Zaštita podataka 2022

Autori:

Aleksandra Milović 2018/0126  
Dobrosav Vlašković 2018/0005

Projekat omogucava bezbednu razmenu mail-ova putem pgp protokola. Korišćeni algoritmi za generisanje parova ključeva i podključeva su DSA(Digital Signature Algorithm) i ElGamal algoritam respektivno. U konfiguraciji 1024 ili 2048 bita za DSA, 1024, 2048 i 4096 bita za ElGamal algoritam. Dok se kod simetričnih algoritama 3DES sa EDE konfiguracijom i tri kljuca i AES algoritam sa ključem veličine 128 bita.

Pretty Good Privacy (PGP) - protokol za [šifrovanje i dešifrovanje](https://sr.m.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%98%D0%B0) podataka koji obezbeđuje kripto privatnost i autorizaciju za prenos podataka. Često se koristi za potpisivanje, šifrovanje i dešifrovanje [tekstova](https://sr.m.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82), imejlova, fajlova, imenika i ceo disk particije i da se poveća bezbednost imejl komunikacije. Ovaj softver je kreirao [Phillip R. Zimmermann](https://sr.m.wikipedia.org/w/index.php?title=Phillip_R._Zimmermann&action=edit&redlink=1) 1991. godine.

DES ([engl.](https://sr.m.wikipedia.org/wiki/%D0%95%D0%BD%D0%B3%D0%BB%D0%B5%D1%81%D0%BA%D0%B8_%D1%98%D0%B5%D0%B7%D0%B8%D0%BA" \o "Engleski jezik) *Data Encryption Standard*) [algoritam](https://sr.m.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%B0%D0%BC) je najviše korišćen [algoritam](https://sr.m.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%B0%D0%BC) za kriptovanje na svetu. Dugo godina, DES je među ljudima bio sinonim za sigurno [šifrovanje](https://sr.m.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%98%D0%B0). Uprkos naporu *Electronic Frontier* fondacije u pravljenju mašine za razbijanje DES-šifrovanih poruka, vredne 250.000 američkih dolara.

AES ([engl.](https://sr.m.wikipedia.org/wiki/%D0%95%D0%BD%D0%B3%D0%BB%D0%B5%D1%81%D0%BA%D0%B8_%D1%98%D0%B5%D0%B7%D0%B8%D0%BA" \o "Engleski jezik) *Advanced Encryption Standard*, napredni standard za enkripciju) je specifikacija za [enkripciju](https://sr.m.wikipedia.org/wiki/Enkripcija) elektronskih podataka. Usvojen je od strane [Vlade SAD](https://sr.m.wikipedia.org/w/index.php?title=%D0%A1%D0%B0%D0%B2%D0%B5%D0%B7%D0%BD%D0%B0_%D0%B2%D0%BB%D0%B0%D0%B4%D0%B0_%D0%A1%D1%98%D0%B5%D0%B4%D0%B8%D1%9A%D0%B5%D0%BD%D0%B8%D1%85_%D0%94%D1%80%D0%B6%D0%B0%D0%B2%D0%B0&action=edit&redlink=1) i koristi se širom sveta. AES je zamenio prethodno korišćeni standard [DES](https://sr.m.wikipedia.org/wiki/DES). AES je [algoritam simetričnog ključa](https://sr.m.wikipedia.org/w/index.php?title=%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%B0%D0%BC_%D1%81%D0%B8%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D1%87%D0%BD%D0%BE%D0%B3_%D0%BA%D1%99%D1%83%D1%87%D0%B0&action=edit&redlink=1), što znači da se isti ključ koristi i za enkripciju i za dekripciju podataka

DSA funkcionisanje

**Key Generation**

* Choose an *L*-bit prime *p*, where 512 <= *L* <= 1024, and *L* is divisible by 64
* Choose a 160-bit prime *q*, such that *p* - 1 = *qz*, where *z* is any natural number
* Choose *h*, where 1 < *h* < *p* - 1 such that *g* = *h*z mod *p* > 1
* Choose *x* by some random method, where 0 < *x* < *q*
* Calculate *y* = *gx* mod *p*
* Public key is (*p*, *q*, *g*, *y*). Private key is *x*

Note that (*p*, *q*, *g*) can be shared between different users of the system, if desired

