malom vremenu. Algoritmi koji koriste veći broj bitova je gotovo nemoguće dekriptirati silom radi velikog broja ključeva kroz koji se mora proći. U praksi je moguće silom naći ključ, ali zbog velike količine bitova i drukčijih matematičkih metoda koje algoritmi koriste vrijeme dobivanja ključa silom trajalo bi milijarde godina ili bi zahtijevalo mnogo sreće.

Primjeri algoritama za kriptiranje podataka:

- **AES (Advanced Encryption Standard)** – simetrični algoritam korišten kao zamjena za DES (Data Encryption Standard) koji koristi isti ključ za kriptiranje i dekriptiranje podataka. Algoritam može imati ključ veličine 128. 192 ili 256 bitova.
- **Triple DES (Triple Dana Encryption Algorithm)** – simetričan algoritam koji primjenjuje DES algoritam tri puta na svaki blok podataka što znači da ima efektivno 112 do 168 bitova. Ovaj algoritam je već zamijenjen s boljim AES algoritmom radi sigurnosti i lakšeg korištenja.

Ovo su samo dva primjera od mnogih algoritama kojim se mogu koristiti za kriptiranje podataka.

2.3. Alati za enkripciju

Postoji mnogo enkripcijskih alata koji se mogu koristiti za kriptiranje zato je bitno znati koji alat za enkripciju koristiti kako bi se podatci najbolje zaštitili. Radi velike količine alata za enkripciju podataka mora se paziti koji alat se preuzima kako bi se podatci kriptirali jer neki od tih enkripcijskih alata mogu biti virusi ili trojanski virusi koji izgledaju i rade kao alati za enkripciju, no zapravo kradu podatke i dokumente, zato je bitno znati koji su alati sigurni za

preuzimanje. Alati poput VeraCrypt, koji je napravljen nakon TrueCrypt-a, može koristiti za enkripciju podataka. Jedan od alata pod nazivom AxCrypt koji se isto može koristiti za kriptiranje podataka omogućava nakon enkripcije podatka da se taj podatak prepravi na način da u trenu kada ovlašteni korisnik pristupi podatku preko tog alata podatak dekriptira, što dopušta korisniku da prepravi grešku te se podatak ponovno kriptira. Najsigurniji alat koji se može koristiti naziva se BitLocker i taj alat za enkripciju dolazi s Windowsima. Ovaj alat koristi AES algoritam enkripcije.

Alati za enkripciju koji su ovdje nabrojani su samo neki koji se mogu koristiti za enkripciju podataka, a od kojih je BitLocker najsigurniji jer dolazi zajedno s Windowsima.

2.4. Zaštita preko javnih mreža

Svi podatci koji se šalju preko interneta mogu biti presretani hakerima ako se šalju preko javne WI-FI konekcije. Kriptirani podatci su sigurni jer se neće moći pročitati, ali bolje je izbjeći situacije u kojima hakeri imaju priliku ukrasti takve podatke. Najbolja opcija je ne koristiti javne WI-FI konekcije, a u slučaju korištenja javne konekcije potrebno je provjeriti je li WI-FI konekciju sigurno koristiti. Hakeri mogu napraviti svoje konekcije kroz koje mogu presretati podatke koji se šalju preko takvih konekcija. Ne kriptirane podatke haker može koristiti na zlonamjerne načine, no kriptirani podatci su hakerima beskorisni dok se ne dekriptiraju. Postoji još jedan način kako zaustaviti hakere da ne mogu presresti podatke neovisno jesu li kriptirani ili ne. Način na koji možemo to postići je korištenje Virtual Private Network (VPN). Koristeći VPN osiguravamo da je hakerima iznimno teško presresti korisničke podatke, a enkripcija osigurava da, ako su presretani, podatci ostanu nečitljivi i nemoguće dekriptirati, stoga su hakeru i beskorisni.

2.5. Virtualna privatna mreža

Virtual Private Network ili skraćeno VPN koriti se kao zaštita na internetu kako podatci ne bi dospjeli u krive ruke. Ova tehnologija osigurava sigurno povezivanje računala na javne i djelovne mrežne infrastrukture. VPN osigurava sigurni medij preko kojeg se može uspostaviti veza privatnih mreža preko javne mreže. To se postiže na način kada se podatci pošalju s računala. Oni izlaze iz privatne mreže i prolaze kroz gateway uređaj, ulaze i prolaze kroz javnu mrežu te izlaze iz javne mreže i prolaze kroz gateway uređaj na drugu stranu koji štiti drugo računalo. VPN osigurava sigurno slanje tih podataka između dvije udaljene privatne mreže tako da automatski kriptira podatke prilikom slanja na drugu stranu, inkapsulira u IP

pakete i kad podatci stignu na drugu stranu kanala komunikacije, paketi se automatski dekriptiraju. Podatci koji se šalju preko VPN-a sigurni su jer se koristi kompleksna enkripcija paketa tijekom slanja pošiljatelja do ciljnog korisnika koji prima kriptirane pakete. Prema ovom može se zaključiti da je enkripcija ili šifriranje podataka iznimno bitna kod VPN-a kako bi se podatci sigurno slali preko javnih mreža. Tehnika koju VPN koristi kako bi sigurno prenijela podatke s jednog računala do drugog naziva se tuneliranje i funkcionira na način da VPN prilikom povezivanja otvori sigurni tunel koji omogućuje kriptiranje i enkapsulaciju podataka i autentikaciju korisnika.

2.6. Dodatna zaštita

Enkripcija nekad nije dovoljna kako bi podatci bili potpuno zaštićeni. Pristup neovlaštene osobe kriptiranoj datoteci i njeno mijenjanje bez traga je veliki problem. Samo kriptiranje podataka nekad nije dovoljno kako bi ti podatci bili i ostali sigurni. Radi bolje sigurnosti da kriptirane podatke nisu promijenile neovlaštene osobe potrebno je koristiti detekciju neovlaštenih upada (eng. Intrusion detection system - IDS) kako bi se osiguralo da se, u slučaju ako dođe do neovlaštene promjene, vidi koje su se aktivnosti dogodile i koja je šteta počinjena. IDS ne može ništa učiniti u slučaju da se dogodi neovlaštena aktivnost na računalu, ali može obavijestiti korisnika računala da se nešto promijenilo ili da se događa napad. Poželjno je da IDS funkcionira u stvarnom vremenu radi brže reakcije na neovlašteni upad na računalo. IDS je veoma poželjan alat pošto većina napada na računalo dolazi s internih mreža računala ili zaposlenih organizacija. Ako nije moguće zaustaviti napad IDS zapisuje logove aktivnosti koje se mogu pregledati da se ustanovi neovlašteni korisnik ili počinjena šteta i koraci koji su potrebni da bi se takvi budući napadi onemogućili.

