Generator certifikata
Tehnička dokumentacija
Verzija 1.0

Studentski tim: Kruno Tomola Fabro

Nastavnik: prof. Marin Golub

Sadržaj

1. Opis razvijenog proizvoda

1.1 Realizirano

Razvijen je programski proizvod za stvaranje i pohranu X.509v3 certifikata. Certifikate izdaje trenutačno aktivno certifikacijsko tijelo. Aktivno certifikacijsko tijelo se mora učitati iz datoteke formata PKCS#12. Svi stvoreni certifikati sa svojim privatnim ključem se pohranjuju u odabranu datoteku formata PKCS#12. Programom se upravlja preko grafičkog sučelja organiziranog po tabovima. A tabovi su:

- Aktivno certifikacijsko tijelo – Preko ovog taba se učitava certifikacijsko tijelo iz PKCS#12 datoteke koja sadrži certifikat i odgovarajući privatni ključ te lanac certifikata do njega. Takvo učitano certifikacijsko tijelo će biti izdavač nadalje stvorenih certifikata.
- Izrada certifikata – Sadrži druge tabove koji služe za stvaranje, spremanje certifikata i privatnog ključa za razne namjene. Certifikate izdaje aktivno certifikacijsko tijelo učitano u prije spomenutom tabu. Osim u jednom dolje spomenutom slučaju. Tabovi koje sadrži su:
    - Certifikat za vršni CA – Stvaranje samo potpisanog certifikata za vršno certifikacijsko tijelo. Za ovaj slučaj nije potrebno da je učitano aktivno certifikacijsko tijelo.
    - Certifikat za CA – Stvaranje certifikata za certifikacijsko tijelo.
    - Certifikat za server – Stvaranje certifikata za potrebu autentifikacije servera preko IP-a, DNS-a ili URI-a.
    - Certifikat za potpisivanje koda – Stvaranje certifikata za potpisivanje izvornog koda programa.

Ostvaren program služi isključivo za stvaranje certifikata. Ako se želi za stvarno koristiti certifikati stvoreni s ovim programom korisnik se mora pobrinuti za sljedeće:

- Distribucija certifikata.
- Zadovoljavanje pravila jedinstvenosti atributa *engl. Distinguished name* subjekata u certifikatima izdanih od strane istog certifikacijskog tijela.
- Izvedba CRL-a ili OCSP-a.

Skup svih opcija za kreiranje certifikata je:

- Kreiranje naziva subjekta sačinjenog od raznih atributa. DN atribut mora biti prisutan dok su ostali proizvoljni.
- Trajanje valjanosti certifikata izraženo u godinama, mjesecima i danima.
- Maksimalna duljina lanca certifikata kod certifikacijskih tijela.
- Unos URI-a ili IP-a ili DNS-a koji će se autentificirati.
- Odabir tipa para ključeva koji će se generirati za certifikat.
- Odabir algoritma potpisa certifikata.
- Spremanje certifikata i privatnog ključa. Za spremanje se unosi putanja do direktorija u kojem se nalazi ili se treba stvoriti PKCS#12 datoteka s unesenim nazivom. Za datoteku je potrebno navesti lozinku za enkripciju/dekripciju te opcionalno lozinku za integritet datoteke. Ako lozinka za integritet nije definirana upotrijebit će se lozinka za enkripciju. U datoteku će se spremiti certifikat, njegov privatni ključ i lanac certifikata pod unijetim *engl. friendly name*. Također će se certifikat spremiti kao zasebna datoteka.

Program poštuje (RFC 5280, https://tools.ietf.org/html/rfc5280) i (RFC 6818, https://tools.ietf.org/html/rfc6818).

1.2 Moguća proširenja

Dodatne opcije koje bi se moglo implementirati, a koje bi značajno proširile uporabivost generiranih certifikata su:

- Definiranje načina dohvaćanja CRL ili točke pristupa za OCSP.
- Definiranje načina dohvaćanja certifikata.
- Definiranje pristupa uslugama *engl. Time stampa*.
- Unos politike uporabe certifikata za podređene certifikate.
- Unos ograničenja na nazive podređenih certifikata.

