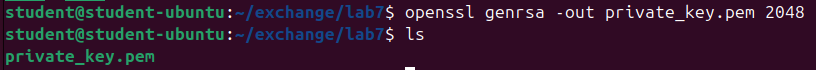
|  |
| --- |
| Politechnika Wrocławska, Informatyka Stosowana |
| ZASTOSOWANIA KRYPTOGRAFII |
| Cyberbezpieczeństwo, Laboratorium nr.6 - raport |

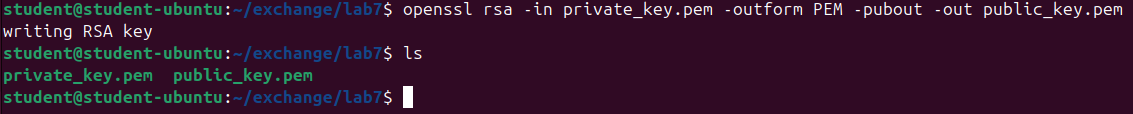
|  |
| --- |
| Autor: Aleksander Stepaniuk  Nr. Indeksu: 272644 |

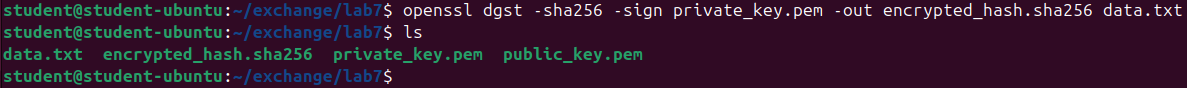
**Zad 1. Zastosowanie kryptografii**

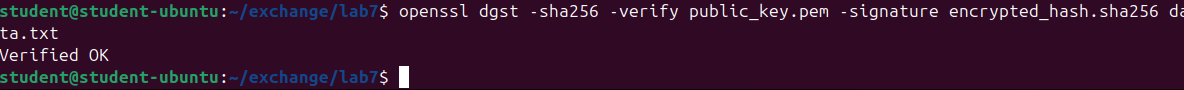
**Zadanie 1.1;**

Przebieg zadań:









Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

**Zadanie 1.2;**

Przebieg zadań:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie



**Zadanie 1.3;**

**Obraz zawierający tekst, Czcionka, zrzut ekranu

Opis wygenerowany automatycznie**

**Zadanie 1.4;**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

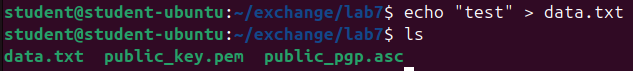
**Obraz zawierający tekst, zrzut ekranu, Czcionka

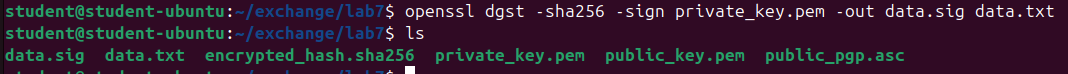
Opis wygenerowany automatycznie**

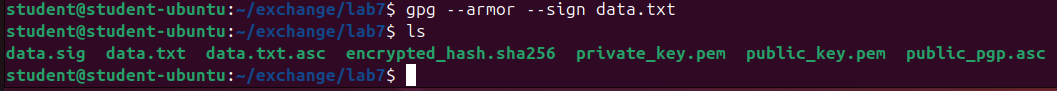
**Obraz zawierający tekst, zrzut ekranu, Czcionka

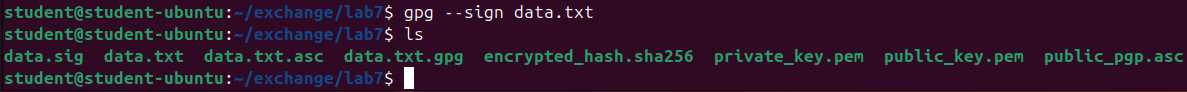
Opis wygenerowany automatycznie**

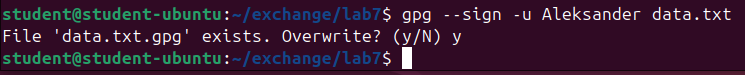
**Zadanie 1.5;**

****

****

****

****

****

**Zadanie 1.6;**

****

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Zadanie 1.7;**

****

**Obraz zawierający tekst, zrzut ekranu, Czcionka

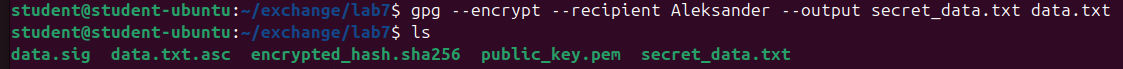
Opis wygenerowany automatycznie**

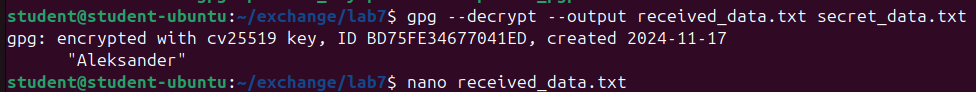
**Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie**

Dla gpg musimy zrobić detach-sign, inaczej zmiana oryginalnego dokumentu nic nie zmieni, bo nie będzie porównywał sygnatury z plikiem data.txt (sygnatura + zawartość pliku będzie zapisana razem w jednym pliku)