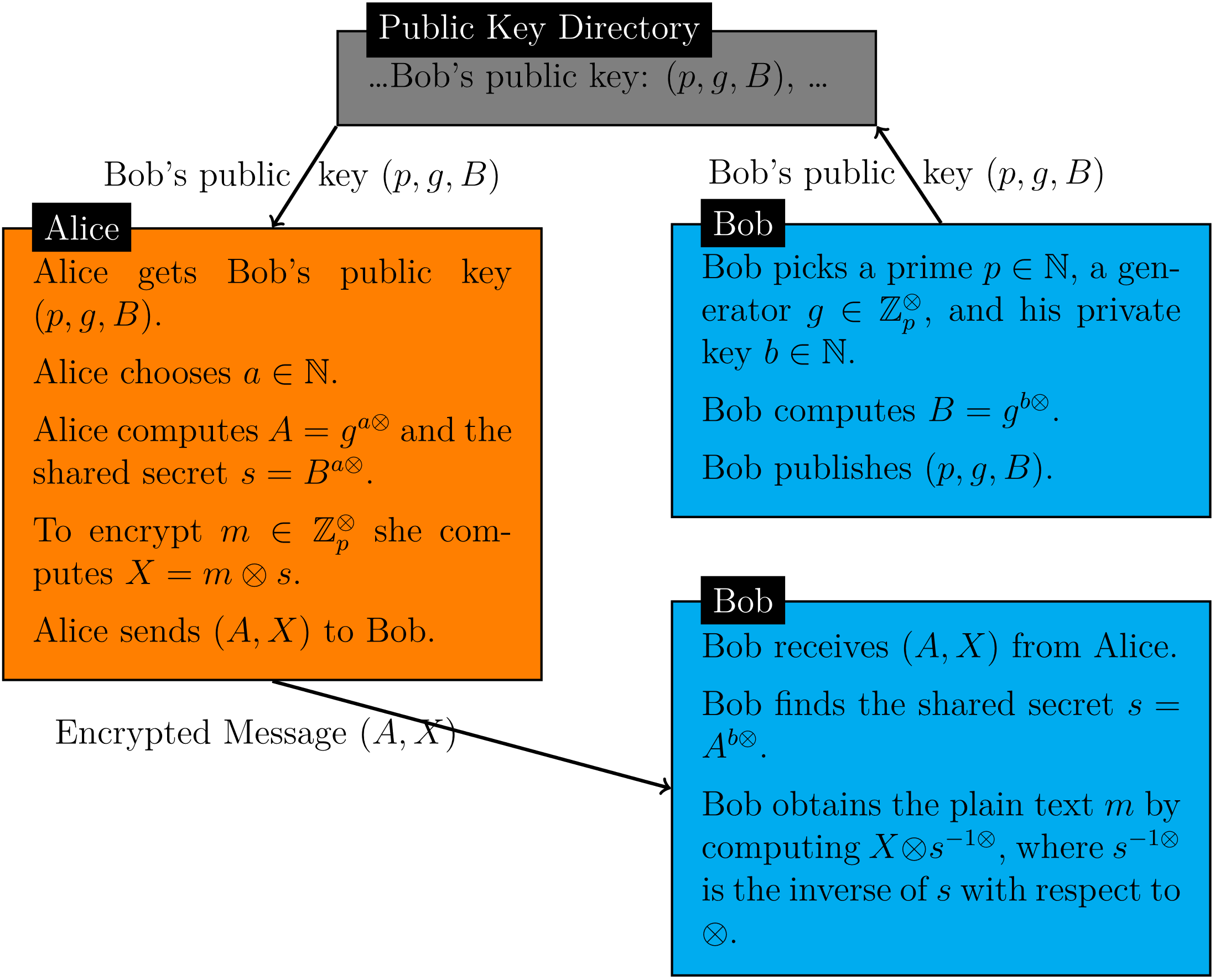
**Signing**

* Choose a random per message value *s* (called a [nonce](http://www.library.snls.org.sz/archive/doc/wikipedia/wikipedia-terodump-0.1/tero-dump/wikipedia/no/Nonce.html)), where 1 < *s* < *q*
* Calculate *s*1 = (*gs* mod *p*) mod *q*
* Calculate *s*2 = (H(*m*) - *s*1\**x*)*s*-1 mod *q*, where H(*m*) is the SHA-1 hash function applied to the message *m*
* Signature is (*s*1,*s*2)