Implementacija dodatnih opcija bi omogućila:

- Provođenje nad stvaranim certifikatima ograničenja na naziv i politiku definiranih u certifikatu aktivnog certifikacijskog tijela.
- Otvara mogućnost za integraciju s certifikacijskim serverom te bi oni zajedno bili certifikacijsko tijelo s uslugama:
    - kreiranja certifikata
    - distribucija i pohrana certifikata
    - distribucija CRL
    - usluga OCSP
    - usluga TSA

2. Tehničke značajke

2.1 Tehnologija

- Izvorni kod je napisan isključivo u Java 8 programskom jeziku.
- Koriste se standardne biblioteke iz JDK 8.
- Za kreiranje i pohranu certifikata se koriste biblioteke „Bouncy Castle Crypto“ dostupnih na stranici https://bouncycastle.org.
- Grafičko sučelje je realizirano Swing kosturom.
- Projekt je ostvaren s programskim okruženjem NetBeans IDE 8.

2.2 Arhitektura

Izvorni kod se sastoji od tri strogo razdvojena sloja:

1. Prezentacijski sloj
2. Servisni sloj
3. Domenski sloj

Za sloj infrastrukture se koriste standardne biblioteke JDK 8 i biblioteka „Bouncy Castle Crypto“. Infrastrukturni sloj se neće detaljnije opisivati.

2.2.1 Prezentacijski sloj

Sastoji se od grafičkog sučelja realiziranog sa Swing kosturom. Ovaj sloj se može podijeliti na 3 strogo razdvojena sloja:

1. Glavni prozor koji donji sloj organizira u tabove. Sastoji se od jedne klase „GUI“. Ovaj sloj se proširuje dodavanjem ploča donjeg sloja kao tabove. Također opcionalno ima referencu na objekt koji realizira sučelje „CertificateCreatorService“ is sloja servisa.
2. Ovaj sloj se sastoji od skupa nezavisnih ploča koje se prikazuju unutar glavnog prozora. Svaka ploča ima svoju jedinstvenu svrhu. U grubo se mogu podijeliti u dvije grupe:
   a. Grupa ploča koja služi za kreiranje specifičnih certifikata. Sve klase ove grupe nasljeđuju jednu apstraktnu klasu „GenCertPanel“. Ovaj sloj se u ovoj grupi proširuje dodavanjem novih klasa koji nasljeđuju „GenCertPanel“ klasu. Svaka ploča je namijenjena da stvara svoj objekt iz sloja servisa koji realizira sučelje „CertificateBroker“.
   b. Ostale ploče. Trenutačno se sastoji od samo jedne ploče „CAPanel“ koja služi za učitavanje i prikaz certifikata aktivnog certifikacijskog tijela. Prilikom aktivacije certifikacijskog tijela se kreira objekt koji implementira sučelje „CertificateCreatorService“ te ga se predaje glavnom prozoru.

   Ploče ovog sloga su većinom sastavljene od manjih ploča iz donjeg sloja.
3. Sloj koji se sastoji od skupa nezavisnih ploča koje su glavni dijelovi u kreiranju gornjeg sloja. Sve klase ovog sloja nasljeđuju „JPanel“ klasu iz Swing kostura. Ovaj sloj se proširuje kreiranjem klasa koje nasljeđuju „JPanel“ klasu.

2.2.2 Servisni sloj

Ovaj sloj provodi razdvajanje po dvije točke:

- Razdvajanje prikupljanja podataka koje mora unijeti korisnik od prikupljanja ostalih podataka potrebnih za kreiranje certifikata.
- Razdvajanje dretve koja prikuplja korisničke podatke od dretve koja skuplja ostale podatke.

Definiraju ga tri sučelja:

- „CertificateBroker“ – Odgovornost klasa koje implementiraju „CertificateBroker“ jest držanje podataka dobivenih iz gornjeg sloja, koje je korisnik unjeo te na temelju njih i usluga koje pruža

„SignerService“ sučelje stvoriti objekt iz domenskog sloja, klase koja implementira sučelje „CertificateBuilder“ koja je također iz domenskog sloja. Realizirana je jedna apstraktna klasa „Broker“ koja implementira „CertificateBroker“ koju nasljeđuju ostale klase ovog sloja s prije spomenutom odgovornošću.