2.7. Idealan sustav enkripcije

Pošto je enkripcija podataka najbitniji dio zaštite tih podataka potrebno je znati što sve sustav za enkripciju mora imati kako bi imali što idealniji sustav enkripcije. Takav sustav bi morao biti jednostavan za korištenje korisniku, kvalitetan radi zaštite podataka i efikasan radi brže enkripcije na način koji ne utječe na efikasnost računala koje kriptira podatke.

Kako bi takav sustav mogao raditi mora zadovoljavati sljedeće uvjete:

- **Zaštita sadržaja datoteka –** datoteke moraju biti ne samo kriptirane kako neovlašteni korisnici ne bi mogli pristupiti datoteci bez odgovarajućeg ključa, nego ih se ne smije moći dekriptirati bez ključa za dekripciju. Isto tako sustav mora biti napravljen na način da kad su

predstavljene dvije iste kriptirane datoteke mora biti nemoguće ustanoviti da su te dvije datoteke iste. Sustav mora isto tako onemogućiti identifikaciju istih sekvenci koji se pojavljuju unutar nekriptirane datoteke.

- **Mrežna zaštita** – tijekom slanja podataka preko mreže moguće je prisluškivanje mrežnog prometa neovlaštenih korisnika ili napadača. Poželjno je da ako napadač uspije presresti podatak bude nemoguće dekriptirati, čime se osigurava nemogućnost čitanja osjetljivih podataka.
- **Transparentne performanse** – sustav za enkripciju podataka ne smije narušavati normalan rad korisnika na računalu. Pošto sustav za enkripciju ima neizbježno manju brzinu čitanja i pisanja od uobičajenih datotečnih sustava.
- **Transparentni pristup datotekama** – datoteke koje su kriptirane ne smiju se razlikovati od datoteka koje su pohranjene na disku nakon što ovlašteni korisnik pristupi datotekama s ispravnim ključem. Pristup datotekama mora biti transparentna aplikacija koje koristi krajnji korisnik.
- **Jednostavno upravljanje ključevima** – sustav mora biti izveden na način da pristup datotekama bude ostvaren preko ispravnog ključa i da sustav ne traži unos ključa svaki put kada se koristi operacija čitanja i pisanja, nego da je potrebno samo jednom unijeti ključ i da se ključ smatra unesenim i pouzdanim tijekom cijelog trajanja čitanja i pisanja.
- **Portabilnost** – sustav za enkripciju datoteka mora biti izveden na način da kad se kriptirana datoteka pošalje na drugi sustav koji ima implementiranu enkripciju podatkovnog sustava koji omogućuje korištenje odgovarajućeg ključa za dekripciju i omogućuje otvaranje kriptirane datoteke uz pomoć istog ključa.
- **Kompatibilnost s računalnim dijelovima** – sustav za kriptiranje datoteka ne smije utjecati na rad drugih komponenti istog računala u bilo kojem smislu.
- **Zaštita meta podataka** – sustav enkripcije mora zaštititi meta podatke osjetljivih kriptiranih datoteka radi prisluškivanja i krađe podataka tako da napadač ne zna o kojem se kriptiranom podatku radi, nego samo da je to neki podatak koji je potencijalno osjetljiv i da je kriptiran.

3. NAČIN ENKRIPCIJE PODATAKA

U ovom djelu pokazan je način brze i lagane enkripcije drivera na računalu. Pojednostavljen način kriptiranja podataka na driveru je prikazan pomoću pojednostavljenog koda. Kod služi kao primjer načina enkripcije podataka i kao primjer simetričnog kripto sustava.

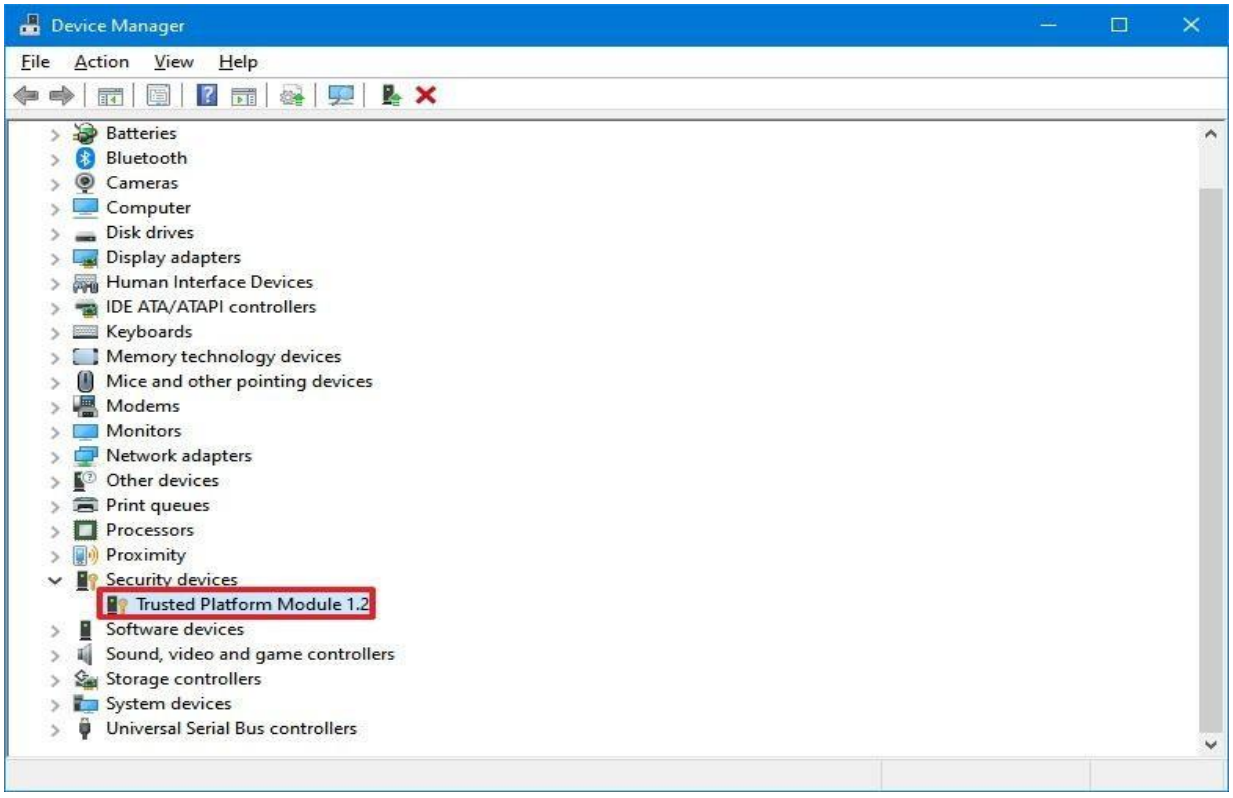
3.1. Korištenje alata za kriptiranje podataka