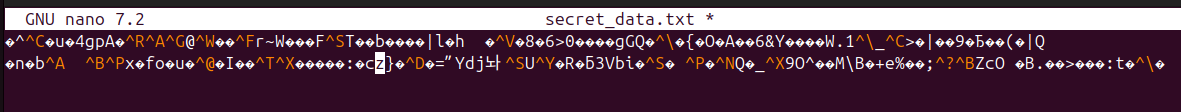
**Zadanie 1.8;**



****

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

****

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Pytanie 1.9;**

Teoretycznie można wygenerować np. tylko klucz prywatny, ale nie ma to sensu, ponieważ klucz publiczny jest niezbędny do szyfrowania wiadomości oraz weryfikacji podpisów. Para kluczy działa tylko i wyłącznie razem przy korzystaniu z komunikacji asymetrycznej.

**Pytanie 1.10;**

Klucz GPG zawiera metadane, takie jak ID użytkownika, adres email, komentarze i jest zapisywany w formacie OpenPGP. Klucz PEM, jest ogólnym formatem do przechowywania kluczy w postaci tekstowej (base64) i nie posiada dodatkowych metadanych, ani funkcjonalności charakterystycznych dla OpenPGP.

**Pytanie 1.11;**

Wymiana kluczy prywatnych jest nieuzasadaniona, a do tego dosyć niebezpieczna, ponieważ klucz prywatny powinien pozostać tajny, aby zapewnić bezpieczeństwo procesu kryptograficznego. Udostępnienie takiego klucza narusza całą zasadę i sens kryptografii asymetrycznej.

**Pytanie 1.12;**

W zadaniu 1.7 podpis cyfrowy stał się niepoprawny po modyfikacji pliku. Wynika to z tego, że podpis związany jest sumą kontrolną z oryginalną treścią i zmiana nawet choćby jednego znaku powoduje, że sumy te nie będą do siebie pasować. Reguła ta nie sprawdza się dla plików, posiadających zapisaną sygnaturę razem z danymi w jednym pliku, bo wtedy oryginalny plik nie ma wpływu na nic, bo została wykonana kopia danych i zapisanie ich razem z podpisem w jednym pliku.

**Pytanie 1.13;**

Odcisk palca to inaczej skrót (hash) klucza publicznego, który umożliwia łatwą i jednoznaczną identyfikację klucza. Służy do weryfikacji, czy klucz nie został zmodyfikowany albo podmieniony.

**Pytanie 1.14;**

Nie udało się odszyfrować pliku po zmianie jednego znaku w zaszyfrowanej treści. Algorytmy asymetryczne takie jak RSA są odporne na modyfikacje danych, więc uszkodzony szyfrogram skutkuje błędem podczas procesu odszyfrowywania.

**Pytanie 1.15;**

Dla algorytmu ECC sygnatura jest krótsza niż dla RSA, co wynika z większej efektywności kluczy eliptycznych przy tej samej przyłożonej sile kryptograficznej. Sygnatury ECC są na ogół bardziej wydajne w kontekście zajmowanej przestrzeni i wykonywanych obliczeń.

**Pytanie 1.16;**

Teoretycznie klucz publiczny może podpisać wiadomość może podpisać wiadomość, ale jest to sprzeczne z założeniami kryptografii asymetrycznej. Wynikiem takiego działania byłoby to, że każdy kto posiada klucz publiczny (czyli każdy, bo klucz jest publiczny) mógłby tworzyć takie podpisy, co prowadziłoby do braku zaufania i możliwości podszywania się pod kogokolwiek. Klucze asymetryczne są jednak matematycznie powiązane, więc możemy technicznie zatrzymać klucz publiczny jako ten „tajny” a udostępnić klucz prywatny i system technicznie będzie działać.

**Zad 2. Utwórz CA**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

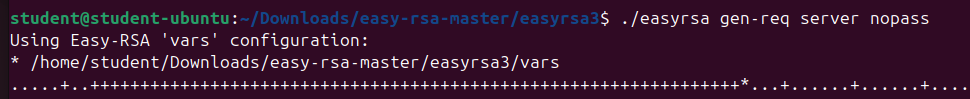
Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Zad 3. Wygeneruj certyfikaty serwera i klienta**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznieObraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie**

**Pytanie 3.1;**

Algorytm podpisu w certyfikatach zazwyczaj jest określony w polu "Signature Algorithm" i zależy od konfiguracji. Często używane są algorytmy takie jak SHA256 wraz z RSA. W powyższym zadaniu używanym algorytmem podpisu jest „*sha256WithRSAEncryption”.*

**Pytanie 3.2;**

Algorytm klucza publicznego zależy od klucza użytego podczas generowania certyfikatu. Dla RSA jest to *rsaEncryption*, a dla ECC na przykład: *id-ecPublicKey*. W zadaniu nr. 2 jest to *rsaEncryption.*

**Pytanie 3.3;**

Easy RSA i większość współczesnych podobnych do niego narzędzi używa standardu X.509 w wersji 3 (X.509v3), co widać na samej górze certyfikatu (*Version: 3 (0x2)*)

**Pytanie 3.4;**

Z certyfikatu można odczytać między innymi: nazwę podmiotu (subject), nazwę wystawcy (Issuer), daty ważności, numer seryjny, rozszerzenia, długość oraz typ klucza publicznego