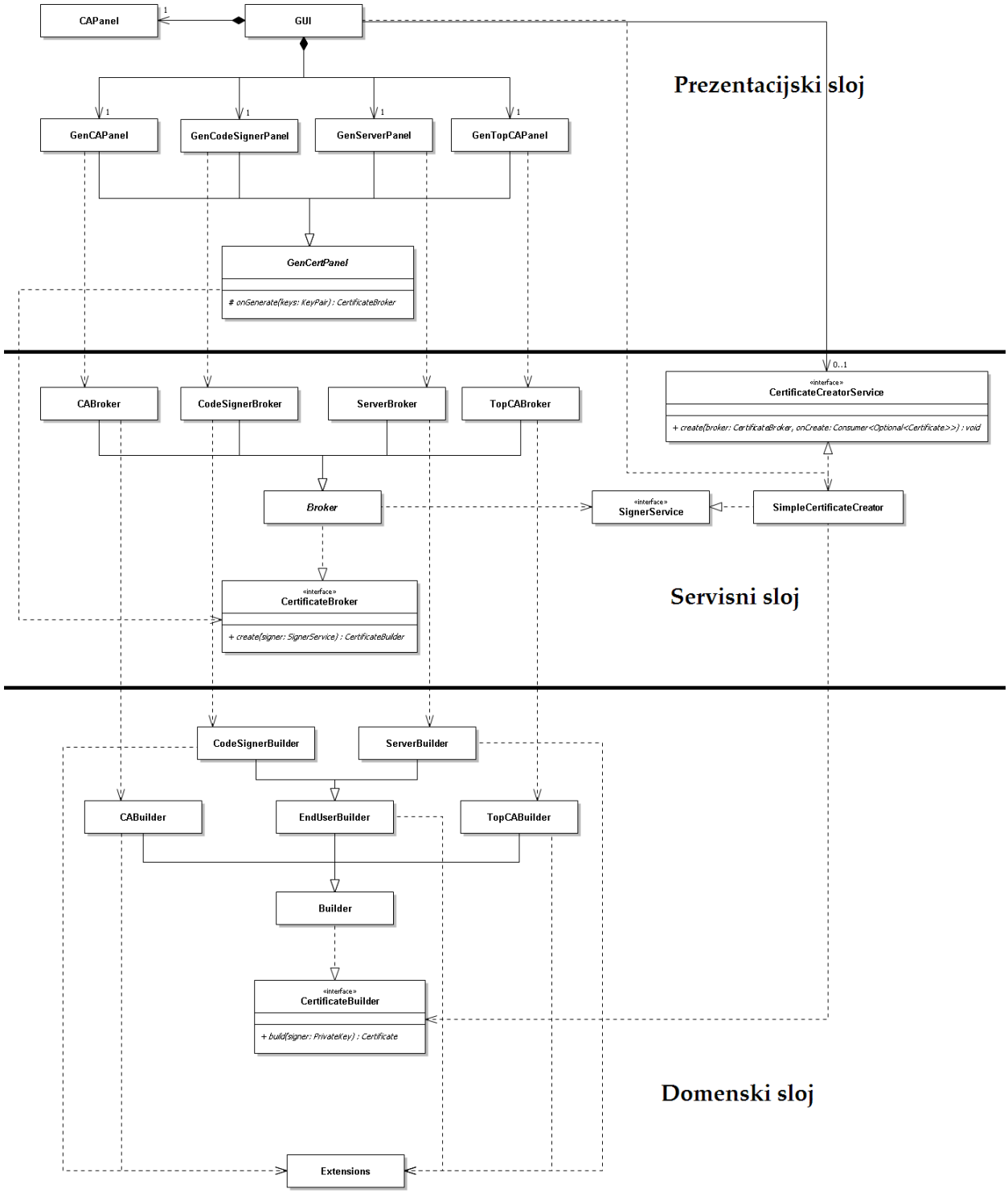
- „SignerService“ – Definira usluge koje se pružaju „CertificateBroker“ sučelju za obavljanje svoje zadaće.
- „CertificateCreatorService“ – Definira uslugu kreiranja certifikata na temelju objekta klase koja implementira „CertificateBroker“ sučelje. Ovo sučelje omogućuje stvaranje certifikata u drugoj dretvi čime se pozivajuća dretva oslobađa tog posla. Implementirana je jedna klasa „SimpleCertificateCreator“ koja realizira sučelja „CertificateCreatorService“ i „SignerService“ te koja posao kreiranja certifikata obavlja u pozivajućoj dretvi.

