Dokumentacja projektu

Bezpieczeństwo Systemów Komputerowych

Produkt: Szyfrator plików wraz z przekazaniem klucza sesyjnego
Autor: Mikołaj Szotowicz, 155208

Agenda:

1. Produkt

a. Opis

Wytworzonym produktem w ramach projektu z przedmiotu Bezpieczeństwo Systemów Komputerowych jest oprogramowanie pozwalające na zaszyfrowanie dowolnego pliku znajdującego się na komputerze, a także na odszyfrowanie pliku wynikowego ww. programu. Ponadto podczas szyfrowania wybieramy odbiorców zdolnych na odszyfrowanie. Używamy ku temu ich kluczy publicznych, zatem tylko właściciel klucza prywatnego (do pary z użytym kluczem publicznym) i znający hasło użyte do zaszyfrowania tegoż klucza prywatnego jest w stanie odszyfrować plik.

Dzięki połączeniu wielu rozwiązań technologii kryptograficznej, m.in. algorytmy szyfrujące, klucz sesyjny, para kluczy, funkcje skrótu, program okazuje się potencjalnie bezpieczny do tego typu operacji.

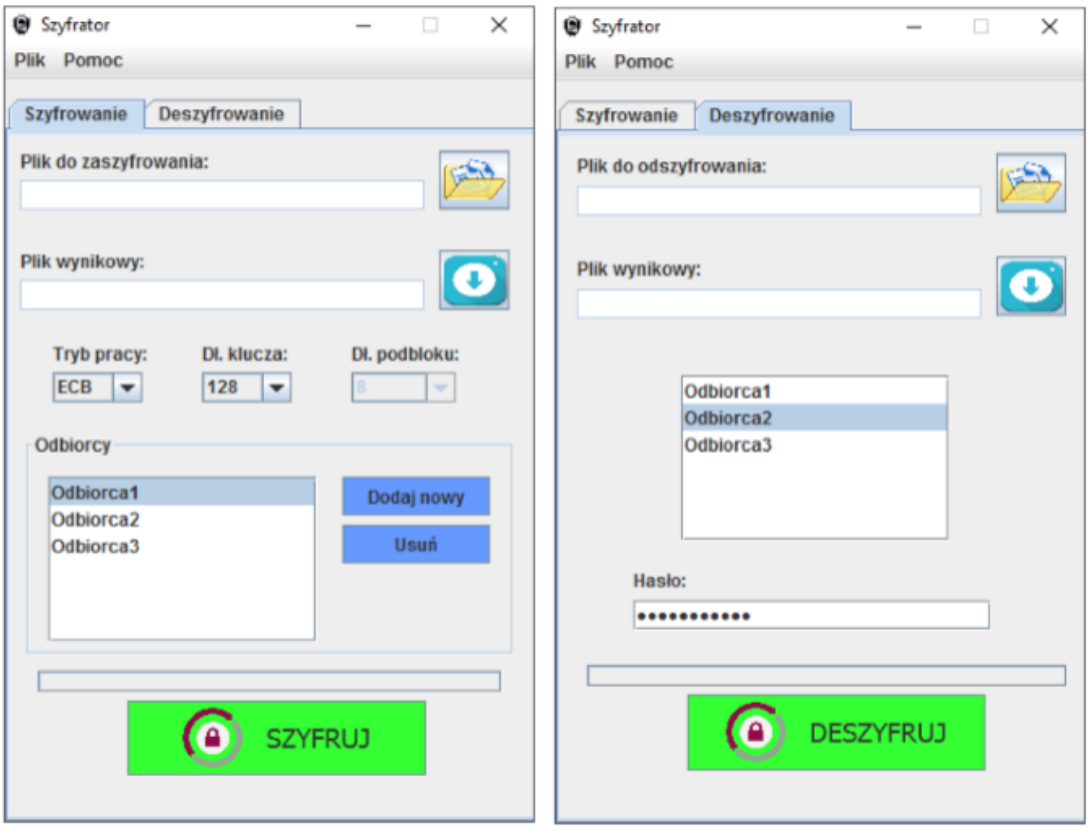
b. Ustawienia szyfrowania

Szyfrator pozwala na prace w czterech trybach pracy (ECB, CBC, CFB oraz OFB), każdorazowo z innym kluczem sesyjnym, co za tym idzie poszczególne szyfrogramy zawsze będą się różniły, nawet podczas pracy na tym samym pliku wejściowym.

Program pozwala również na wybór długości klucza sesyjnego, użytkownik ma do wyboru jedną z trzech opcji: 128, 192 i 256 bitów, ponadto przy wyborze trybu szyfrowania CFB lub OFB możliwy jest wybór długości podbloku z zakresu od 8 do 128 bitów.

2. Projekt interfejsu

Interfejs użytkownika miał być z założenia intuicyjny i łatwy w obsłudze, nawet dla osób średnio zaawansowanych technicznie. Tym samym każde niepożądane działanie programu jest niezwłocznie i w sposób klarowny informowane.