Zaštita podataka je bitna stvar, stoga je važno znati kako kriptirati podatke na računalu. Primjer kriptiranja podataka ovdje je prikazan preko alata koji se dobije uz Windowse.

Naziv alata koji se ovdje prikazuje je BitLocker koji je razvijen od strane Microsofta i koristi algoritam kriptiranja AES s 128 ili 256-bitnim ključem. Primjer prikazuje korištenje BitLocker-a na Windows 10. Nakon prolaza korištenja alata ispod se nalazi pojednostavljen kod koji će pobliže prikazati kako se podatci kriptiraju.

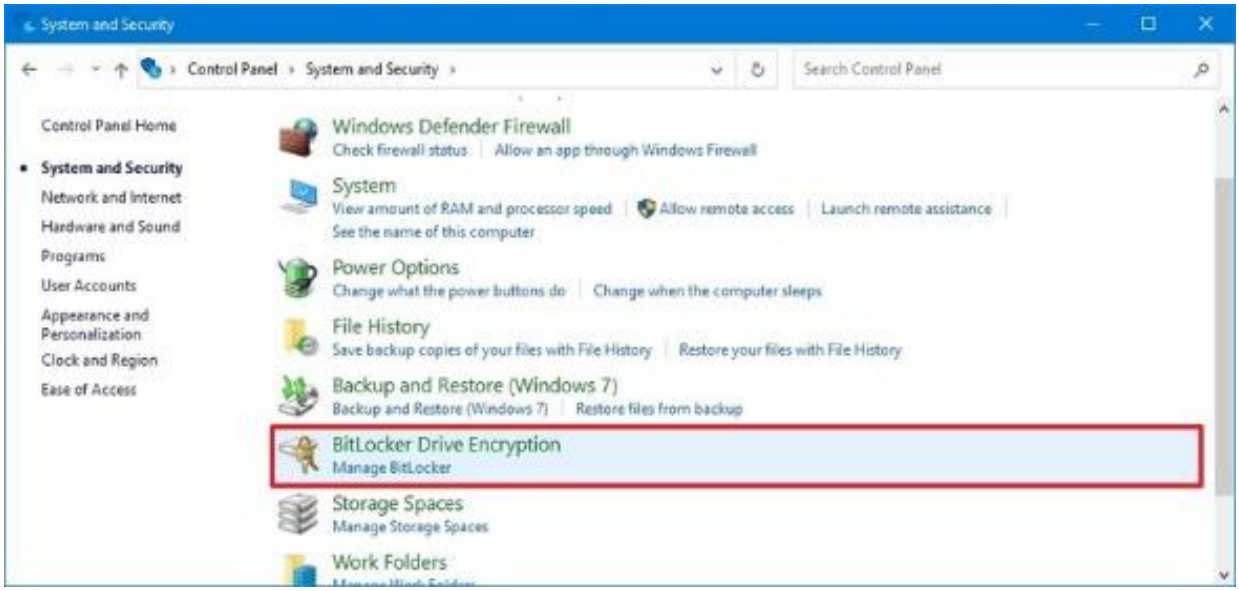
Prije početka, važno je provjeriti ima li računalo *Trusted Platform Module* (TPM) jer se u suprotnom neće moći koristiti alat za kriptiranje BitLocker.

Za provjeru je potrebno na računalu otići na *Device Manager* i tamo pronaći *Security devices*. Nakon što se utvrdi da računalo ima TPM može se početi s kriptiranjem podataka preko alata BitLocker.



Slika 1. Pronalaženje TPM-a na sustavu Windows (Windows Central, 2022.)

Postupak započinje otvaranjem opcije *System and Security* na računalu koja nudi mnogo opcija koje nisu relevantne za enkripciju podataka. *BitLocker Drive Encryption* je opcija koja je relevantna kako bi kriptirali podatke na računalu.



