Elektrotehnički fakultet, Univerzitet u Beogradu

Projekat iz Zaštite podataka

Grupa 2

Autori:

Dunja Ćulafić 2016/0236

Lazar Lazić 2016/0245

Jun 2020.

U projektnom zadatku je implementiran PGP protokol poštovanjem OpenPGP strukture poruke definisane dokumentom RFC-4880. Za intuitivno korišćenje napravljen je GUI. Celokupan zadatak realizovan je u Java programskom jeziku.

Algoritmi

U projektnom zadatku korišćeni su sledeći algoritmi:

1. SHA-1
   - U pitanju je algoritam za heširanje koji pravi heš veličine 160b. Koristi se prilikom potpisivanja i prilikom enkripcije privatnog ključa u prstenu privatnih ključeva za heširanje lozinke.
2. AES-256
   - U pitanju je simetričan algoritam i korišćen je u svrhu enkripcije privatnog ključa u prstenu privatnih ključeva.
3. DSA sa ključevima veličine 1024b i 2048b
   - U pitanju je asimetričan algoritam koji je korišćen za potpisivanje poruka. Baziran je na modularnoj eksponencijaciji. Za heširanje se koristi SHA-1. Korisnik bira veličinu ključa.
4. ElGamal sa ključevima veličine 1024b, 2048b i 4096b
   - U pitanju je asimetričan algoritam koji je korišćen za enkripciju ključa sesije, tj. ključa koji se koristi za enkripciju poruke simetričnim algoritmom. Ne treba mešati algoritam sa ElGamal šemom za potpisivanje. Takođe je baziran na modularnoj aritmetici. Korisnik bira veličinu ključa.
5. TripleDES sa EDE konfiguracijom i tri ključa
   - U pitanju je simetričan algoritam koji je korišćen za enkripciju poruke koja se šalje. Podrazumeva trostruko izvršavanje DES algoritma redosledom enkripcija-dekripcija-enkripcija, gde se u svakom koraku koristi različiti ključ. Svaki ključ je dužine 56b, ukupno 168b.
6. IDEA
   - U pitanju je simetričan algoritam koji je korišćen za enkripciju poruke koja se šalje. Veličina ključa je 128b, a veličina bloka je 64b.
7. ZIP
   - U pitanju je algoritam koji služi za kompresiju podataka.

Realizovane klase

Sve realizovane klase se nalaze u paketu etf.openpgp.cd160236dll160245d.

1. DSAElGamalKeyRingGenerator
   - Klasa služi za generisanje novog para ključeva DSA za potpisivanje i ElGamal za enkripciju ključa sesije. Takođe smešta novogenerisane ključeve u prsten privatnih ključeva.
   - Metode:
     1) **void** generateKeyPair(String identity, String passPhrase, **int** DSAKeyLength, **int** ElGamalKeyLength) – javna metoda koja generiše ključeve i poziva exportKeyPair().
     2) **void** exportKeyPair(KeyPair dsaKp, KeyPair elgKp, String identity, **char**[] passPhrase, String path) – privatna metoda koja čuva generisane ključeve u prsten privatnih ključeva.
2. IncorrectPassPhraseException
   - Klasa koja se izvodi iz klase Exception i služi za bacanje izuzetka prilikom lošeg unosa lozinke za pristup privatnom ključu.
3. Main
   - Klasa iz koje se pokreće aplikacija.
   - Metode:
     1) **void** main(String[] args) – metoda koja pravi osnovni izgled aplikacije (GUI) i poziva odgovarajuće metode ostalih klasa radi obrade korisničkih zahteva.
4. PublicKeys
   - Klasa koja služi za rad sa ključevima u prstenu javnih ključeva.
   - Metode:
     1) **void** listPublicKeys(JPanel panel) – javna metoda koja služi za vizuelni prikaz ključeva iz prstena javnih ključeva korisnika.
     2) **void** exportPublicKey() – javna metoda koja služi za izvoz odabranog javnog ključa na odabranu destinaciju.
     3) **void** importPublicKey() – javna metoda koja služi za uvoz javnog ključa sa odabrane destinacije i njegovo smeštanje u prsten javnih ključeva.

4) **void** refreshPublicKeysPanel() – javna metoda koja služi za osvežavanje prikaza javnih ključeva u aplikaciji.

5. ReceiveMessage
   - Klasa koja služi za prijem poruke.
   - Metode:
     1) **void** receive() – javna metoda koja služi za učitavanje poruke sa odabrane destinacije. Poziva metodu decryptVerify() radi dalje obrade poruke.
     2) **boolean** verifyFile(String filePathTo, Object passedObject, JcaPGPObjectFactory pgpFact) – privatna metoda koja služi za verifikaciju potpisa.
     3) **void** decryptVerify(String filePathFrom, String filePathTo) – javna metoda koja radi dekripciju, dekompresiju i poziva metodu verifyFile() po potrebi. Takođe smešta „otpakovanu“ poruku na željenu destinaciju.
6. SecretKeys
   - Klasa koja služi za rad sa ključevima u prstenu privatnih ključeva.
   - Metode:
     1) **void** listSecretKeys(JPanel panel) – javna metoda koja služi za vizuelni prikaz ključeva iz prstena privatnih ključeva korisnika.
     2) **boolean** generateNewKeyPairDialog() – javna metoda koja služi za vizuelni prikaz i odabir parametara pri generisanju novog para ključeva. Poziva metodu DSAElGamalKeyRingGenerator.generateKeyPair().
     3) String enterPassPhraseDialog() – javna metoda koja služi za vizuelni prikaz i unos lozinke.
     4) **void** refreshSecretKeysPanel() – javna metoda koja služi za osvežavanje prikaza privatnih ključeva korisnika u aplikaciji.
     5) **void** removeKeyPair() – javna metoda koja služi za uklanjanje para ključeva iz prstena privatnih ključeva.
     6) **void** exportKeyPair() – javna metoda koja služi za izvoz privatnih ključeva iz prstena privatnih ključeva na odabranu destinaciju.
     7) PGPPrivateKey extractPrivateKey(PGPSecretKey pgpSecKey, **char**[] passPhrase) – javna metoda koja služi za dohvatanje privatnog ključa

iz prstena privatnih ključeva uz njegovu dekripciju i proveru validnosti lozinke.

8) **void** importKeyPair() – javna metoda koja služi za uvoz privatnog ključa sa odabrane destinacije i njegovo smeštanje u prsten privatnih ključeva.

7. SendMessage
   - Klasa koja služi za slanje poruke.
   - Metode:
     1) **void** send() – javna metoda koja služi za vizuelni prikaz opcija za slanje poruke. Poziva po potrebi odgovarajuće metode ove klase.
     2) **byte**[] readMessageFromFile(File file) – javna metoda koja služi za čitanje sadržaja fajla.
     3) **void** signFile(String fileName, PGPSecretKey pgpSec, OutputStream out, **char**[] pass, **boolean** shouldZIP) – privatna metoda koja služi za potpisivanje.
     4) **byte**[] compressFile(String fileName, **int** algorithm) – privatna metoda koja služi za komprimovanje.
     5) **void** encryptFile(OutputStream out, String fileName, PGPPublicKeyRing[] encKeys, **boolean** withIntegrityCheck, **int** symmetricAlgorithm) – privatna metoda koja služi za enkripciju.