Vizuelna reprezentacija postkvantnog kriptografskog algoritma Kyber

ALEKSA D. VUJNIĆ, Univerzitet u Beogradu,
Elektrotehnički fakultet, Beograd
ORCID: 0009-0004-8928-8713
ŽARKO S. STANISAVLJEVIĆ, Univerzitet u Beogradu,
Elektrotehnički fakultet, Beograd
ORCID: 0000-0003-0272-5139
MAJA B. VUKASOVIĆ, Univerzitet u Beogradu,
Elektrotehnički fakultet, Beograd
ORCID: 0000-0003-0647-1922

Stručni rad
UDC: 004:530.145
DOI: 10.5937/tehnika2501079V

Kriptografski algoritmi čine osnovu svih protokola i aplikacija koje obezbeđuju tajnost, integritet i dostupnost u savremenim informaciono-komunikacionim tehnologijama. Osnovna podela kriptografskih algoritama je na simetrične i asimetrične. Kada su u pitanju asimetrični kriptografski algoritmi, postoji osnovana indicija da bi sa razvojem kvantnih računara matematički aparat na kome se ovi algoritmi trenutno zasnivaju mogao biti prevaziđen. Iz tog razloga u toku je proces standardizacije tzv. Postkvantnih kriptografskih algoritama, odnosno algoritama koji bi bili sigurni i u slučaju postojanja kvantnih računara. Postkvantni kriptografski algoritam CRYSTALS-Kyber je jedan od najperspektivnijih kandidata za standardizaciju zbog svog jakog matematičkog utemeljenja i izuzetne efikasnosti. Njegova sigurnost zasniva se na matematičkom problemu rešetki, tačnije na Module-LWE problemu. Iako dobro dokumentovan, Kyber algoritam je složen i teško razumljiv. Ovaj rad opisuje razvoj vizuelne reprezentacije Kyber algoritma, čime se omogućava intuitivnije razumevanje njegovih operacija i strukture. Rešenje je posebno prilagođeno upotrebi u edukativne svrhe, sa namerom da olakša usvajanje znanja o novim konceptima u polju zaštite podataka.

Ključne reči: *vizuelizacija algoritama, zaštita podataka, kvantno računarstvo, kriptografija sa javnim ključem*

1. UVOD

Standardni kriptografski algoritmi su u širokoj upotrebi u vreme kada je pristup internetu dostupan velikom broju ljudi koji ga svakodnevno koristi. Za bilo koji pristup internet stranicama ili ostvarivanje komunikacije i razmene podataka preko interneta koristi se više različitih kriptografskih algoritama. Oni obezbeđuju tajnost i sigurnost komunikacije, poverenje u druge korisnike na mreži i integritet podataka. Njihov značaj je zbog toga veliki, a samim tim i potreba da budu sigurni i otporni na napade. Pretnja standardnim asimetričnim kriptografskim algoritmima postaju kvantni računari. Asimetrični kriptografski algoritmi koriste par ključeva – javni ključ i privatni ključ. Svaka strana u komunikaciji svoj privatni ključ čuva u tajnosti, dok javni objavljuje. Namena ovakvih algoritama je šifrovanje i potpisivanje podataka. Ubrzani razvoj kvantnih računara omogućava da se sve brže izvršavaju složenija matematička izračunavanja na kojima se temelje standardni asimetrični kriptografski algoritmi. To zahteva inovacije u domenu kriptografskih algoritama, sa posebnom pažnjom na otpornost na napade kvantnim računarima. Postoji više postkvantnih algoritama koji su u procesu standardizacije od strane Nacionalnog instituta za standarde i tehnologiju Sjedinjenih Američkih Država, među kojima se *CRYSTALS-Kyber* asimetrični algoritam izdvaja kao jedan od najefikasnijih protokola namenjenih za razmenu ključeva [1], [2]. Njegova sigurnost i performanse temelje se na matematičkom problemu rešetki koji ni kvantni računari nemaju mogućnost efikasno da rešavaju. Algoritam je u užem izboru za stan-

Adresa autora: Aleksa Vujnić, Univerzitet u Beogradu, Elektrotehnički fakultet, Beograd, Bulevar kralja Aleksandra 73
e-mail: aleksa.vujnic@gmail.com
Rad primljen: 03.02.2025.
Rad prihvaćen: 10.02.2025.

dardizaciju i kao takav je dobro dokumentovan i detaljno opisan. To ne menja činjenicu da je složeniji od standardnih kriptografskih algoritama. Razumevanje algoritma otežano je čak i stručnjacima u oblasti računarstva i informacionih tehnologija. Vizuelna reprezentacija rada Kyber algoritma bi omogućila intuitivno razumevanje operacija i struktura u algoritmu. Tako bi se olakšala edukacija i unapredile mogućnosti za analizu performansi i ranjivosti algoritma. Ovaj rad se fokusira na opis aplikacije za vizuelnu reprezentaciju Kyber algoritma.