2.2.3 Domenski sloj

Pristupa mu se preko sučelja „CertificateBuilder“ koje omogućuje izradu certifikata. To sučelje realizira klasa „Builder“ koja nudi stvaranje generičkog certifikata. Izvedene klase definiraju kakav se točno certifikat treba izraditi. To definiranje se ostvaruje preko ekstenzija definiranih za X.509v3 certifikat. Te ekstenzije se ostvaruju u jednoj statičnoj klasi „Extensions“ koja tražene ekstenzije implementira pomoću klasa iz paketa „sun.security.x509“. Klase iz „sun.security.x509“ su vlasništvo Sun korporacije te je moguće da budu bili uklonjene u budućnosti. Zbog tog razloga su sve ekstenzije sadržane u klasi „Extensions“ kako bi se izbjegla krutost dizajna u slučaju takvog događaja.

2.3 Dijagram razreda

Spomenuta arhitektura je vidljiva na slici 1.



Slika 1. UML dijagram razreda

3. Upute za korištenje

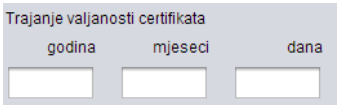
U ovom poglavlju se razmatraju razni scenariji korištenja. Ukupno tri scenarija. Gdje se zadnji scenarij može provesti tek nakon što se provede drugi scenarij. Prilikom stvaranja bilo kojeg certifikata će se pojavljivati podskup skupa implementiranih dijelova sučelja za unos podataka. Svi implementirani dijelovi su prikazani i objašnjeni u nastavku:

- U ovom dijelu sučelja (slika 2.) se unosi naziv subjekta certifikata. Postupak za dodavanje jedne informacije je sljedeći:
  - Odabrati vrstu informacije preko padajućeg izbornika.
  - U polje lijevo od gumba „Dodaj“ treba unijeti podatak.
  - Pritiskom gumba „Dodaj“ podatak se dodaje u donji popis od kojeg će se sastaviti naziv subjekta certifikata.

  Moguće je imati samo jedan podatak iste vrste. Gumb „Dodaj“ se pretvara u „Ukloni“ ako je podatak odabranog tipa već dodan u donji popis. Pritiskom na „Ukloni“ podatak odabranog tipa će se ukloniti s donjeg popisa. Obavezno se mora unijeti informacija tipa „DN“ koji je jedinstven za certifikacijsko tijelo koje će izdati certifikat s ovim nazivom subjekta.

Slika 2. Naziv subjekta

- Trajanje valjanosti certifikata (slika 3.) se unosi preko ovog dijela sučelja. Moguće je definirati trajanje u mjernim jedinicama:
  - godina
  - mjesec
  - dan



Slika 3. Trajanje valjanosti certifikata

- Ovaj dio sučelja (slika 4.) će se pojavljivati kada se želi kreirati certifikat za certifikacijsko tijelo. Sastoji se od samo jednog polja u koje je potrebno unijeti maksimalnu duljinu lanca certifikata podređenih ovom certifikatu.

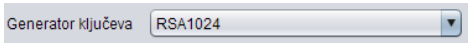
Slika 4. Maksimalna duljina lanca certifikata

- Kada se radi certifikat za server, pojavit će se dio (slika 5.) s kojim se može definirati jedan od tri podatka servera koji se bude autentificirao s ovim certifikatom:
  - IP
  - URI

- DNS

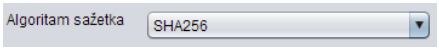
Slika 5. URI, IP, DNS

- U ovom dijelu (slika 6.) se odabire algoritam javnog i privatnog ključa gdje će javni ključ biti dio certifikata.



Slika 6. Algoritam za generiranje ključeva

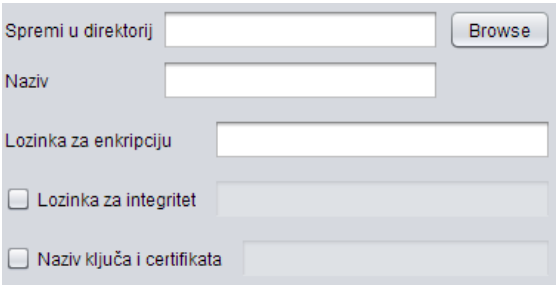
- Odabire se algoritam za izračun sažetka certifikata (slika 7.).



