Implementacija sistema zaštite elektronskih dokumenata primjenom AES algoritma

Dušan Perišić, Andrija Karadžić, Ognjen Letić, Ivan Tot

***Apstrakt*—Pored mnoštva pozitivnih efekata današnje informaciono društvo sa sobom nosi i mnoge prijetnje pri kojima se mnoge kompanije i pojedinci susreću sa gubicima kako finansijske tako i druge prirode, pa sve više raste potreba za zaštitom informacija. U radu je predstavljena implementacija sistema zaštite elektronskih dokumenata primjenom AES algoritma, sa fokusom na ličnim dokumentima. Osnovni elemente sistema čine: upravljačka aplikacija, ključ i elektronski dokument. Implementacija upravljačke aplikacije rađena je u programskom jeziku C#; kao ključ koristi se usb flash memorija formatirana, označena sa generisanim elementima ključa od strane upravljačke aplikacije; dok elektronski dokument predstavlja fajl koji se štiti.**

***Ključne riječi*—kriptografija, enkripcija podataka, AES algoritam.**

I. UVOD

Činjenice pokazuju da živimo u informacionom društvu koje teži ka što je moguće većoj digitalizaciji, što doprinosti većoj povezanosti, bržem razvoju i prosperitetu, sa jedne i većoj izloženosti informacionim rizicima i prijetnjama, sa druge strane.

U tekstovima koji se bave predviđanjima budućnosti danas se veoma mnogo govori o nanotehnologiji i biotehnologiji kao naukama vremena koje je pred nama. Očekuje se da će one dati sasvim nove, sofisticirane projekte i omogućiti primjenu u elektronici, računarstvu, vojsci, medicini, proizvodnji hrane, energije, itd. Ova predviđanja i naznačeni okviri novih projekata navode nas na sve ozbiljnija razmišljanja i o promjenama danas poznatih granica između čovjeka i mašina. Između ostalog, projekti budućnosti, koji se usmjeravaju na period posle 2030; godine, sadrže i mogućnosti implantacije nanorobota ili ultramikročipova (ne većih od nekoliko nanometara) u ljudski organizam, čija bi uloga bila nadgradnja i poboljšanje intelektualnih sposobnosti čovjeka.[2]

Sve to govori o sve većoj vezi između čovjeka i računara (mašine), značaja informacija i informacionim izazovima i prijetnjama koji prožimaju današnje društveno-ekonomske infrastructure.

Dušan Perišić – student, Vojnoelektronsko inžinjerstvo, Vojna akademija, Univerzitet odbrane Beograd, Pavla Jurišića Šturma 33, 11000 Beograd, Srbija (e-mail: kontakt@dusanperisic.com).

Andrija Karadžić – student, Vojnoelektronsko inžinjerstvo, Vojna akademija, Univerzitet odbrane Beograd, Pavla Jurišića Šturma 33, 11000 Beograd, Srbija (e-mail: andrija.karadzic@va.mod.gov.rs).

Ognjen Letić – student, Vojnoelektronsko inžinjerstvo, Vojna akademija, Univerzitet odbrane Beograd, Pavla Jurišića Šturma 33, 11000 Beograd, Srbija (e-mail: ognjen.letic@va.mod.gov.rs).

Ivan Tot – docent, Vojnoelektronsko inžinjerstvo, Vojna akademija, Univerzitet odbrane Beograd, Pavla Jurišića Šturma 33, 11000 Beograd, Srbija (e-mail: totivan@gmail.com).

II. OSNOVNI POJMOVI

U savremenoj kriptologiji **kriptoanaliza** se može definisati kao nauka o mogućnosti otkrivanja otvorenog teksta poruke, na osnovu posjedovanja šifrata, bez poznavanja ključa. Uspješnom kriptoanalizom šifrovane poruke dolazi se do otvorenog teksta ili ključa.

Otvoreni tekst (engl. plaintext) predstavlja svaki sadržaj (govor, računarski podaci, itd) koji se sastoji od konačnog broja simbola određenog alfabeta (slova, brojevi, bitovi, bajtovi itd). Otvoreni tekst koji predstavlja određenu logičku cjelinu naziva se **poruka** (engl. message).