Rad je organizovan po poglavljima. U drugom poglavlju dat je opis literature i opis koncepta postkvantne kriptografije, a zatim pregled i analiza postojećih aplikacija za vizuelni prikaz kriptografskih algoritama. U trećem poglavlju dati su detalji o načinu rada i specifikaciji Kyber algoritma. Četvrto poglavlje sadrži opis arhitekture aplikacije za vizuelnu reprezentaciju algoritma Kyber. U petom poglavlju dat je opis načina implementacije aplikacije i radni okviri i biblioteke koje su korišćene. Šesto poglavlje sadrži kvalitativnu evaluaciju implementirane aplikacije datu kroz poređenje sa aplikacijama opisanim u poglavlju 2. Zaključak je dat u sedmom poglavlju.

2. PREGLED LITERATURE

U ovoj sekciji biće opisan koncept postkvantnih kriptografskih algoritama, osnovne karakteristike Kyber algoritma kao jednog od kandidata za standardizaciju i značaj i primeri nekih softverskih rešenja za vizuelnu reprezentaciju kriptografskih algoritama. Informacije od značaja za idejno rešenje i implementaciju aplikacije za vizuelnu reprezentaciju algoritma Kyber uključuju opis principa postkvantne kriptografije i razlike u odnosu na tradicionalne kriptografske metode, karakteristike Kyber algoritma i značaj vizuelizacije algoritama u edukativne svrhe.

2.1. Postkvantna kriptografija i Kyber algoritam

Standardni kriptografski algoritmi svoju sigurnost utemeljuju na težini matematičkih problema faktorisanja velikih brojeva i izračunavanja diskretnih logaritama. Primeri algoritama koji počivaju na ovim problema su RSA algoritam [3] za šifrovanje pomoću javnog ključa i ECC grupa kriptografskih algoritama baziranih na eliptičnim krivama [4]. Ovi algoritmi se smatraju sigurnim i danas su u širokoj upotrebi jer klasični računari ne mogu u razumnom vremenu da reše matematičke probleme na kojima se zasnivaju. Sa druge strane, kvantni računari izvršavaju posebne algoritme koji nisu podržani za izvršavanje od strane klasičnih računara. Primer takvog algoritma je Šorov algoritam [5]. Formulisan je 1994. godine i služi za razlaganje celih brojeva na proste činioce. Algoritam radi u polinomijalnom vremenu, preciznije $O((\log N)^3)$, što znači da je ovo znatno brži algoritam za rastavljanje celog broja od većine drugih trenutno poznatih. Ako bi kvantni računar mogao da radi bez uticaja šumova i ostalih smetnji, Šorov algoritam bi mogao da se koristi za razbijanje šema kriptografije javnog ključa kao što je RSA. Motivacija za razvoj dovoljno velikih i naprednih kvantnih računara postoji usled navedenog, a sve to za posledicu ima i ubrzan razvoj postkvantne kriptografije. Dešava se razvoj novih algoritama koji su otporni i na napade kvantnim računarima i postoji veliko interesovanje za njihovu standardizaciju i izučavanje, a sve to u funkciji omogućavanja zaštite poverljivih informacija u budućnosti.

CRYSTALS-Kyber je postkvantni algoritam za razmenu ključeva i KEM (Key Encapsulation Mechanism) mehanizam za enkapsulaciju ključeva. Mehanizam za enkpasulaciju ključeva je sistem koji omogućava da se simetrični ključ razmeni između dve strane u komunikaciji. Ovaj mehanizam radi tako što jedna strana u komunikaciji šifruje simetrični ključ koji treba da se razmeni javnim ključem druge strane u komunikaciji. Zatim se taj šifrovani tekst šalje drugoj strani i ona dešifruje ključ koristeći sopstveni tajni ključ. Tako se vrši razmena simetričnog ključa uz generisanje samo jednog para asimetričnih ključeva. U transportu simetrični ključ je nečitljiv trećim stranama jer se uvek transportuje u šifrovanom obliku.

Dizajniran je da bude otporan na napade kvantnim računarima. Kao osnovu za sigurnost ima problem matematičkih rešetki, konkretno LWE (Learning With Errors) problem učenja sa greškom. Ovaj problem predstavlja proširenje sistema linearnih jednačina vektorom greške koji predstavlja vrstu šuma i značajno komplikuje dolazak do rešenja. Takođe, u Kyber algoritmu jednačine su definisane nad prstenom polinoma celobrojnih koeficijenata ograničenih modularnom aritmetikom. Konkretno varijanta LWE problema sa karakteristikom modularne aritmetike ima naziv Module-LWE [6]. Nad tim prstenom su definisane operacije sabiranja i množenja, a algoritam vrši optimizacije u izračunavanju istih koristeći NTT (Number Theoretic Transform) teorijsku transformaciju brojeva. Ona omogućava da se polinomi množe u $O(n \log n)$ algoritamskoj složenosti umesto u $O(n^2)$, što značajno ubrzava rad algoritama koji se u velikoj meri oslanjaju na operacije sa polinomima. Ova transformacija koristi algoritam nalik na brzu Furijeovu transformaciju FFT (*Fast Fourier Transform*) [7].