Slika 7. Algoritam sažetka

- Ovaj cijeli dio (slika 8.) služi za pohranu certifikata i privatnog ključa. Certifikat i privatni ključ se pohranjuju u datoteku formata PKCS#12. Podaci koji se trebaju unijeti su:
  - Treba odabrati direktorij u koji će se spremati.
  - Naziv nove ili postojeće PKCS#12 datoteke. Bez nastavka „.p12“.
  - Lozinka s kojom će se enkriptirati datoteka.
  - Opcionalno lozinka za integritet datoteke. Ako ova lozinka nije definirana, upotrijebit će se lozinka za enkriptiranje.
  - Opcionalno naziv para certifikat i ključ pod kojim će se spremiti u datoteku.

  Biti će stvorene dvije datoteke. Jedna PKCS#12 s certifikatom i privatnim ključem te druga datoteka koja će sadržavati samo certifikat.



Slika 8. Spremanje certifikata i privatnog ključa

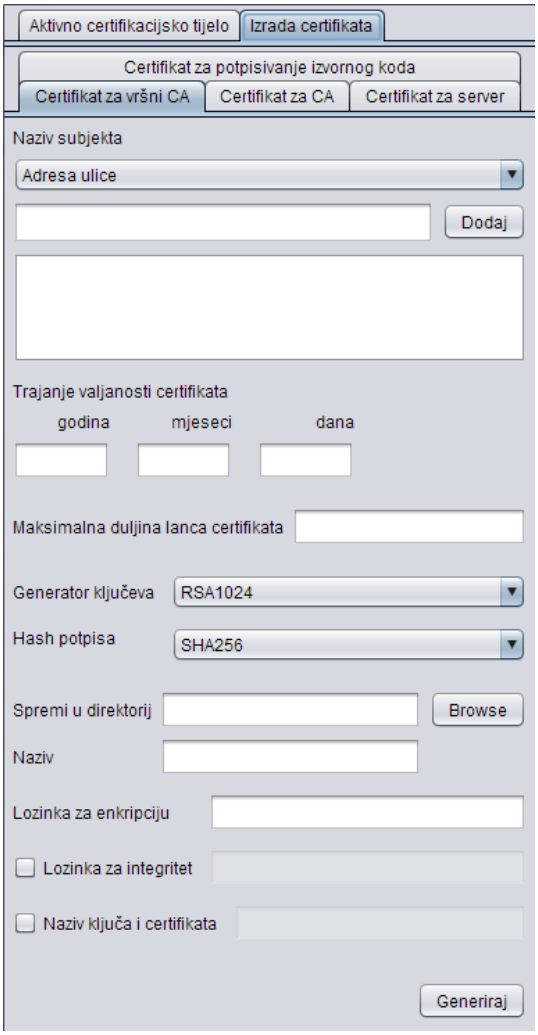
3.1 Kreiranje certifikata vršnog certifikacijskog tijela

Ovaj scenarij uporabe opisuje kako kreirati certifikat za vršno certifikacijsko tijelo koje se može upotrijebiti za izdavanje drugih certifikata. Postupak:

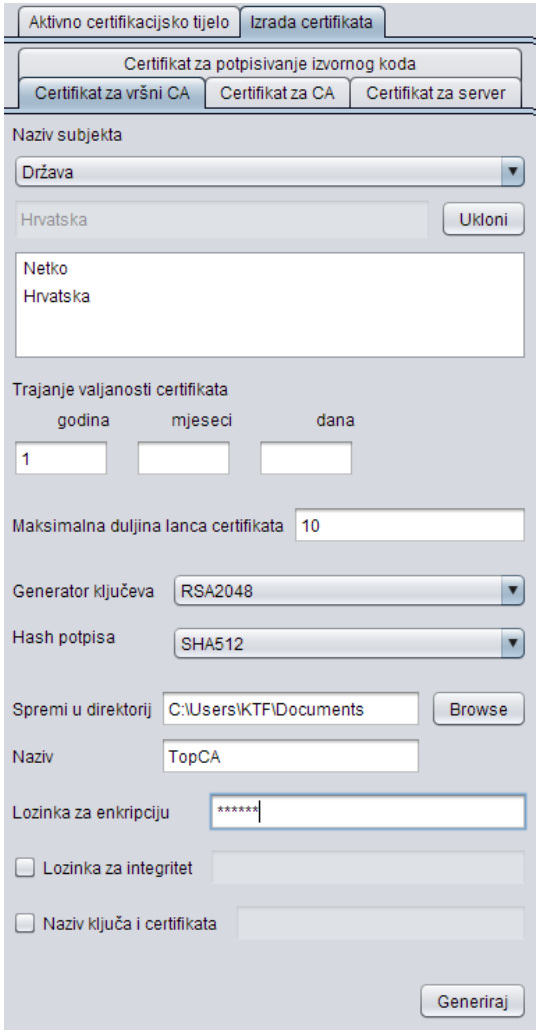
1. Na grafičkom sučelju aplikacije se odabirom tabova „Izrada certifikata“ pa „Certifikat za vršni CA“ dobiva izgled prikazan na slici 9.
2. Potrebno je unijeti sve nužne podatke u grafičko sučelje. Mogući izgled sučelja nakon ovog koraka je prikazan na slici 10.
3. Pritisnuti gumb „Generiraj“.

4. Pričekati pojavu prozora sa slike 11. Također je moguće da će se pojaviti prozor koji javlja grešku ako certifikat nije uspješno izrađen.

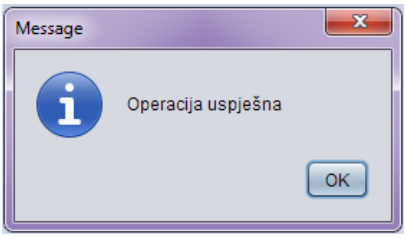
Uspješnim provođenjem ovog scenarija su se stvorile dvije datoteke *.p12 i *.crt koje sadrže stvoreni certifikat i privatni ključ.



Slika 9. Tab za kreiranje certifikata vršnog certifikacijskog tijela



Slika 10. Vršni certifikat spreman za izradu



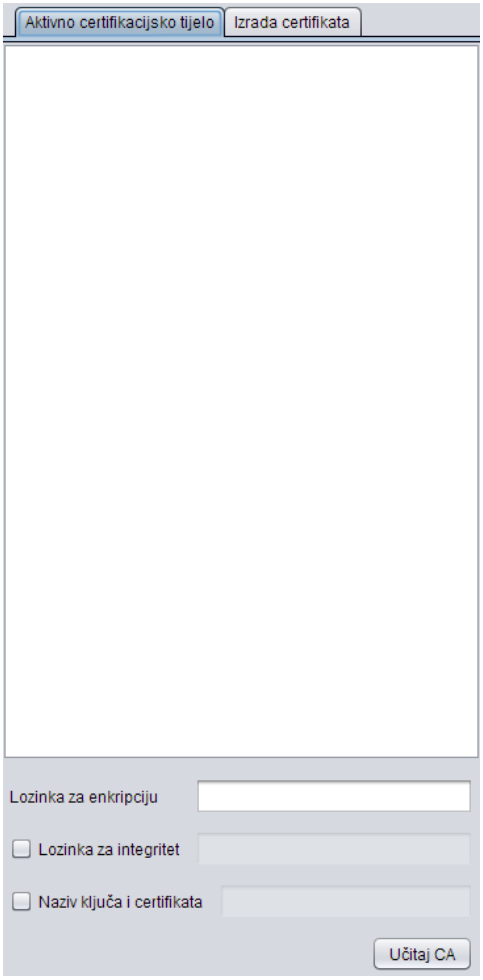
Slika 11. Obavijest o uspješnoj operaciji