Postupak transformacije otvorenog teksta u zaštićen oblik primjenom određenog kriptografskog postupka naziva se **šifrovanje** ili **enkripcija** (engl. Encription, Encipherment). U savremenim sistemima zaštite šifrovanje poruke realizuje se primjenom odgovarajuće kriptografske transformacije (algoritma) i korišćenjem tajnog parametra (ili javnog kod asimetričnih algoritama) – kriptografskog ključa.

Kod **simetričnih** kriptografskih algoritama isti ključ se koristi i za šifrovanje i za dešifrovanje, dok se kod **asimetričnih** algoritama koriste različiti ključevi pri šifrovanju i dešifrovanju otvorenog teksta, odnosno šifrata.

Sadržaj dobijen nakon realizacije procesa šifrovanja naziva se **šifrovani tekst** ili **šifrat** (engl. ciphertext). Inverzan postupak, transformacija šifrata u otvoreni tekst naziva se **dešifrovanje** ili **dekripcija** (Decryption, Decipherment). [1]

III. KRATAK ISTORIJAT KRIPTOGRAFIJE

Kada je pismo postalo sredstvo komunikacije, pojavila se potreba da se neka pisma sačuvaju od tuđih pogleda. Tada je i kriptografija ugledala svjetlost dana. Od samog početka, enkripcija podataka koristila se prvenstveno u vojne svrhe. Jedan od prvih velikih vojskovođa koji je koristio šifrovane poruke bio je Gaj Julije Cezar. Naime, kada je Cezar slao poruke svojim vojskovođama, on je te poruke šifrovao tako što su sva ili pojedina slova u tekstu bila pomjerana za tri, četri ili više mjesta u abecedi. Takvu poruku mogli su da dešifruju samo oni koji su poznavali pomeri za pravilo. Poznata Cezarova izjava prilikom prelaska Rubikona u šifriranom dopisivanju glasila bi: fqkf ofhzf kyz. Pomicanjem svakog slova za šest mjesta u abecedi lako se može pročitati pravi smisao poruke: Alea iacta est (kocka je bačena). [4]

Danas se primjenjuju različiti mnogo složeniji postupci koji se implementiraju na računarima velikih brzina.

IV. AES ALGORITAM

AES (engl. Advanced Encryption Standard) je jedan od kriptografskih algoritama za zaštitu digitalnih podataka. AES standard temelji se na simetričnom Rijndael algoritmu, a kao standard razvijen je da bi postupno zamijenio DES, čija sigurnost u današnje vrijeme nije dovoljna.

Slično kao i DES, AES je razvijen u privatnom sektoru, no u saradnji sa američkom vladom. AES je definisan i prihvaćen od strane NIST-a (engl. National Institute of Standards and Technology) u FIPS 197 dokumentu, te se kao takav može koristiti u američkim državnim i drugim institucijama.

U ovom trenutku, što se tiče uporabe na Internetu, DES, pa i drugi simetrični algoritmi još uvijek su zastupljeni u većoj mjeri, no vremenom će i AES pronaći svoje mjesto, pogotovo zato što DES kao takav ne predstavlja adekvatno rješenje u sistemima koji zahtijevaju najviše nivoe sigurnosti.

Za razliku od npr. DES i IDEA algoritama koji podatke šifruju u 64-bitnim blokovima, AES, odnosno Rijndael algoritam, šifruje 128-bitne blokove podataka. Dužina ključa može biti 128, 192, ili 256 bita. Sam Rijndael algoritam je oblikovan tako da je moguće šifrovanje podataka u blokovima različitih dužina i s različitim dužinama ključeva, no to nije definisano kroz standard. Obzirom na ključeve, uobičajeno je algoritme nazivati AES-128, AES-192 i AES-256.

Prednosti AES algoritma se ogledaju u tome što je otporan na poznate napade, veoma je brz, moguć je paralelni dizajn, kao i implementacija na mnogim procesorima i smart karticama. [3]

V. KRIPTOANALIZA AES ALGORITMA

AES standard definiše tri dužine ključa: 128 bita, 192 bita i 256 bitova. Obzirom na dužinu ključa, razlikuje se i broj koraka koji se izvode tokom procesa šifriranja odnosno dešifrovanja.

U ovom trenutku dužine ključeva zadovoljavaju sigurnosne zahtjeve u većini, ako ne i svim primjenama. Osim toga, AES se temelji na Rijndael algoritmu, koji omogućava šifrovanje blokova podataka različitih dužina te primjenu ključeva koji također mogu imati različitu dužinu, što omogućava buduća unapređenja standarda, ukoliko to bude potrebno. U algoritmu do sad nisu pronađeni nesigurni ili potencijalno nesigurni ključevi (kao npr. kod DES-a).