2.2. Programi za vizuelnu reprezentaciju kriptografskih algoritama

Vizuelizacija algoritama ima veliki značaj u edukaciji i analizi složenih algoritama. Posmatranje specifikacija algoritama i njihove implementacije u programskim jezicima ili pseudo-kodu može iziskivati

mnogo energije i vremena kako bi se razumeo način funkcionisanja i struktura podataka sa kojima algoritam radi. Tek kasnije pojedinac potencijalno može da se bavi implementacijom na nekom posebnom programskom jeziku ili platformi ili analizama sigurnosti, slabih tačaka ili performansi kriptografskih algoritama. Veliki broj stručnjaka i studenata se u nekoj meri bavi kriptografskim algoritmima i alati za vizuelizaciju im pomažu da takav rad za njih bude produktivniji, a razumevanje temeljnije.

Primer uspešnog softvera za vizuelizaciju je alat *CrypTool2* [8]. U pitanju je projekat otvorenog koda na čijem razvoju rade ljudi iz akademske zajednice i drugi volonteri. Razvijan je kao desktop aplikacija za Windows operativni sistem. Ovaj alat ima podršku za vizuelni prikaz unutrašnjih komponenti kriptografskih algoritama i njihove interakcije sa drugim pojedinačnim komponentama algoritma. Moguće je kreiranje radnih prostora na kojima se nalaze međusobno povezani elementi koji vrše jednostavnije obrade nad tokovima podataka i dobijanje simulacija celih algoritama njihovim povezivanjem. Na sajtu aplikacije nalazi se katalog radnih okvira sa različitim kriptografskim algoritmima koje je moguće modifikovati u svrhe eksperimentisanja i istraživanja. Dodatno, postoji i podrška za funkcije i elemente kojima se vrši kriptoanaliza standardnih i modernih algoritama za enkripciju. Tako je moguće na interaktivan način otkriti slabosti algoritma koji se vizualizuje. Takođe, postoji mogućnost za proširenje funkcionalnosti razvojem dodataka za aplikaciju (plugin programski interfejs) i time prilagoditi softver specifičnim potrebama pojedinca ili tima koji sa njim radi. Uz to, aplikacija CrypTool Online nudi podskup funkcionalnosti osnovne aplikacije u vidu veb aplikacije koja je samim tim i nezavisna od platforme.

Drugi primer dobrog softvera za vizuelizaciju, posebno u obrazovne svrhe je SHAVisual [9]. Ovaj program razvijan je na Tehnološkom univerzitetu u Mičigenu i koristi se kao alat u nastavi. Dizajniran je da pomogne studentima u razumevanju načina rada SHA (Secure Hash Algorithm) algoritma, a nastavnom osoblju kao pomoćni alat u nastavi. Program konkretno simulira SHA-512 verziju algoritma. Postoje verzije programa za Windows, MacOS i Linux operativne sisteme. Podržava tri moda rada:

- mod za demonstraciju (*Demo Mode*)
- mod za vežbu (*Practice Mode*)
- mod potpune funkcionalnosti (*Full Mode*).

Postoji prozor globalnog prikaza koji prikazuje fazu algoritma čiji se detaljni prikaz daje na glavnom prozoru aplikacije. Mod za demonstraciju dizajniran je tako da uvede korisnika u način rada algoritma, tačnije njegovih osnovnih komponenata i operacija. Koriste se kraće poruke osnovnog teksta i samo jedna runda algoritma. Fokus korisnika usmerava se na ključne operacije i detalje umesto na ponavljajuće elemente. Mod za vežbu ima sličnu strukturu kao mod za demonstraciju s tim da se korisnik kreće korak po korak kroz algoritam. U ovom modu rezultati operacija su delimično sakriveni i korisnik mora da dopuni tačan rezultat operacije kako bi otvorio sledeći korak algoritma. Mod potpune funkcionalnosti daje uvid u potpuni oblik SHA-512 algoritma. Korisnik može da unese proizvoljan tekst ili generiše slučajan niz karaktera za ulaz algoritma.

Oba opisana rešenja predstavljaju dobro realizovane aplikacije koje daju verodostojan prikaz kriptografskih algoritama. Obim njihovih funkcionalnosti je drugačiji i prilagođen je konkretnim slučajevima upotrebe svake od aplikacija. *CrypTool2* daje mnogo veću fleksibilnost u realizovanju vizuelnih reprezentacija različitih algoritama i brojne dodatne funkcionalnosti. Sa druge strane, *SHAVisual* je usko specijalizovan za prikaz SHA-512 algoritma. Postojanje više modova rada, od kojih su dva posebno namenjena za lakše učenje i usvajanje detalja u vezi sa algoritmom su od velike važnosti i izdvajaju ga od drugih aplikacija. Lakše je, brže i intuitivnije naučiti detalje konkretno tog algoritma kroz *SHAVisual* iako program nema funkcionalnosti koje nudi *CrypTool2*.