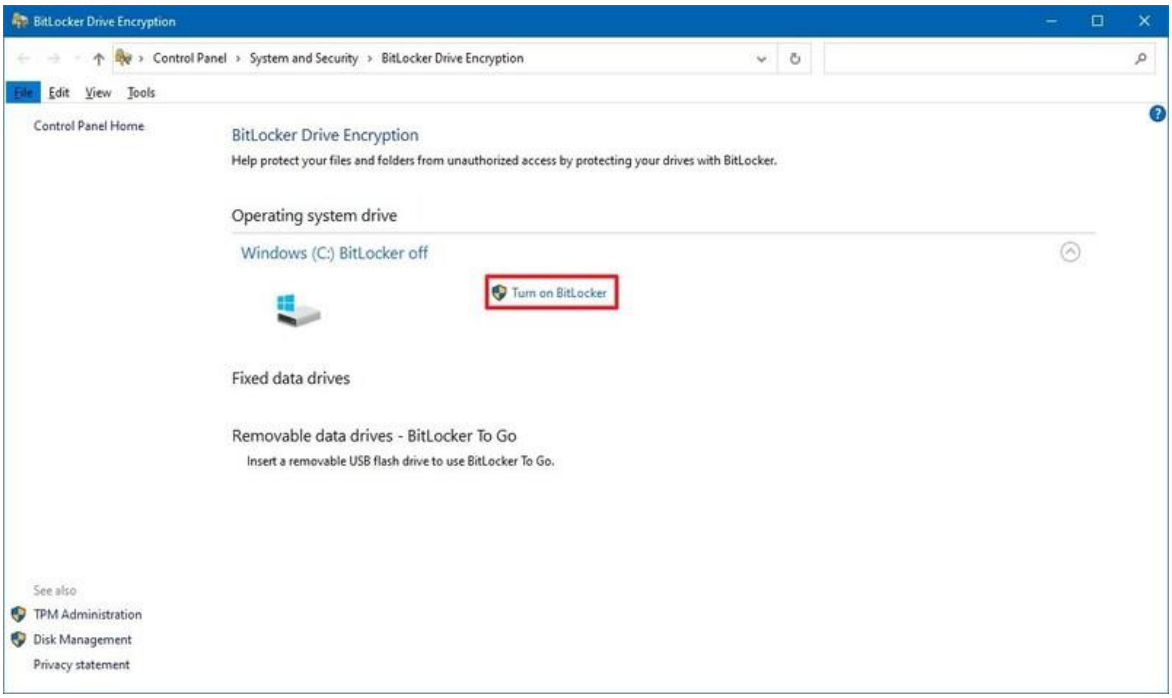
Slika 2. Lociranje alata BitLocker na System and Security (Windows Central, 2022.)

Nakon pronalaska BitLocker-a, potrebno je kliknuti na gumb za otvaranje BitLocker prozora u kojem su vidljivi svi driveri. Postoji i dugi način kako pristupiti BitLocker-u, a to je putem pretrage preko *Search* bara s ikonicom povećala pored Windows ikonice. Potrebno je kliknuti na ikonicu povećala i upisati BitLocker te stisnuti na *Manage BitLocker*.

Slika 3. Pronalazak BitLocker-a u Search baru (Windows Central, 2022.)

Unutar prozora *BitLocker Drive Encryption* moguće je odabrati driver koji želimo. Nakon odabira drivera kojeg želimo kriptirati potrebno je stisnuti gumb na kojem piše *turn on*

BitLocker koji se nalazi odmah pored odabranog drivera zajedno s ikonicom štita što znači da samo administrator računala može napraviti enkripciju drivera.

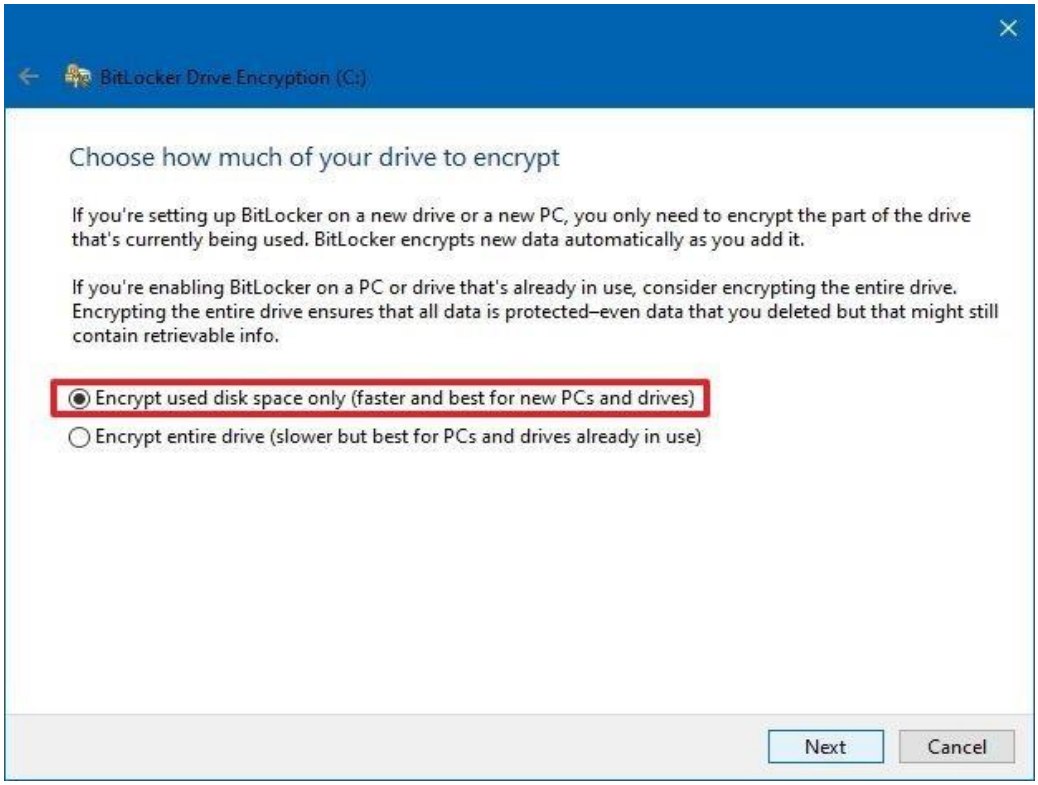


Slika 4. Prikaz lokacije *turn on BitLocker* gumba (Windows Central, 2022.)

