**Elektrotehnički fakultet**

**Univerzitet u Beogradu**



**Projekat iz predmeta Zaštita podataka**

**Implementacija PGP protokola**

Aleksa Bogdanović 2017/0578

Mara Bolić 2017/0614

Beograd, jun 2021.

**Grupa 5**.

**Algoritami za asimetrične ključeve:** DSA za potpisivanje sa ključevima veličine 1024 i 2048 i ElGamal za enkripciju sa kljucevima veličine 1024, 2048 i 4096 bita

**Algoritmi za simetrične ključeve:** 3DES sa EDE konfiguracijom i tri ključa i AES sa ključem veličine 128 bita

**Cilj projektnog zadatka** je bolje razumevanje PGP protokola, mogućnosti koje on pruža i načina čnjegovog korišćenja. U tu svrhu projektovana je i implementirana aplikacija sa grafičkim korisničkim interfejsom u programskom jeziku Java.

**Aplikacija omogućava:**

* Generisanje i brisanje postojećeg para ključeva
* Uvoz i izvoz javnog ili privatnog ključa u .asc format
* Prikaz prstena javnih i privatnih ključeva sa svim potrebnim informacijama
* Slanje poruke (uz obezbeđivanje enkripcije i potpisivanja)
* Primanje poruke (uz obezbeđivanje dekripcije i verifikacije)

Rešenje je implementirano po standardu RFC4480 koji opisuje OpenPGP protokol. Testirano je tako da bude kompatibilno sa aplikacijom Kleopatra. U implementaciji su iskorišćene strukture i algoritmi iz org.bouncycastle.openpgp biblioteke.

**Pregled glavnih klasa, funkcija i implementiranih algoritama**

**Klasa MainForm**

**/\*\***

**\***

**\*/**

Public MainForm()

**/\*\***

**\*** @param args

**\*/**

Public static void main(String [] args)

**/\*\***

**\***

**\*/**

Public void setVisibleMainForm()

**Klasa GenerateKeyForm**

**/\*\***

**\***

**\*** @param email

\* @return

**\*/**

boolean ValidEmail(String email)

**/\*\***

**\***

**\*** @param MainFrame

**\*/**

void initComponents(JFrame MainFrame)

/\*\*

\*

\* @param frame

\*/

public GenerateKeyForm(JFrame frame)

/\*\*

\*

\*/

private void overviewKey()

/\*\*

\*

\* @param keyID

\* @param password

\* @return

\*/

private boolean deleteKeys(String keyID,String password)

**Klasa ReceiveMessageForm**

/\*\*

\*

\* @param MainFrame

\*/

public void initComponents(JFrame MainFrame)

/\*\*

\*

\* @param MainFrame

\*/

public RecieveMessageForm(JFrame MainFrame)

/\*\*

\*

\* @return

\*/

private String importEncryptedMessage()

**Klasa ImportExport**

/\*\*

\*

\* @param MainFrame

\*/

public void init(JFrame MainFrame)

/\*\*

\*

\* @param MainFrame

\*/

public ImportExport(JFrame MainFrame)

/\*\*

\*

\*/

private void exportPublicKeysInComboBox()

/\*\*

\*

\*/

private void exportSecretKeysInComboBox()

/\*\*

\*

\*/

private void exportPublicKey()

/\*\*

\*

\*/

private void exportSecretKey()

/\*\*

\*

\*/

private void importPublicKey()

/\*\*

\*

\*/

private void importSecretKey()

**Klasa KeyRing**

/\*\*

\*

\* @param mainFrame

\*/

private void init(JFrame mainFrame)

\*\*

\*

\* @param mainFrame

\*/

public KeyRing(JFrame mainFrame)

**Klasa SendMessageForm**

/\*\*

\*

\* @param MainFrame

\*/

void initComponent(JFrame MainFrame)

/\*\*

\*

\* @param MainFrame

\*/

public SendMessageFrom(JFrame MainFrame)

**Klasa GenerateKeys**

/\*\*

\*

\* @param dsaParam

\* @param elgParam

\* @param email

\* @param password

\*/

public static void generateKeys(int dsaParam, int elgParam, String email, char [] password)

/\*\*

\*

\* @param dsaParam

\* @param elgParam

\* @param email

\* @param password

\* @return

\*/

public final static PGPKeyRingGenerator generateKeyRingGenerator(int dsaParam, int elgParam, String email, char[] password)

/\*\*

\*

\* @param dsaKeyPair

\* @param elGamalKeyPair

\* @param identity

\* @param passphrase

\* @return

\* @throws Exception

\*/

public static final PGPKeyRingGenerator createPGPKeyRingGenerator(KeyPair dsaKeyPair, KeyPair elGamalKeyPair, String identity, char[] passphrase)

/\*\*

\*

\* @param keySize

\* @return

\* @throws NoSuchAlgorithmException

\* @throws NoSuchProviderException

\*/

public static final KeyPair generateDsaKeyPair(int keySize)

/\*\*

\*

\* @param keySize

\* @return

\* @throws Exception

\*/

public static final KeyPair generateElGamalKeyPair(int keySize)

/\*\*

\*

\* @param paramSpecs

\* @return

\* @throws Exception

\*/

public static final KeyPair generateElGamalKeyPair(ElGamalParameterSpec paramSpecs)

**Klasa EncryptMessage**

/\*\*

\*

\* @param message

\* @param publicKey

\* @param conversion

\* @param compression

\* @param symetricAlg

\* @return

\*/

public static boolean EncryptMessage(String message, PGPPublicKey publicKey, boolean conversion, boolean compression, int symetricAlg)

/\*\*

\*

\* @param message

\* @param name

\* @param publicKey

\* @param secretKey

\* @param privateKey

\* @param conversion

\* @param compression

\* @param encryption

\* @param symetricAlg

\* @return

\*/

public static boolean SignEncryptMessage(String message, String name, PGPPublicKey publicKey, PGPSecretKey secretKey, PGPPrivateKey privateKey, boolean conversion, boolean compression, boolean encryption, int symetricAlg)

/\*\*

\*

\* @param passwd

\* @param o

\*/

public static void DecryptMessage(char [] passwd, Object o)

/\*\*

\*

\* @param o

\*/

public static void ValidateMessage(Object o)