3. DETALJI KYBER ALGORITMA

Kyber algoritam po definiciji ima sledeće celobrojne parametre: *n, k, q, η1, η2, du*, i *dv*. Vrednosti parametara u zavisnosti od varijante Kyber algoritma date su u tabeli 1.

Tabela 1. Prikaz varijanti Kyber algoritma sa pridruženim vrednostima parametara

Varijanta	n	k	Q	η1	η2	du	dv
Kyber512	256	2	3329	3	2	10	4
Kyber768	256	3	3329	2	2	10	4
Kyber1024	256	4	3329	2	2	11	5

U svim varijantama parametar *n* ima vrednost 256, a parametar *q* ima vrednost 3329. Razlog zašto *n* ima vrednost 256 je to što je cilj ovog algoritma da enkriptuje i razmeni ključ veličine 256 bita. Manje vrednosti parametra *n* bi zahtevale da se više bita ključa enkodira u jedan koeficijent polinoma.

Posledica toga bilo bi da sigurnost algoritma postaje manja od predviđene. Veće vrednosti parametra *n* bi smanjile mogućnosti za lako skaliranje sigurnosti algoritma korišćenjem parametra *k* o čemu će naknadno biti reči. Parametar *q* odabran je kao prost broj koji bi zadovoljio uslov da parametar *n* deli (*q*-1) bez ostatka. To je neophodan uslov za ubrzanje operacije množenja polinoma putem NTT transformacije.

Parametar *k* varira kako bi podesio dimenziju rešetke kao umnožak od *n*. Menjanje ovog parametra u celobrojnim vrednostima od 2 do 4 podešava nivo sigurnosti u algoritmu. To takođe menja i efikasnost algoritma, koja se povećava sa smanjenjem nivoa sigurnosti. Ostali parametri su odabrani da balansiraju između sigurnosti, veličine šifrovanog teksta i verovatnoće greške. Sva tri skupa parametara postižu verovatnoću greške manju od 2^{128}.

Koraci u radu *Kyber* algoritma su generisanje ključeva, šifrovanje i dešifrovanje. *Kyber* radi sa nizovima bajtova i polinomima stepena *n* po modulu *q* i svako pominjanje polinoma odnosiće se na polinom sa ovim karakteristikama. Javni ključ sadrži seme za generisanje *A* matrice polinoma dimenzije *k* i enkodovani polinom *T* dobijen kao proizvod matrice *A* i vektora slučajno generisanih polinoma *S* na koji je dodat E vektor polinoma šuma tj. greške. Vektori polinoma *S* i *E* su oba dužine *k*, a polinomi koji su u njima sadržani imaju male vrednosti koeficijenta (između -2 i 2 za *Kyber* algoritam gde parametri *η1* i *η2* imaju vrednost 2). Tajni ključ sadrži enkodovan vektor slučajno generisanih polinoma *S*. Javni ključ je veličine 1184 bajta, dok je tajni ključ veličine 1152 bajta.

Proces enkripcije zahteva javni ključ i 128 bita podataka simetričnog ključa koji se enkriptuju i treba da se razmene sa drugom stranom. Koristeći seme za generisanje matrice *A* iz javnog ključa, u proces enkripcije prvo se generiše transponovana matrica A^T i dekoduje polinom *T*. Zatim se generiše niz slučajnih polinoma sa koeficijentima malih vrednosti *R* i *E1*, oba dužine *k*. Proizvod matrice A^T i vektora *R* na koji je dodat vektor *E1* predstavlja polinom *U*. On se enkoduje u prvi deo šifrovanog teksta *C1*. Dalje se formira polinom *V*. On nastaje od slučajno generisanog polinoma sa malim vrednostima koeficijenata *E2* koji se dodaje na proizvod vektora polinoma *T* i *R*. Od 128 bita poruke koja se enkriptuje formira se polinom sa koeficijentima koji enkoduju bite poruke i on se dodaje na polinom *V*. Ovo predstavlja drugi deo šifrovanog teksta *C2*. Šifrovani tekst *C* ukupne je dužine od 1088 bajtova, a sastoji se od dela *C1* dužine 128 bajtova i dela *C2* dužine 960 bajtova.