3.2 Učitavanje aktivnog certifikacijskog tijela

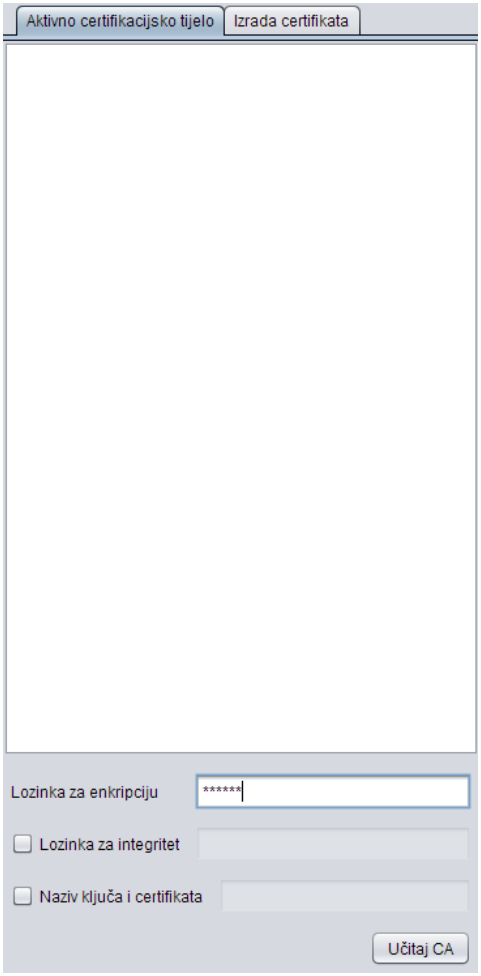
U ovom scenariju se učitava certifikat i privatni ključ iz PKCS#12 datoteke koji će služiti kao lokalno certifikacijsko tijelo koje će biti izdavač certifikata koji će se nakon ovoga stvarati. Postupak:

1. Na grafičkom sučelju aplikacije treba odabrati tab „Aktivno certifikacijsko tijelo“. Izgled je vidljiv na slici 12.
2. Unijeti potrebne lozinke i naziv para ključa i certifikata. Izgled je vidljiv na slici 13.
3. Učitati datoteku.

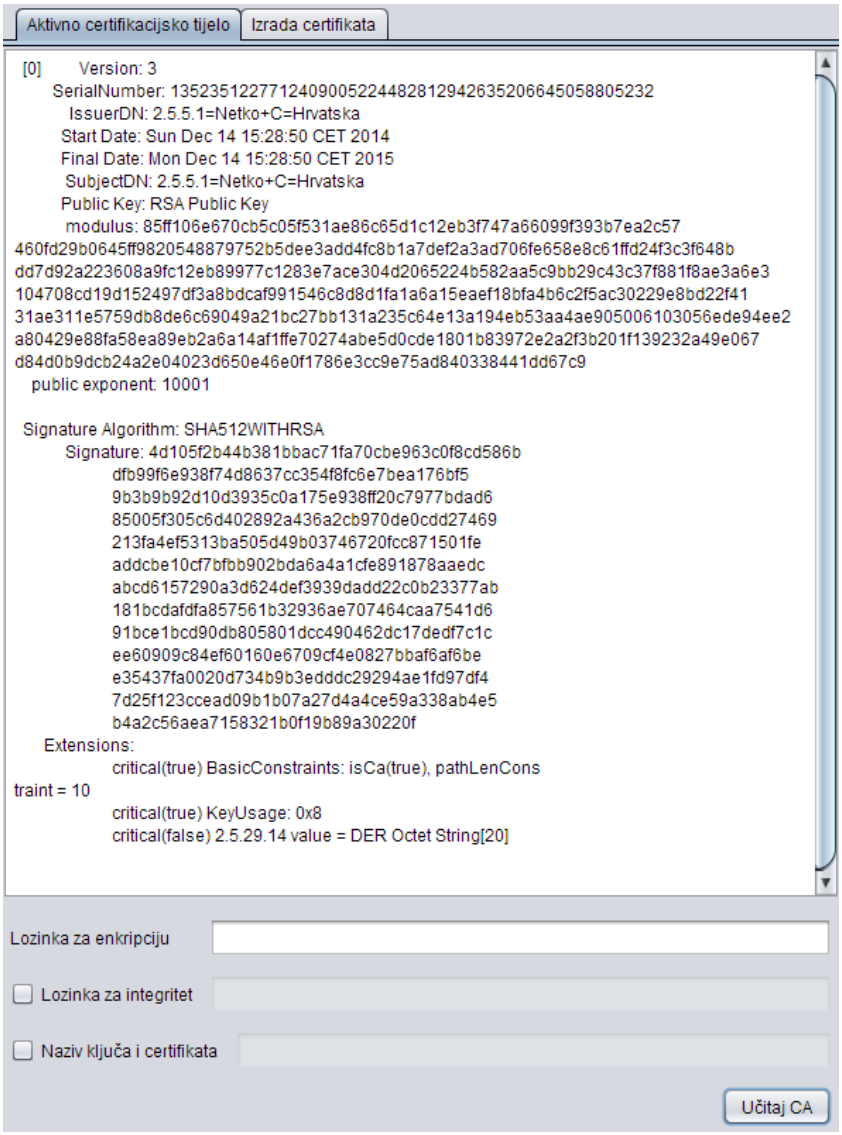
Ako su certifikat i ključ uspješno učitani prikazat će se u tabu „Aktivno certifikacijsko tijelo“. Mogući izgled tog taba nakon ovog scenarija je vidljiv na slici 14.