Nakon što se pritisne gumb prikazat će se prozor koji nudi dvije opcije od kojih se može odabrati hoće li se postaviti lozinka ili će se koristiti USB kako bi se pristupilo driveru. Ovisno o preferenci korisnika odabire se pristup kriptiranom driveru. Nakon odabira pristupa driveru otvara se novi prozor koji daje tri opcije. Prikazane opcije vezane su za rezervni ključ ili bolje rečeno ključ za oporavak. Ključ se koristi u slučaju da se izgubi USB ili u slučaju da se lozinka za pristup zaboravi ili, ako je negdje zapisana, izgubi. Opcije koje su prikazane pitaju korisnika želi li spremiti ključ za oporavak na *Microsoft account*, u neku datoteku unutar računala ili želi li isprintati ključ za oporavak, ovisno o preferenci korisnika odabere se jedna od tih opcija. Svaka opcija ima svoje prednosti i mane. Istaknuta mana što se tiče svih opcija je kod printanja samog ključa za oporavak. Printajući ključ za oporavak riskiramo da neovlaštene osobe pronađu taj ključ u slučaju da se papir na kojem je ključ izgubi, što je rizik sam po sebi veći. Problem ne nastaje ako se papir s ključem uništi, nego nastaje ako je lozinka za pristup driveru zaboravljena. Ovo je jedna istaknuta mana ove tri opcije koja se lagano može zanemariti ako se papir na kojem je ključ za oporavak drži na sigurnom mjestu.

Nakon odabira mjesta gdje će se ključ za oporavak držati klikne se na gumb *next* koji se nalazi desno pri dnu tog prozora. Nakon pritiska gumba otvorit će se novi prozor koji će upitati korisnika želi li kriptirati dio drivera koji se koristi ili želi kriptirati cijeli driver.

Ovisno o tome koliko se dugo koristi računalo mora se odabrati kako se želi kriptirati driver. Računala koja su nova ili ako je driver novi preporučuje se kriptirati samo dio koji se koristi što će se kriptirati relativno brzo. Za računala koja se koriste duže vrijeme preporučuje se kriptirati cijeli driver što će trajati duže, ali je preporučeno za računala i drivere koji se dugo koriste. Nakon toga pritisne se gumb *next* dolje desno unutar prozora.



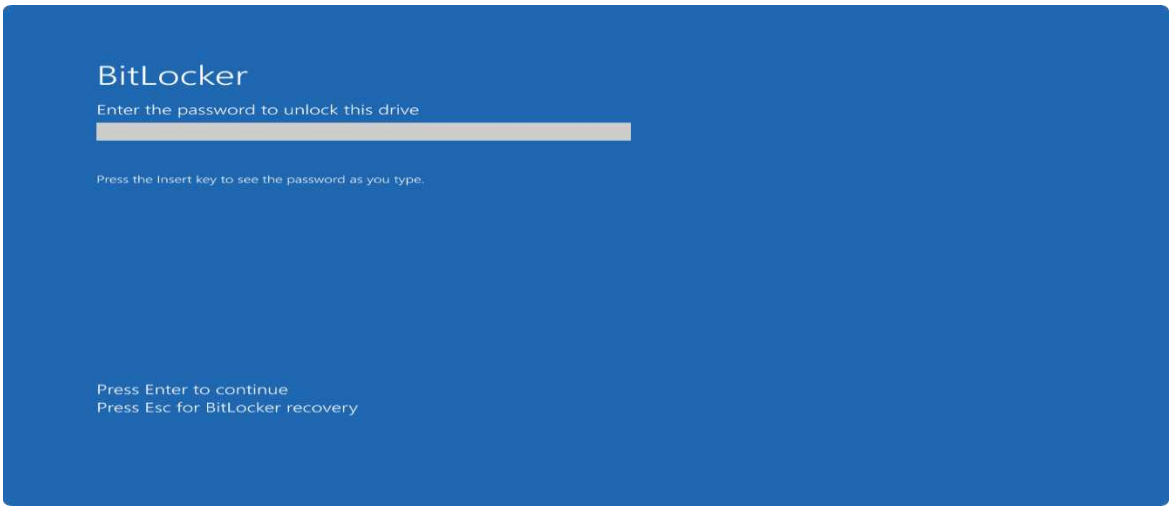
Slika 5. Odabir enkripcije drivera (Windows Central, 2022.)

Sljedeći prozor daje dvije opcije koje su bitne što se tiče kompatibilnosti s novijim i starijim verzijama Windowsa. Isto tako odabir ovisi o tome hoće će li driver biti spojen na jedno računalo ili će se driver odspajati od računala i spajati na druga računala. Dvije opcije koje su prikazane su *New encryption mode* i *Compatible mode*. Prva opcija se odabire ako se driver neće odspajati od računala, ali onda treba imati na umu da će driver jedino biti kompatibilan s verzijama Windowsa na kojima je enkripcija napravljena i na novijim verzijama. Druga opcija se odabire ako će se driver odspajati od računala i spajati na starije verzije Windowsa. Kada se odabere ako se želi da se driver kriptira stisnemo gumb *next* dolje desno unutar prozora.

Slika 6. Odabir kompatibilnosti enkripcije (Windows Central, 2022.)

Zadnji prozor koji se otvori daje do znanja da je sve spremno za enkripciju drivera. Prije nego što se klikne na gumb *next* mora se provjeriti je li kućica s *Run BitLocker system check* označena. Ako nije označena preporučuje se da se označi kako bi se provela provjera sistema kako bi se osiguralo da BitLocker može pročitati ključeve za oporavak i enkripciju. Ovisno o veličini drivera kriptiranje može trajati duže vremena. Kako bi BitLocker počeo s enkripcijom potrebno je restartati računalo.

Pri uključivanju računala ovisno koji driver smo kriptirali može se dogoditi da se mora odmah upisati ključ kako bismo mogli ući u računalo. Ovo se događa ako se kriptira driver koji ima podatke o pokretanju Windowsa.



Slika 7. Izgled unosa lozinke prije ulaska u računalo (Windows Central, 2022.)

4. PRAKTIČNI RAD - POZADINSKI RAD ENKRIPCIJE

4.1. Pojednostavljen kod

Kod koji se koristi kako bi se opisao rad pojednostavljenog kriptiranja jednostavan je i napravljen na način da se ujedno može demonstrirati kako se koristi simetričan ključ za enkripciju. Simetričan ključ se koristi ujedno za enkripciju i dekripciju. Unutar koda postoji normalan niz znakova koji služi za pronalaženje određenog znaka u ključu. Unutar ključa postoji niz znakova koji su nasumično poredani i svaki znak u nizu ključa mora imati svoj znak unutar normalnog niza znakova. Što znači da dužina ključa mora biti jednaka dužini niza normalno poredanih znakova. Na takav način osiguravamo da svaki znak u normalnom nizu ima svoj znak unutar niza ključa.