Proces dekripcije zahteva šifrovani tekst (*C1* i *C2*) i tajni ključ od kog se dekoduje niz polinoma *S.* Od *C1* dela tajnog ključa dobija se vektor polinoma *U*, a od *C2* dela tajnog ključa dobija se polinom *V*. Od polinoma *V* oduzima se polinom nastao kao proizvod polinoma *S* i *U* nakon čega se dobija polinom iz kog se dekoduju biti poruke, njih 256. [2]

4. ARHITEKTURA APLIKACIJE ZA VIZUELNU REPREZENTACIJU ALGORITMA KYBER

Razumevanje načina rada *Kyber* algoritma je izazovno iz razloga što algoritam uključuje kompleksne matematičke operacije i strukture. Ideja je da se taj proces u što većoj meri olakša i da za ključne operacije i delove algoritma postoje jasni vizuelni prikazi sa detaljima o stanju podataka koji se menjaju. Cilj je da to olakša analizu algoritma i edukaciju o načinu njegovog rada. Kroz posmatranje stanja elemenata u algoritmu i njihove međusobne interakcije, cilj je jasno prikazati zavisnosti operacija i podataka u algoritmu. Za tako nešto je samo čitanjem implementacije i specifikacije potrebno mnogo vremena i pažnje.

Kada se ponudi vizuelni prikaz, brže je i lakše ustanoviti strukturu algoritma na više nivoa. To znači primetiti koje su osnovne operacije i koraci u algoritmu koje deli sa većinom drugih kriptografskih algoritama – generisanje ključeva, enkripcija i dekripcija. Dodatno, interakcijom kroz korisnički interfejs korisnik može da ustanovi specifičnosti u vezi sa Kyber algoritmom, šta uključuje svaki od glavnih delova algoritma i koja je razlika u odnosu na druge, jednostavnije kriptografske algoritme. Prikaz svakog od koraka bi nudio mogućnost ulaska u stranice koje prikazuju detalje o delovima glavnih koraka u algoritmu. Tu se jasno vidi kako se generišu slučajni podaci i transformišu iz jednog oblika u drugi. Pod transformacijom najpre se misli na prikazivanje procesa enkodovanja i dekodovanja uz opcionu kompresiju i dekompresiju podataka. Svaki bajt i koeficijent polinoma treba da budu prikazani sa jasno naznačenom vrednosti koju imaju. Takođe je bitno prikazati kako i na osnovu čega su te vrednosti dobijene.

Kao što je prethodno opisano, cilj je dati vizuelni prikaz tako da se na osnovnom prikazu aplikacije nalaze osnovni koraci u algoritmu, sa čijih stranica se može pristupiti detaljnijem prikazu pojedinačnih koraka u okviru njih. Ideja je da algoritam poseduje tri osnovne stranice:

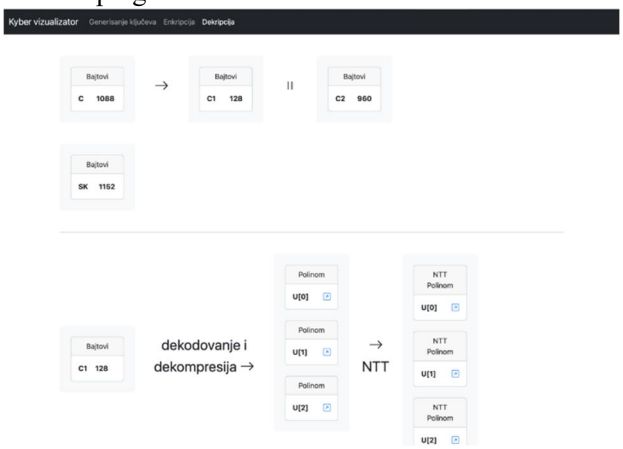
- stranica za generisanje ključeva
- stranica za enkripciju
- stranica za dekripciju.

Prelazak sa jedne na bilo koju drugu osnovnu stranicu moguće je uraditi putem navigacionog menija na vrhu stranice. Na svakoj od stranica biće prikazane strukture podataka sa kojima algoritam radi. To mogu

biti nizovi bajtova ili polinomi. Svaki je prikazan komponentom nalik na klasni dijagram u UML jeziku, sa osnovnim podacima o tipu, imenu i sadržaju na samoj komponenti. Za neke elemente moguće je kliknuti na strelicu koja vodi do stranice koja prikazuje trenutni sadržaj podataka u strukturi. Pored toga, moguće je videti kako se do tih vrednosti došlo.

U *Kyber* algoritmu podaci mogu biti slučajno generisani, mogu nastati od ulaznih podataka ili biti proizvod operacije nad postojećim podacima. Korisnicima je to jasno prikazano i sa osnovne stranice koraka algoritma, dok za detalje o transformacijama nad podacima treba da otvore stranicu za pregled određene strukture. Stranice koje demonstriraju detalje o transformacijama i nastanku podataka u algoritmu su:

- stranica za generisanje polinoma matrice javnog ključa
- stranica za generisanje polinoma sa malim vrednostima koeficijenata u vektoru greške i privatnom ključu
- stranica za enkodovanje
- stranica za dekodovanje
- stranica za enkodovanje i kompresiju
- stranica za dekodovanje i dekompresiju.