Slika 12. Tab za učitavanje CA



Slika 13. Tab spreman za učitavanje CA



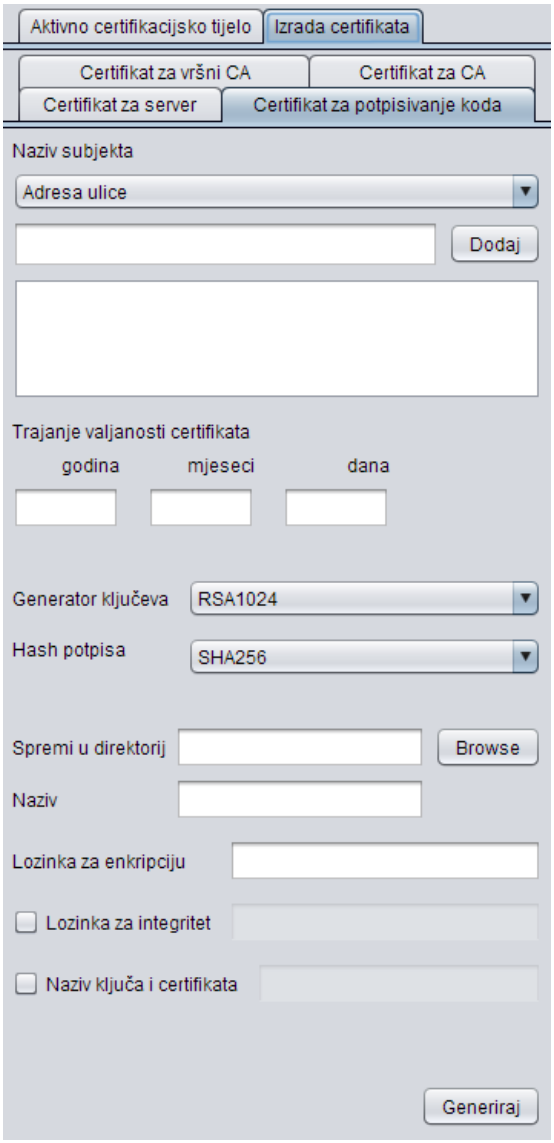
Slika 14. Učitano certifikacijsko tijelo

3.3 Kreiranje certifikata za potpisivanje koda

U ovom scenariju se kreira jedan certifikat za krajnjeg korisnika s namjenom za potpisivanje koda programa. Postupak:

1. Na grafičkom sučelju aplikacije se odabirom tabova „Izrada certifikata“ pa „Certifikat za potpisivanje koda“ dobiva izgled prikazan na slici 15.
2. Potrebno je unijeti sve nužne podatke u grafičko sučelje.
3. Pritisnuti gumb „Generiraj“.
4. Pričekati pojavu prozora sa slike 11. Također je moguće da će se pojaviti prozor koji javlja grešku ako certifikat nije uspješno izrađen.

Uspješnim provođenjem ovog scenarija su se stvorile dvije datoteke *.p12 i *.crt koje sadrže stvoreni certifikat i privatni ključ.



Slika 15. Izgled taba za izradu certifikata za potpisivanje koda

4. Literatura

1. D. Cooper, S. Santesson, S. Farell, S. Boeyen, R. Housley, W. Polk, NIST, Microsoft, Trinity College Dublin, Entrust, Vigil Security; RFC5289; svibanj 2008; *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*; https://tools.ietf.org/html/rfc5280
2. P. Yee, AKAYLA; RFC6818; siječanj 2013; *Updates to the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profil;* https://tools.ietf.org/html/rfc6818