Nadalje, prilikom razvoja Rijndael algoritma posebna pažnja posvećena je tome da algoritam bude K-siguran i hermetički. K-sigurnost znači da je algoritam siguran ukoliko sve moguće strategije napada na njega imaju očekivanje, trajanje i zahtjeve za memorijski prostor identične ili veće u odnosu na ostale algoritme koji šifruju blokove podataka iste dužine. Hermetičnost znači da algoritam ne sadrži druge slabosti koje nisu prisutne niti u ostalim algoritmima koji koriste ekvivalentne blokove i dužine ključeva.

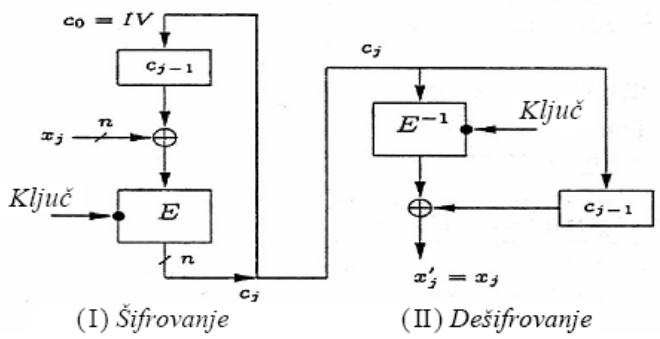
Iako AES nije osjetljiv na napade korištenjem linearne i diferencijalne kriptoanalize, postoje naznake da je osjetljiv na algebarske napade, a posebno na novu XSL (engl. eXtended Sparse Linearization) metodu napada. Pokazuje se da se takvim napadom mogu postići rezultati mnogo bolji nego napadima primjenom sile (engl. brute force).[3]

VI. CBC KRITOGRAFSKI MOD

Mod ulančavanja blokova (CBC – Cipher Block Chaining) povezuje blokove šifrata tako što se rezultat šifrovanja prethodnih blokova koristi pri šifrovanju tekućeg bloka. Drugim riječima, svaki blok se koristi za modifikaciju šifrovanja sledećeg bloka tako da svaki blok šifrata (ST) zavisi ne samo od tekućeg bloka otvorenog teksta (OT) već i od svih prethodnih blokova OT. Načini na koje se to može ostvariti su raznovrsni.

U CBC modu (Sl. 1.), taj uticaj se realizuje tako što se izvršava operacija "ekskluzivno ili" (XOR) između OT i neposredno prethodnog bloka ST, a zatim se tako dobijeni blok podataka šifruje. Preciznije, koraci ovog postupka se mogu opisati na sledeći način:

1. U povratni registar se smjesti inicijalna vrijednost.
2. Blok otvorenog teksta i sadržaj povratnog registra se spregnu operacijom ekskluzivne disjunkcije i tako dobijeni blok se transformiše šifarskom transformacijom E čime se formira blok šifrata c_j.
3. U povratni registar se smesti c_j i proces se ponavlja od koraka 2 sve dok ima blokova za šifrovanje.



Sl. 1. Na slici je prikazan grafički prikaz operacija u CBC modu

Proces dešifrovanja sledi direktno i odvija se na sledeći način:

1. U povratni registar se smjesti inicijalna vrijednost.
2. Blok šifrata c_j dešifruje se primjenom transformacije E^{-1}, tako dobijeni blok teksta i sadržaj povratnog registra se spregnu operacijom ekskluzivne disjunkcije i tako se dobije blok otvorenog teksta.
3. U povratni registar se smjesti c_j i proces se ponavlja od koraka 2 sve dok ima blokova za dešifrovanje.