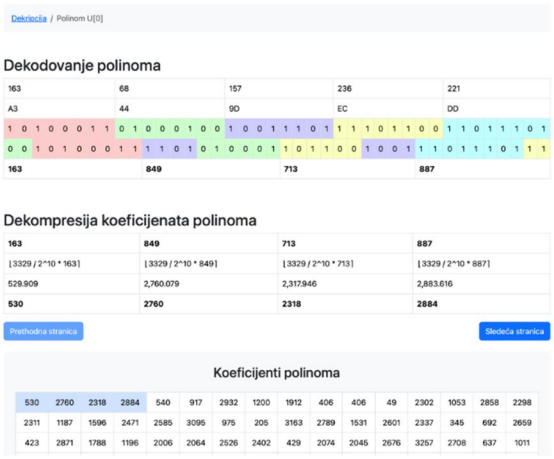


Slika 1 - Prikaz dekripcije, osnovna stranica

Izazov u vizuelizaciji navedenih elemenata i stranica je celovit prikaz vrednosti polinoma sa 256 koeficijenata i šire slike o interakciji između elemenata u algoritmu. Vrednosti svakog polinoma i niza bajtova biće sakriveni sa osnovnih stranica. Biće prikazane samo komponente sa osnovnim informacijama i mogućnost za otvaranje stranice na kojoj se prikazuju detalji o stanju i načinu nastanka podataka. Primer je stranica za dekripciju na slici 1, gde vizuelni elementi jasno prikazuju kako se šifrovani tekst deli na delove *C1* i *C2* sa naznačenim veličinama u bajtovima.

Od tih elemenata nastaje novi element, niz polinoma *U* kroz proces dekompresije i dekodovanja. Svaki je predstavljen posebnim elementom na korisničkom interfejsu, dok su operacije naznačene simbolima i nazivima za lako uočavanje i povezivanje kroz posmatranje interfejsa.

Detalje procesa dobijanja niza polinoma *U* moguće je pregledati klikom na dugme za više detalja koje otvara novu stranicu sa detaljima dekompresije i dekodovanja, prikazanih na slici 2.



Slika 2 - Prikaz stranice za dekodovanje i dekompresiju

5. IMPLEMENTACIJA APLIKACIJE ZA VIZUELNU REPREZENTACIJU ALGORITMA KYBER

Implementacija aplikacije za vizuelnu reprezentaciju *CRYSTALS-Kyber* algoritma izvedena je korišćenjem *Angular* radnog okvira [10] za razvoj *front-end* dela aplikacije. Takođe je korišćena *Bootstrap* biblioteka [11] za dizajn interfejsa i pripadajućih elemenata. Ključne operacije algoritma i detalje neophodno je prikazati kroz interaktivni grafički interfejs. Radi jednostavnosti, aplikacija je razvijena da podrži *Kyber768* verziju algoritma koja ima jedinstveno određen skup parametara.

Klijentska aplikacija organizovana je po uzoru na predloženu arhitekturu radnog okvira *Angular* u kome se i razvija. Logika aplikacije i operacija sa podacima smeštena je u kod klase servisa *Angular* koji se oslanja na *TypeScript* implementaciju *Kyber* algoritma. Elementi korisničkog interfejsa i logika koja se tiče obrade unosa i interakcije sa korisnikom smeštena je u klase *Angular* komponenti.

Logika aplikacije oslanja se na *TypeScript* implementaciju *Kyber* algoritma. Osnovne operacije algoritma su modifikovane tako da čuvaju stanje svakog podatka u okviru objekata kojima raspolaže servis. Podaci koji se modifikuju tokom obrade zahtevaju da se njihovo stanje eksplicitno čuva pre nastavka obrade. Tako aplikacija može da prikaže kako su podaci izgledali u svim koracima rada algoritma. Servis čuva stanje i preko jedinstvenog objekta servisa sve komponente koje se koriste za prikaz interfejsa imaju pristup unikatu istog. Ovakav mehanizam ubacivanja zavisnosti između komponenti je izuzetno intuitivan za korišćenje i *Angular* radni okvir to obezbeđuje bez potrebe za eksternim bibliotekama i zavisnostima.

Svaka stranica, uključujući osnovne i one koje demonstriraju detalje o transformacijama i detaljima struktura podataka u algoritmu implementirane su kao *Angular* komponente. Svaka stranica ima deo koji definiše HTML šablon izgleda stranice i *TypeScript* deo koji definiše logiku prikaza podataka i interakciju sa korisnikom.

Svaka osnovna stranica dostupna je direktno iz navigacionog menija koji se uvek nalazi u gornjem delu stranice. Sa njega moguće je direktno pristupiti stranici koja prikazuje generisanje ključeva, enkripciju i dekripciju. Na osnovnim stranicama će simboli i grafički elementi pomoći u tome da se razume interakcija između struktura podataka. Naime, bojenje pozadine radi grupisanja polinoma koji čine matricu ili vektor i dodavanje simbola operacija i transformacija između struktura nad kojima se primenjuju čine da korisnik lako uoči veze i tok algoritma. Sa druge strane, otvaranje stranica sa detaljima omogućava duboko razumevanje toka procesa obrade nad podacima bez vizuelnog opterećivanja osnovnih stranica.