Kod koji je napravljen u Pythonu i prikazuje jednostavno kriptiranje podataka je sljedeći:

```
import random
import string

znakovi = " " + string.punctuation + string.digits +
string.ascii_letters
znakovi = list(znakovi)
ključ = znakovi.copy()

random.shuffle(ključ)

#print(f"Normalan niz znakova: {znakovi}")
#print(f"Niz ključ: {ključ}")

odabir = int()
```

Kod 1. Prvi dio koda

```
while(True):
    print("Unesite 1 za enkripciju.")
    print("Unesite 2 za dekripciju.")
    print("Unesite 0 za izlaz.")

    try:
        odabir = int(input("Unesite broj: "))
    except ValueError:
        print("Ovdje se unosi broj!")
        continue

    if odabir == 1:
        tekst = input("Unesite tekst za enkriptirati: ")
        kriptirani_tekst = ""

        for znak in tekst:
            index = znakovi.index(znak)
            kriptirani_tekst += ključ[index]

        print(f"Originalan tekst: {tekst}")
        print(f"Kriptirani tekst: {kriptirani_tekst}")

    elif odabir == 2:
        kriptirani_tekst = input("Unesite tekst za dekripciju: ")
        tekst = ""

        for znak in kriptirani_tekst:
            index = ključ.index(znak)
            tekst += znakovi[index]

        print(f"Originalan tekst: {kriptirani_tekst}")
        print(f"Kriptirani tekst: {tekst}")
```

Kod 2. Drugi dio koda

```
    elif odabir == 0:
        break

    else:
        print("Niste unijeli ispravan broj!")
```

Kod 3. treći dio koda

```
znakovi = " " + string.punctuation + string.digits +
string.ascii_letters
znakovi = list(znakovi)
ključ = znakovi.copy()

random.shuffle(ključ)
```

Kod 4. Prikaz početka koda

4.2. **Prolazak kroz kod**

Prije početka kriptiranja stvara se ključ koji se koristi za enkripciju na način da se prvo kopira normalan niz znakova i nasumično ih se posloži u novi niz koji postaje ključ. Preko ključa se može kriptirati i dekriptirati.

Za provjeru mogu se na konzolu isprintati nizovi kako bi se potvrdilo da je zapravo dobiven niz normalnih znakova i niz znakova za ključ. Znak ljestvi znači da je taj dio koda zakomentiran što znači da se tijekom izvođenja koda ne prolazi kroz dio koda označen ljestvama. Ovdje je još definiran odabir i da je taj odabir neki broj. Odabir je u sljedećem dijelu koda bitan jer se mora odabrati hoće li tekst biti kriptiran ili dekriptiran, a koristi se i kako bi se izašlo iz koda.

```
#print(f"Normalan niz znakova: {znakovi}")
#print(f"Niz ključ: {ključ}")

odabir = int()
```

Kod 5. Prikaz komentara ispisa ključa

Do ovog dijela koda sve se obavlja bez potrebe unosa korisnika dok sljedeći dio koda ovisi o tome što korisnik želi učiniti.

```
while(True):
    print("Unesite 1 za enkripciju.")
    print("Unesite 2 za dekripciju.")
    print("Unesite 0 za izlaz.")
    try:
        odabir = int(input("Unesite broj: "))
    except ValueError:
        print("Ovdje se unosi broj!")
        continue
```

Kod 6. Prikaz koda menu

Početak koda je ulaz u beskonačnu petlju koja se obavlja sve dok je uvjet petlje istinit. Ovdje se osigurava da je uvjet uvijek istinit koristeći *true*. Unutar petlje na konzolu se ispisuju tri poruke. Prva je da se stisne broj jedan za kriptiranje, druga je da se stisne dva za dekriptiranje, a treća je da se stisne nula za izlaz iz koda. Pošto postoji šansa da korisnik pritisne neku tipku koja nije broj provjerava se pomoću *try* petlje je li korisnik unio neki drugi znak koji ne može trenutno unijeti. Ako je korisnik unio neki znak koji nije broj poziva se *exception* koji printa poruku da korisnik mora unijeti broj. Time se osigurava da korisnik ne može unijeti znakove i ponovno se ispočetka pokreće petlja.
Nakon što je unesen ispravan broj pokreće se petlja za kriptiranje ili dekriptiranje.

```
if odabir == 1:
        tekst = input("Unesite tekst za enkriptirati: ")
        kriptirani_tekst = ""

        for znak in tekst:
            index = znakovi.index(znak)
            kriptirani_tekst += ključ[index]

        print(f"Originalan tekst: {tekst}")
        print(f"Kriptirani tekst: {kriptirani_tekst}")
```

Kod 7. Prikaz koda za enkripciju

Započinjući od kriptiranja, za koje je potrebno stisnuti broj jedan, zatražit će se unos teksta kojeg se želi kriptirati. Nakon što je tekst unesen počet će proces kriptiranja s pomoću ključa za enkripciju. Unutar petlje kod prolazi kroz svaki znak unesenog teksta, traži na kojoj se poziciji u normalnom nizu znakovi nalazi znak u tekst te zamjenjuje taj znak sa znakom u nizu znakova ključ i dodaje taj znak u prije definiranu varijablu zvanu kriptirani_tekst. Tako se osigurava da je tekst koji smo unijeli kriptiran s pomoću ključa za enkripciju. Na kraju se na konzoli ispisuju originalni tekst i kriptirani tekst koji će se koristiti u dekripciji.

```
elif odabir == 2:
    kriptirani_tekst = input("Unesite tekst za dekripciju: ")
    tekst = ""

    for znak in kriptirani_tekst:
        index = ključ.index(znak)
        tekst += znakovi[index]

    print(f"Originalan tekst: {kriptirani_tekst}")
    print(f"Kriptirani tekst: {tekst}")
```

Kod 8. Prikaz koda za dekripciju

Dekriptiranje će se provesti tako što se unese broj dva. Potrebno je unijeti već kriptirani tekst koji se mora dekriptirati te nakon što je unesen kriptirani tekst potražit će se znakovi da se dekriptira tekst. Kod kriptiranja uzimali su se znakovi iz normalnog niza pod nazivom znakovi, a ovdje će se prvo uzimati znakovi iz niza ključ. Svaki znak kriptiranog teksta će biti zamijenjen na način da se pozicija tog znaka pronađe unutar niza ključa i zamijeni sa znakom na istoj poziciji u nizu znakovi. Nakon što je znak pronađen u nizu znakova taj znak se doda na već postojeću varijablu teksta. Na kraju se na konzoli ispisuje kriptirani tekst i tekst koji je dobiven nakon dekripcije kriptiranog teksta.