Matematički, proces šifrovanja i dešifrovanja može se prikazati sledećim relacijama, respektivno:

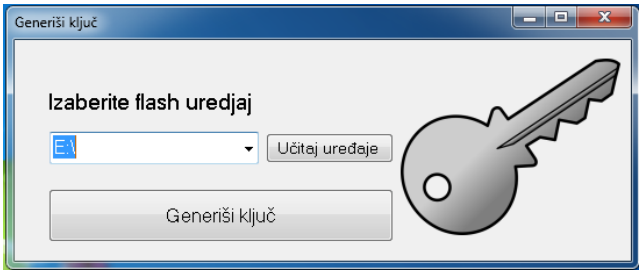
$$ST_i = E_k(OT_i \oplus ST_{i-1})$$
$$OT_i = ST_{i-1} \oplus D_k(ST_i)$$

VII. IMPLEMENTACIJA KLJUČA

Ključ koji se koristi pri šifrovanju i dešifrovanju digitalnih dokumenata predstavlja tajni elemenat i realizovan je pomoću usb flash memorije (flash uređaj).

Zahtjevi koje je potrebno da zadovoljava flash uređaj prije generisanja ključa su da memorijski prostor bude potpuno prazan i da ima naziv "SIGURNOSNI_KLJUC". U slučaju da zahtjevi nisu zadovoljeni aplikacija neće prikazivati uredjaj u listi uređaja.

Ključ se generiše pri korisničkim pokretanjem dijela aplikacije za te namjene (Sl. 2.).



Sl. 2. Na slici je prikazan prozor za generisanje ključa gdje korisnik, nakon što je prehodno pripremio flash uredjaj po osnovnim zahtjevima, ima mogućnost da iz komboboksa izabere jedan od flash uredjaja koji su spremni za generisanje ključa (u ovom slučaju izabran je uređaj na adresi E:\).

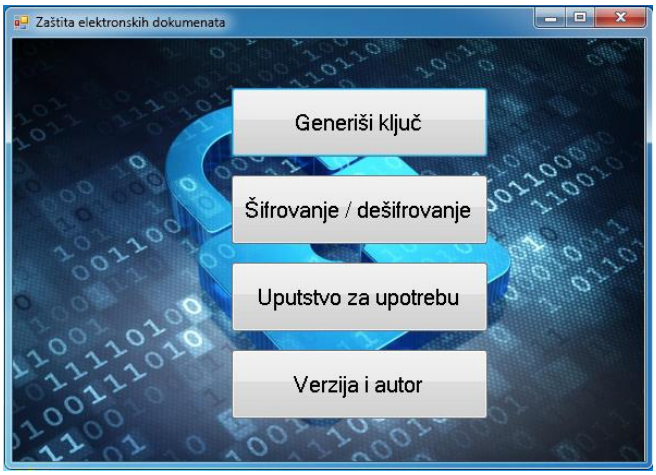
Aplikacija generiše ključ tako što u memorijski prostor pripremljenog flash uredjaja upisuje fajl u kome se nalaze četiri linije:

1. hash otisak koji se formira MD5 hash funkcijom na osnvu informacija o flash uredjaju (ukupna veličina memorijskog prostora, naziv – labela, tip uređaja, veličina slobodnog prostora na uređaju);
2. hash otisak koji se formira MD5 hash funkcijom na osnovu otiska iz prve linije, ključa AES algoritma i inicijalnog vektora;
3. ključ AES algoritma koji se generiše od 1024 slučajno generisana karaktera;
4. inicijalni vektor koji se generiše na isti način kao i ključ AES algoritma.

VIII. UPRAVLJAČKA APLIKACIJA

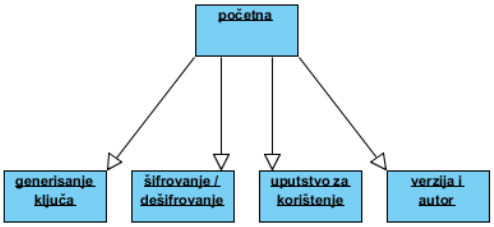
Upravljačka aplikacija dizajnirana je tako da korisniku omogući maksimalnu funkcionalnost, kako bi prilagođenost korisnicima koji nisu iz informatičkog sektora bila što prihvatljivija.

Upravljčka aplikacija sastoji se od nekoliko elemenata koji su predstavljeni dijalog prozorima (Sl. 3).



Sl. 3. Na slici je prikazan izgled početnog, odnosno glavnog, prozora upravljačke aplikacije koja omogućava kretanje kroz aplikaciju.

Raspored prozora i osnovnih funkcionalnosti prikazan je na Sl. 4.



Sl. 4. Na slici je prikazana osnovna hijerarhijska struktura dijalog prozora aplikacije koja odgovra njenim funkcionalnostima. Početna stranica (dijalog prozor) sadrži veze ka svim drugim.