Implementacija stranice sa detaljima o generisanju slučajne matrice polinoma A sadrži prikaz niza generisanih pseudo-slučajnih vrednosti na osnovu kojih se formiraju koeficijenti polinoma koji čine matricu. Na dnu stranice nalazi se prikaz svih koeficijenata polinoma i bojom su označeni oni koeficijenti čije se izračunavanje trenutno prikazuje na gornjem delu ekrana. Na isti način organizovana je i stranica koja nudi prikaz detalja generisanja polinoma sa malim vrednostima koeficijenata u okviru vektora greške i tajnog ključa. Prikazan je način na koji se vrši odabir vrednosti malih koeficijenata na osnovu generisanog niza bitova.

Stranice koje prikazuju detalje enkodovanja i dekodovanja služe da jasno naznače način na koji se dešava konverzija između polinoma u niz bajtova i obrnuto. Postoje sledeće konfiguracije enkodovanja i dekodovanja u zavisnosti od broja bita kojim se predstavlja pojedinačni koeficijent polinoma uz opciono postojanje kompresije:

- 12 bita bez kompresije
- 10 bita sa kompresijom
- 4 bita sa kompresijom
- 1 bit sa kompresijom.

Na ovim stranicama je dat prikaz procesa za svaki pojedinačni koeficijent polinoma uz mogućnost hoda kroz ceo polinom. Bojama je naznačeno smeštanje vrednosti koeficijenata u određene bite kako bi bilo jasno uočiti proces koji nije toliko jednostavan za razumevanje iz specifikacije. U donjem delu stranice posebnom bojom naznačeni su biti i koeficijenti čiji se metod konverzije trenutno nalazi na vrhu stranice. Moguće je kretati se kroz koeficijente polinoma i niz bajtova pritiskom na dugme „prethodni“ ili „sledeći“ koji se nalaze sa leve i desne strane stranice, odmah ispod prikaza konverzije i opciono kompresije. Ovi procesi su suština rada *Kyber* algoritma i LWE principa na kojem se temelji njegova sigurnost. Zato je uložena posebna pažnja da svaki pojedinačni način enkodovanja i dekodovanja ima posebnu komponentu koja daje vizuelni prikaz načina rada. Na vrhu svake stranice sa detaljima postoji *breadcrumbs* statusna traka koja na sebi ima ucrtanu putanju do trenutno prikazanog elementa algoritma. Ovaj panel je interaktivan i moguće je kliknuti na druge elemente u putanji za navigaciju do njih.

Još jedan izazov u implementaciji je čuvanje stanja *TypeScript* objekata koji se menjaju u toku rada algoritma. Neophodno je napraviti duboke kopije objekata koji se obrađuju u algoritmu i čuvati ih u okviru stanja klase *Angular* servisa. Kako *TypeScript* jezik

nema ugrađene metode za duboko kopiranje objekata, potrebno je osloniti se na biblioteke koje nude tu funkcionalnost. *Lodash JavaScript* biblioteka [12] je pouzdana uslužna biblioteka otvorenog koda koja ima mnoštvo funkcionalnosti i u širokoj je upotrebi. Njena *cloneDeep* metoda je u stanju da kreira duboku kopiju proizvoljnog objekta.

6. KVALITATIVNA EVALUACIJA

U ovom poglavlju biće dat uporedni prikaz prethodno pomenutih alata za vizuelnu reprezentaciju kriptografskih algoritama i aplikacije za vizuelizaciju algoritma Kyber.

Ono što je zajedničko svim aplikacijama za vizuelnu reprezentaciju kriptografskih algoritama pomenutih u drugom poglavlju, pa i vizuelizatoru Kyber algoritma, je da prikazuju redosled koraka algoritma, tok podataka, njihovu transformaciju i neke specifične detalje o operacijama koje se nad njima primenjuju. Sve aplikacije to rade na dobar način, nudeći pregledan i jasan prikaz svega što je bitno da se zapazi u vezi sa predstavljenim algoritmom.

Aplikacija za vizuelnu reprezentaciju algoritma Kyber je razvijena sa ciljem da bude upotrebljena u edukativne svrhe. Ona ima mogućnost da prikaže jedan algoritam, što važi i za *SHAVisual*, dok *CrypTool2* nudi mogućnost prikaza više algoritama i radne prostore za pravljenje novih vizuelizacija od dostupnih komponenata. Sa druge strane, Kyber aplikacija daje prikaz šire slike o radu algoritma i interaktivan korisnički interfejs kojim se mogu otvarati stranice sa detaljnijim pregledom stanja podataka u algoritmu i operacijama u toku obrade. *SHAVisual* ne nudi interaktivan interfejs na tom nivou, ali prikazuje neophodne detalje na glavnom prikazu. On sadrži i modove funkcionisanja koji su posebno namenjeni učenju algoritma, a Kyber aplikacija za vizuelizaciju ima samo osnovni mod funkcionisanja.