**Verifying**

* Calculate *w* = (*s*2)-1 (mod *q*)
* Calculate *u*1 = H(*m*)\**w* (mod *q*)
* Calculate *u*2 = *s*1\**w* (mod *q*)
* Calculate *v* = [*gu*1\**yy*2 mod *p*] mod *q*
* Signature valid if *v* = *s*1

El Gamal



Implementacija

Klasa KeyManager

Ima sledece metode sa potpisima

loadKeRings služi da na početku izvršavanja programa iz odgovarajucih gpg fajlova procita u i upiše ključeve u odgovarajuću strukturu. Poziva se statički pre početka izvršavanje programa

**private** **static** **void** loadKeyRings()

generateSubPacketVector definiše simetrične algoritme koje ćemo koristiti za šifrovanje pgp paketa, takođe definiše i algoritam za potvrđivanje integriteta.

**private** **static** PGPSignatureSubpacketVector generateKeySubpacketVector()

generateKey vraća par javnog i privatnog smešta u odgovarajuću strukturu i upisuje u gpg fajl, vraća par informacija potrebnih za ispisivanje na gui

**public** **static** **final** Pair<PublicKeyInfo, SecretKeyInfo> generateKeys(User user, KeyMaterial keyMaterial, SubkeyMaterial subkeyMaterial)

Generisanje para kljućeva

**private** **static** **final** KeyPair generateKeyPair(KeyMaterial keyMaterial)

generisanje para podključeva

**private** **static** **final** KeyPair generateSubkeyPair(SubkeyMaterial subkeyMaterial)

Generisanje prstena ključeva

**private** **static** PGPKeyRingGenerator createKeyRingGenerator(User user, KeyPair keyPair, KeyPair subkeyPair)

Vraća kolekciju javnih ključeva

**public** **static** Collection<PublicKeyInfo> getPublicKeyInfoCollection()

vraća kolekkciju privatnih ključeva

**public** **static** Collection<SecretKeyInfo> getSecretKeyInfoCollection()

**public** **static** PGPSecretKeyRing getSecretKeyRing(**long** keyID)

**public** **static** PGPSecretKey getSecretKey(**long** keyId)

**public** **static** PGPPublicKey getPublicKey(**long** keyId)

**public** **static** PGPPublicKey getPublicSubkey(**long** keyIdLong)

Klasa Sender

Ova klasa kontrolise sve oko slanja. Pravi fajlove, određuje output putanju za fajl koji šaljemo. Radi radix64 formatiranje, kompresiju potpisivanje.

Konstruktor klase sender

Prosleđujemo mu sve informacije, koje su mu potrebne za potpisivanje, enkripciju…

**public** Sender(File file, File outputDirectory, **boolean** compressionEnabled, **boolean** radix64Enabled, **boolean** encryptEnabled, **int** symmetricAlgorithmId, PublicKeyInfo publicKeyInfo, **boolean** signEnabled, String passphrase, SecretKeyInfo secretKeyInfo)

Generisanje apsolutne putanje output file-a

**private** String generateOutputFilePath()

Slanje i potpisivanje, enkripcija i sve ovo što sam naveo gore.

**public** **void** send() **throws** FileNotFoundException, RuntimeException

Klasa Receiver

Receiver je zaduzen da primi fajl, da ga dekriptuje, proveri potpis, dekompresuje. Vraća sadržaj fajla u originalno stanje u kojem je bio pre bilo kakve operacije.

Receive metoda vrši osnovne operacije kao što su dešifrovanje. Vrši dekompresiju, proveru validnosti sertifikata

**public** **void** receive()

Klasa ReceiverStatus

Samo vraca i postavlja odgovarajuce statuse za klasu Receiver.

Paket etf.openpgp.ma180126d.exceptions

U njemu se nalaze klase izvedene iz klase Exception. Te klase služe da bi bacale izuzetak u slučaju neke ne regularne situacije kao sto je nevalidan format fajla i mnoge druge.

Klasa KeyInfo

Konstruktor date klase, samo postavlja keyId i userId

**public** KeyInfo(**long** keyId, String userId)