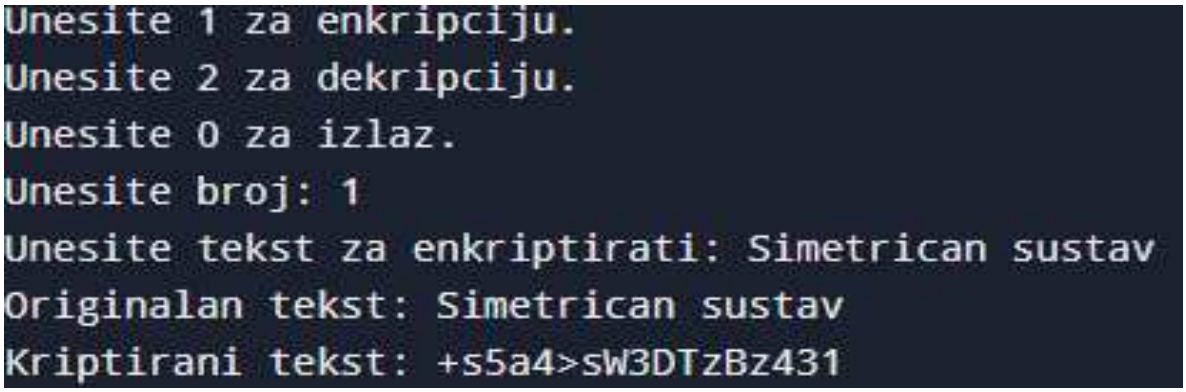
Zadnja dva dijela koda su za izlaz iz beskonačne petlje na način da se unese nula kad je dozvoljen unos za odabiranje kriptiranja i dekriptiranja ili će se ispisati na konzolu ako je unesen broj koji nije ispravan. Ako je unesen neki broj koji nije jedan, dva ili nula ispisuje se poruka „niste unijeli ispravan broj”.

```
elif odabir == 0:
        break
    else:
        print("Niste unijeli ispravan broj!")
```

Kod 9. Prikaz koda za izlaz iz petlje

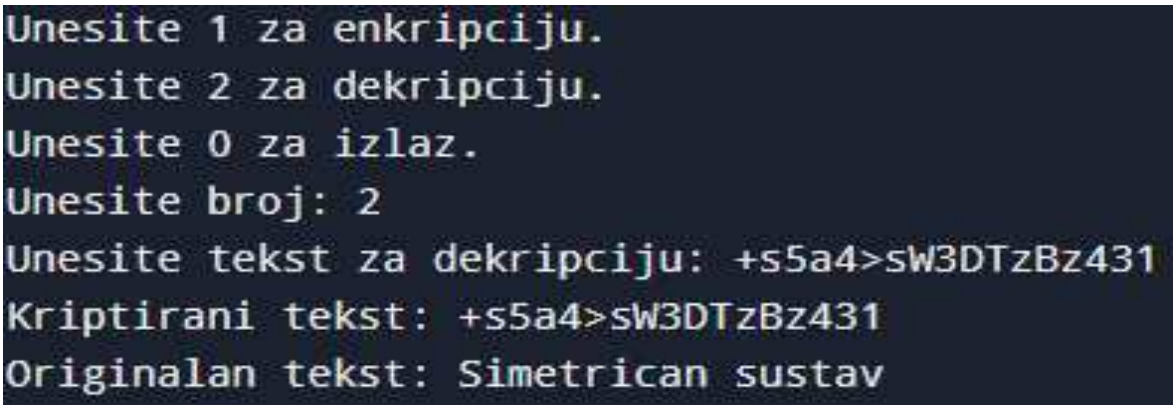
Ovo je isto tako i jednostavan primjer simetričnog kripto sustava u kojem se ključ koristi za enkripciju i dekripciju podataka ili u ovom slučaju teksta. Kako bi se ovaj kod mogao koristiti za simetrični kripto sustav potrebno je jednom pokrenuti program nakon što je odkomentiran dio koda koji na konzolu ispisuje niz nasumično raspoređenih znakova ili ključ te ga napraviti

konstantnim unutar koda. Kada bi se koristio takav ključ poslane poruke pošiljatelja bi izgledale kao na sljedećoj slici:



Slika 8. Prikaz kriptiranja poruke

Nakon uspješnog korištenja kriptiranja originalni tekst koji je razumljiv je zamijenjen s nizom znakova koji su nerazumljivi. Kriptirani tekst se kopira i pošalje osobi koja prima tekst i unosi ga u petlju za dekripciju koristeći isti ključ koji se koristio za kriptiranje. Dobiveni rezultat dekriptiranja kriptirane poruke se može vidjeti na sljedećoj slici:



Slika 9. Prikaz dekriptiranja poruke

Ovime se može potvrditi da je ključ korišten za kriptiranje i dekriptiranje i da je ovo simetrični kripto sustav. Ponovnim pokretanjem programa koji se ovdje nalazi se ponovno nasumično poredaju znakovi ključa tako da ako se pokuša unijeti isti kriptirani tekst neće biti moguće dobiti originalni tekst, a ako se unese isti originalni tekst dobiti će se novi kriptirani tekst koji se onda s pomoću istog ključa može dekriptirati.