Rys. 1. Interfejs użytkownika

Powyższy rysunek (Rys. 1) przestawia interfejs użytkownika. Po lewej znajduje się panel do szyfrowania, zaś po prawej do odszyfrowywania. O pracy programu świadczy także pasek postępu znajdujący się nad przyciskiem potwierdzającym wybrane zadanie.

W panelu szyfrowania oprócz dostępnych ustawień (patrz pkt. 1.b) oferuje dodanie odbiorcy z katalogu domyślnego przechowującego klucze publiczne, a także po błędnym wyborze na usunięcie niechcianego odbiorcy.

W panelu odszyfrowania po wyborze pliku wejściowego (do odszyfrowania) ukazuje nam się lista dostępnych odbiorców pliku. Wybieramy swój identyfikator, następnie wpisujemy hasło używane do szyfrowania klucza prywatnego. Program sam lokalizuje klucz prywatny po wyborze odpowiedniego identyfikatora odbiorcy.

Należy pamiętać, że każdorazowo mamy możliwość wybrania lokalizacji, nazwy, a nawet i rozszerzenia pliku wynikowego.

3. Technologia

Środowisko programistyczne
Program został napisany za pomocą otwartego oprogramowania IDE „NetBeans" w języku programowania Java. Do stworzenia interfejsu użytkownika wykorzystano wbudowaną bibliotekę graficzną „Swing". Ponadto do celów kryptograficznych skorzystano z zewnętrznej biblioteki „Bouncy Castle", również na licencji Open Source.
Algorytm szyfrujący
• Szyfrowanie pliku wejściowego odbywa się za pomocą symetrycznego szyfru blokowego **RC6**. Algorytm ten obsługuje bloki o długości 128-bitów, oraz klucze o długości 128, 192 lub 256 bitów. Hasłem szyfrującym jest klucz sesyjny. • Klucze prywatne są zaszyfrowane z użyciem ww. szyfru, w trybie ECB, gdzie hasłem szyfrującym jest indywidualne hasło posiadacza pary kluczy, zapisane za pomocą funkcji skrótu. • Klucz sesyjny zaszyfrowany jest asymetrycznym algorytmem kryptograficznych z kluczem publicznym (**RSA**) zamierzonego odbiorcy.
Generowanie klucza sesyjnego
Do wygenerowania klucza sesyjnego wykorzystywany jest silny generator liczb klasy kryptograficznej **SecureRandom**, pracujący w trybie losowym – co nieco wydłuża ich generowanie, jednak sprawia, że klucze stają się nie do przewidzenie (generowanie nieliniowe). Sam mechanizm jako wartość początkową (ziarno) pobiera dane z otoczenia, jednak w celach bezpieczeństwa nie jest znany dokładny czynnik, co jeszcze bardziej uskutecznia jego prace.
Funkcja skrótu
Klucze prywatnie są przechowywane w postaci zaszyfrowanej przez RC6 w trybie ECB, a kluczem szyfrującym jest skrót hasła. Użyto do tego funkcji skrótu **SHA-256**.

4.Struktura pliku

a. Plik wynikowy

Plik wynikowy składa się z nagłówka XML-owego oraz szyfrogramu (Rys. 2). W nagłówku przechowywane są informacje przekazywane odbiorcy:

Element	Opis
<EncryptedFile>	Znacznik nagłówka XML (root)
<Algorith>	Nazwa użytego algorytmu do zaszyfrowania pliku.
<BlockSize>	Długość bloku obsługiwanego przez algorytm. RC6 obsługuje 128 bitowe bloki.
<CipherMode>	Tryb szyfrowania

<KeySize>	Długość klucza
<SubblockSize>	Długość podbloku wyrażana w bitach (Opcjonalny element, wyłącznie podczas użycia trybu CFB lub OFB)
<IV>	Długość wektora początkującego wyrażana w bitach. (Opcjonalny element, wyłącznie podczas użycia trybu CBC, CFB lub OFB)
<ApprovedUsers>	Znacznik listy odbiorców
<User>	Znacznik poszczególnego odbiorcy
<Name>	Nazwa odbiorcy/identyfikator
<SessionKey>	Klucz sesyjny w postaci zaszyfrowanej

Rys. 2. Fragment pliku wynikowego

Poniżej nagłówka XML-owego znajduję się fragment szyfrogramu. Jest zapisany w postaci czystych bajtów.

b. Klucz publiczny

Klucz publiczny przechowywany jest w plikach z rozszerzeniem „.key”. Ma postaci strukturalnej XML (Rys. 3). Przykładowe klucze publiczne dołączone są do programu w katalogu lib > public.