Svaka aplikacija je razvijena da odgovara definisanim zahtevima i slučajevima upotrebe i stoga postoje razlike u dizajnu i implementaciji. Implementirana aplikacija je u skladu sa zahtevima koji se tiču upotrebe u nastavi i poseduje sve karakteristike i funkcionalnosti koje su za to važne.

7. ZAKLJUČAK

Ovaj rad je predstavio aplikaciju za vizuelnu reprezentaciju CRYSTALS-Kyber algoritma. Postkvantni kriptografski algoritmi igraju ključnu ulogu u bezbednosti podataka u budućnosti, ali njihova složenost otežava intuitivno razumevanje. Intuitivan korisnički interfejs i detaljan uvid u svaki korak algoritma pomažu da se algoritam razume bolje i detaljnije nego čitanjem specifikacije.

Optimizacija memorijske efikasnosti i podrška za složenije varijante algoritma mogu biti potencijalne tačke nadogradnje postojećeg rešenja koje bi dodatno doprinele kvalitetu i korisničkom iskustvu. Takođe, proširenje podrškom za druge postkvantne algoritme stvorila bi se šira platforma za edukaciju o postkvantnoj kriptografiji.

8. ZAHVALNICA

Ovaj rad je delimično finansiran od strane Ministarstva nauke, tehnološkog razvoja i inovacija prema ugovoru: 451-03-65/2024-03/200103.

LITERATURA

[1] Joppe Bos, Leo Ducas, Eike Kiltz, Ttancrede Lepoint, Vadim Lybashevsky, John M. Schanck, Peter Schwabe, *CRYSTALS Kyber: a CCA-secure module-lattice-based KEM,* 2018.

[2] Joppe Bos, Leo Ducas, Eike Kiltz, Ttancrede Lepoint, Vadim Lybashevsky, John M. Schanck, Peter Schwabe, *CRYSTALS Kyber – Algorithm Specifications And Supporting Documentation*, 2019.

[3] Evgeny Milanov, *The RSA algorithm*. RSA laboratories pp. 1–11, 2009.

[4] Shamsher Ullah, Jiangbin Zheng, Nizamud Din, Muhammad Tanveer Hussain, Farhan Ullah, Mahwish Yousaf, *Elliptic Curve Cryptography; Applications, challenges, recent advances, and future trends: A comprehensive survey*, Computer Science Review, Vol. 47, 2023.

[5] Ray LaPierre, *Introduction to Quantum Com-puting*, pp. 177-192, 2021.

[6] Oded Regev, *The Learning with Errors Problem*, 2010.

[7] Adrianto Satriawan, Rella Mareta, Hanho Lee, *A Complete Beginner Guide to the Number Theoretic Transformation (NTT)*, 2024.

[8] CrypTool 2 Documentation and Resources [Internet] CrypTool Org; 2023 [citirano 26.01.2025.] Dostupno na: https://www.cryptool.org/en/ct2/resources/

[9] Jun Ma, Jun Tao, Melissa Keranen, Jean Mayo, Ching-Kuang Shene, Chaoli Wang, " *SHAvisual: A Visualization Tool for the Secure Hash Algorithm*"

[10] Angular 17 Documentation [Internet] Google Inc.; 2023 [citirano 26.01.2025.] Dostupno na: https://v17.angular.io/docs

[11] Bootstrap 5.3 Documentation [Internet] Bootstrap; 2024 [citirano 26.01.2025.] Dostupno na: https://getbootstrap.com/docs/5.3

[12] Lodash 4.17 Documentation [Internet] Lodash; 2024 [citirano 26.01.2025.] Dostupno na: https://-lodash.-com/docs/4.17.15

SUMMARY

VISUAL REPRESENTATION OF POST-QUANTUM CRYPTOGRAPHIC ALGORITHM KYBER

Cryptographic algorithms form the foundation of protocols and applications that ensure confidentiality, integrity, and availability in modern information and communication technologies. These algorithms are broadly classified into symmetric and asymmetric cryptography. However, with the advancement of quantum computing, the mathematical foundations of current asymmetric cryptographic algorithms may become vulnerable. To address this challenge, efforts are underway to standardize post-quantum cryptographic algorithms that remain secure even in the presence of quantum computers. CRYSTALS-Kyber post-quantum cryptographic algorithm is one of the most promising candidates for standardization due to its strong mathematical foundation and exceptional efficiency. Its security is based on the mathematical lattice problem, more precisely the Module-LWE problem. Although well documented, the Kyber algorithm is complex and difficult to understand. This paper describes the development of a visual representation of the Kyber algorithm, thus enabling a more intuitive understanding of its operations and structure. The solution is specially adapted to be used for educational purposes, with the intention of facilitating the acquisition of knowledge about new concepts in the field of data protection.

Key Words: *algorithm visualization, data security, quantum computing, public key cryptography*