5. ZAKLJUČAK

Enkripcija podataka predstavlja jednu od najvažnijih mjera zaštite informacija u digitalnom dobu gledajući da nikad nije bilo lakše pristupiti informacijama. Sama količina podatka koja se prenosi i pohranjuje na mrežama i lokalnim sustavima čini enkripciju neophodnom za čuvanje privatnosti i integriteta informacija.

Tehnološki napredci računala dovode do boljih i bržih računala. Jedan od potencijalnih problema su kvantna računala koja koriste algoritme koji imaju sposobnost rješavanja određenih matematičkih problema brže od klasičnih računala što je loša vijest za trenutne algoritme kriptiranja. Kako bi se zaštitili podatci razvija se post kvantni kriptografski algoritmi koji će se odupirati dekripciji kvantnog računala. Post kvantni kriptografski algoritmi ne ovise o matematičkim problemima, nego o novim kompleksnim problemima. Ovime će se osigurati da informacije postanu zaštićene od dekriptiranja kvantnim računalom.

U ovom završnom radu objašnjeno je da je enkripcija pretvaranje razumnog teksta u nerazumljiv tekst.

Spomenuto je kako algoritmi utječu na zaštitu kriptiranih podatka i zašto je bolje imati više bitova za enkripciju podataka. Spomenuta su dva algoritma.

Objašnjeno je zašto je bitno poznavati koji alat za enkripciju koristiti ako se skida s interneta i na što je potrebno paziti te zašto je potencijalno opasno skidati alat za enkripciju od nepouzdanih izvora.

Pošto komunikacije preko kojih šaljemo podatke mogu biti prisluškivane. Postoje opasnosti slanja podataka preko javnih mreža i na to je potrebno paziti prije i tijekom korištenja javnih mreža te se zaštititi u slučaju da koristimo javnu mrežu. Korištenje VPN-a najbolja je zaštita za sigurno slanje podataka preko javnih mreža jer podatci su zaštićeni kompliciranom enkripcijom.

Dodatna zaštita u slučaju neovlaštenog pristupa isto je dio zaštite podatka u slučaju zlonamjerne promjene podataka ili kopiranja podataka.

Objašnjen je idealan sustav za enkripciju podatka i svi uvjeti koje taj sustav mora zadovoljiti.

Prikazan je postupak enkripcije podataka na driveru računala i postupak koji je potreban da bi driver bio kriptiran i na kraju preko pojednostavljenog koda objašnjeno je što se događa tijekom enkripcije podataka i može se koristiti kao primjer kako funkcionira simetrični kripto sustav.

Tema završnog rada je odabrana iz znatiželje koliko je sigurna enkripcija podataka i kako se podatci kriptiraju. Gledajući na razvoj tehnologije kriptiranje podataka jedan je od najbitnijih

načina zaštite informacija i ostat će jedan od najbitnijih načina zaštite, jedino što se može promijeniti je kako se podatci kriptiraju tj. algoritmi kriptiranja. Pisanje koda za jednostavno kriptiranje bilo je izuzetno zanimljivo iskustvo. Na kraju kod koji je planiran za pojednostavljeno objašnjenje kako kriptiranje funkcionira mogao se koristiti i kao primjer simetričnog kripto sustava.

Literatura

1. Mauro Huculak. (5. Listopad 2022.) *Windows Central*. (Pristupljeno 10. Rujan 2024.) iz https://www.windowscentral.com/how-use-bitlocker-encryption-windows-10
2. Microsoft. (n.d.). *BitLocker overview*. (Pristupljeno 14. Rujan 2024) iz https://learn.microsoft.com/en-us/windows/security/operating-system-security/data-protection/bitlocker/
3. Nacionalni CERT i LS&S. (04. Rujan 2000). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 12. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2000-09-04.pdf
4. Nacionalni CERT i LS&S. (05. veljača 2003). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 10. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2003-02-05.pdf
5. Nacionalni CERT i LS&S. (22. Prosinac 2005). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 11. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2005-12-142.pdf
6. Nacionalni CERT i LS&S. (21. Studeni 2005). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 10. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2005-11-141.pdf
7. Nacionalni CERT i LS&S. (29. Veljača 2006). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 12. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2006-02-149.pdf
8. Nacionalni CERT i LS&S. (23. Travanj 2010). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 12. Rujan 2024) iz ttps://www.cis.hr/www.edicija/LinkedDocuments/NCERT-PUBDOC-2010-04-296.pdf
9. Nacionalni CERT i LS&S. (23. Travanj 2010). *Hrvatska akademska i istraživačka mreža.* (Pristupljeno 12. Rujan 2024) iz https://www.cis.hr/www.edicija/LinkedDocuments/NCERT-PUBDOC-2010-04-

296.pdf
10. *Triple_DES*. (n.d.). (Pristupljeno 11. Rujan 2024) iz Wikipedija: https://en.wikipedia.org/wiki/Triple_DES
11. *Wikipedija*. (n.d.). (Pristupljeno 11. Rujan 2024) iz Advanced_Encryption_Standard.

SAŽETAK

Ovaj završni rad objašnjava što je enkripcija podataka, koje sve enkripcije podataka postoje, kako algoritmi utječu na zaštitu kriptiranih podataka te zaštita podataka od prisluškivanja na javnoj mreži.

Spominje se dodatna zaštita od neovlaštenih pristupa podatcima i idealan sustav enkripcije podataka i koje uvjete taj sustav mora zadovoljavati.

Objašnjeno je na koji način se kriptira driver računala i pomoću pojednostavljenog primjera simetričnog kripto sustava što se događa tijekom enkripcije podataka.

Ključne riječi:

enkripcija, enkripcija podataka, algoritmi, alati za enkripciju, dodatna zaštita, zaštita preko javnih mreža, VPN, idealan sustav enkripcije, proces enkripcije

SUMMARY

This thesis explains what data encryption is, what data encryption exists, how algorithms affect the protection of encrypted data, and the protection of data from eavesdropping on a public network.

It mentions additional protection against unauthorized data access and the ideal data encryption system and the conditions that system must meet.

It is explained how the computer driver is encrypted and using a simplified example of a symmetric crypto system that occurs during data encryption.

Keywords:

encryption, data encryption, algorithms, encryption tools, additional protection, Protection over public networks, VPN, ideal encryption system, encryption process