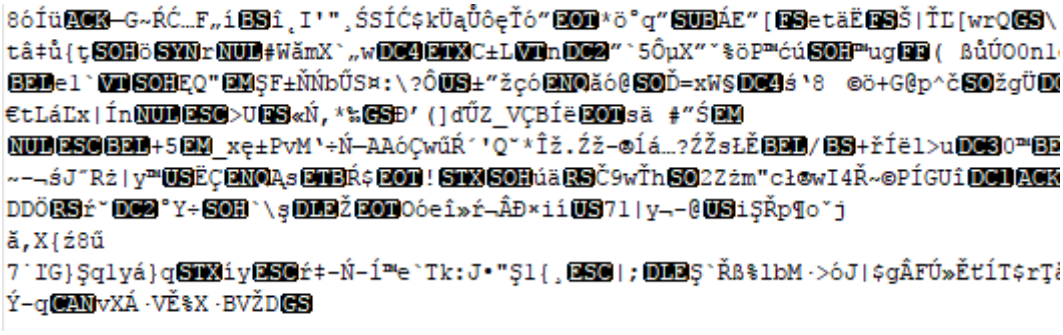
Element	Opis
<User>	Znacznik nagłówka XML (root)
<Name>	Nazwa właściciela klucza
<Email>	Adres email właściciela klucza
<KeyType>	Typ klucz
<RSAKeyValue>	Znacznik określający cel klucza – szyfrowanie algorytmem RSA
<Exponent>	Wykładnik szyfrujący
<Modulus>	Moduł, klucz publiczny do pary do prywatnego

```
<User>
    <Name>odbiorca1</Name>
    <Email>odbiorca1@wp.pl</Email>
    <KeyType>Public</KeyType>
    <RSAKeyValue>
        <Exponent>AQAB</Exponent>
        <Modulus>
            MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsetJLb6uEZU3kwHjG
        </Modulus>
    </RSAKeyValue>
</User>
```

Rys. 3. Fragment klucza publicznego

c. Klucz prywatny

Klucz prywatny przechowywany jest w plikach z rozszerzeniem „.key”. Jest zapisany w postaci czystych bajtów (Rys. 4).



Rys. 4. Fragment Klucza prywatnego

W załączonym do dokumentacji programie w katalogu lib > private znajdują się przykładowe klucze prywatne. Klucze są zaszyfrowane z użyciem RC6. Hasła do kluczy mają taką samą nazwę co nazwy kluczy.

Przykład:
Plik o nazwie „odbiorca1.key” to klucz należący do użytkownika „odbiorca1”, a hasło to „odbiorca1”.

5. Uruchomienie dołączonego programu

Plik wykonywalny programu znajduję się w dołączonym na płycie katalogu „Program" o nazwie „Szyfrator.jar". Uruchomienie programu niczym nie różni się od zwykłe uruchomienia, tj. mamy dwie możliwości: podwójne kliknięcie lewym przyciskiem myszy na plik lub prawym i wybranie opcji z menu kontekstowego „Otwórz"[1].

Do prawidłowej pracy programu niezbędne jest prawidłowo zainstalowane środowisko Java. Aby sprawdzić to w systemie Windows należy:

1. Menu podręcznego wybrać pozycję Panel sterowania.
2. W wyświetlonym Panelu sterowania wybrać kategorię Programy.
3. Wybrać opcję Programy i funkcje.
4. Na liście są uwzględniane zainstalowane wersje oprogramowania Java.

6. Testy

Przykładowe testy wydajnościowe programu.

Maszyna testująca	Asus R510L Procesor: Intel Core i5-4200 RAM: 8GB Rodzaj dysku: HDD Karta graficzna: NVIDIA GeForce GT 720M
System operacyjny	Microsoft Windows 10 Education 64-bit
Plik wejściowy	Plik o rozmiarze 80 MB (.exe)
Czas szyfrowania (ECB, klucz: 128b)	49:06 sekund
Czas odszyfrowania	31:41 sekund
Czas szyfrowania (CBC, klucz: 128b)	50:64 sekund
Czas odszyfrowania	30:58 sekund
Czas szyfrowania (CFB, klucz: 128b, **podblok: 8**)	14:03:71 minut
Czas odszyfrowania	08:31:82 minut
Czas szyfrowania (CFB, klucz: 128b, podblok: 128)	50:98 sekund
Czas odszyfrowania	31:63 sekund
Czas szyfrowania (OFB, klucz: 128b, podblok: 128)	51:92 sekund
Czas odszyfrowania	31:54 sekund

Podane czasy są średnią z 3 pomiarów tego samego pliku i wykonywane wyłącznie dla jednego, tego samego, odbiorcy.

Pojedyncze uruchomienie programu odbyło się na komputerach z systemem operacyjnym:

MacOS Sierra (MacBook Air)
Microsoft Windows 10 Education 64-bit (PC)
Microsoft Windows 10 Home 64-bit (PC)
Microsoft Windows 7 Professional 64-bit (PC)

[1] Instrukcja dotyczy pracy na systemie operacyjnym Microsoft Windows