Osnovna ili početna stranica sadrži veze ka drugim prozorima koja realizuju funkcionalnosti aplikacije, a to su:

- generisanje ključa
- šifrovanje/dešifrovanje
- uputstvo za korištenje
- podaci o verziji i autoru

Funkcionalnost generisanja ključa je objašnjeno u prethodnom poglavlju. Šifrovanje/dešifrovanje obezbjeđuje funkcionalnost šifrovanja odnosno dešifrovanja. Aplikacija automatski prepoznaje postupak koji treba da se radi pomoću ekstenzije izabranog fajla i na osnovu toga prikazuje dugme za početak šifrovanja, odnosno dešifrovanja (Sl. 5.).

U dijelu uputstvo za korištenje dato je detaljno korisničko objašnjenje koje vodećim putem pokazuje korisniku način upotrebe funkcionalnosti alikacije, dok u poslednjem dijelu korisnik može da pronađe informacije o verziji aplikacije, autoru i kontakt podacima autora.

Ekstenziju šifrovanog fajla (“.cry”) aplikacija dodaje prilikom šifrovanja, a uklanja je nakon dešifrovanja.

Sl. 5. Na slici je prikazan izgled prozora za šifrovanje / dešifrovanje fajla, odnosno elektronskog dokumenta. Obzirom da je izabran elektronski dokumenat PDF formata aplikacija zaključuje da je sledeća aktivnost šifrovanje i prikazuje dugme “Šifruj fajl” koje pokreće funkcju šifrovanja.

IX. ZAKLJUČAK

Činjenicu da je bezbijednost informacija jedan on segmenata kritične informacione infrastructure sve više i sve ozbiljnije počinju da shvataju kako državni i privatni organi tako i pojedinci.

U radu je predstavljeno jedno od rješenja zaštite ličnih elektronskih dokumenata. Predstavljena je prva verzija aplikacije koja vrši šifrovanje i dešifrovanje elektronskih dokumenata. Aplikacija ne pravi razliku u tipovima fajlova osim što prepoznaje šifrovani fajl, tako da je moguće šifrovati fajlove bilo kog tipa.

Obzirom da se AES smatra jakim kriptografskim algoritmom bezbijednost šifrovanih podataka se ne dovodi u pitanje, jedini problem predstavlja tajnost ključa što je briga korisnika.

Aplikaciju je moguće i dalje razvijati, neki od prijedloga na kojima je moguće napraviti poboljšanja:

- u fajlu ključa čuvati i hash otisak datuma kreiranja ključa, u aplikaciji podesiti vremenski interval nakon kog će aplikacija upozoravati korisnika da bi bilo poželjno da promijeni svoj ključ, a u fajlu čuvati i stare ključeve, kako bi fajl koji je ranije zaključan bilo moguće otključati;
- omogućiti razmjenu i kopiranje ključa
- omogućiti da se pri prepoznavanju flash uredjaja, pri kreiranju heširane informacione vrijednosti koristi i serijski broj flash uređaja.

To su neki od prijedloga za nastavak rada na ovom projektu i razvoju sledeće verzije aplikacije.

LITERATURA

[1] B. Jovanović, "Bezbednost informacija", Zaštita računarskih sistema – skripta za kadete Vojne akademije, 2014 Beograd

[2] D. Koridić, "Sistem čovek – mašina", Vojna psihologija – skripta za kadete Vojne akademije, 2012 Beograd

[3] CARNet CERT u saradnji s LS&S, "AES algoritam", http://www.cis.hr/www.edicija/LinkedDocuments/CCERT-PUBDOC-2003-08-37.pdf

[4] Singidunum, Univerzitet."KRIPTOLOGIJA I." *Acta facultatis medicae Naissensis* 27.4.

ABSTRACT

Aside from the many positive effects of today's IT society carries with it a lot of threats, because of which a lot of companies and individuals face losses of both financial and other types, therefore the need for information security rises. This paper presents an implementation of a system to protect electronic documents by applying the AES algorithm, mainly focusing on personal identifications. System comprises some basic elements: control application, key, and an e-document. Implementation of the control application was coded in C#; the key is an USB flash memory, formatted marked with generated elements of the key by the control application; the e-document is the file to protect.

Implementing an electronic document protection system using AES algorithm

Dušan Perišić, Andrija Karadžić, Ognjen Letić, Ivan Tot