Ústav informatiky a matematiky

Slovenská technická univerzita v Bratislave

Zadanie č.2

TurboPowerCrypt

Implementácia aplikácie na šifrovanie súborov

Úvod do počítačovej bezpečnosti
2016/2017

This document is classified as confidential

Autor: Michal Drahovský, Ján Dzvoník

1 Overview

1.1 Programovací jazyk a popis

Predstavujeme Vám aplikáciu TurboPowerCrypt na šifrovanie súborov. Je to multi-thread aplikácia vytvorená v programovacom jazyku C#, ktorý umožňuje jednoducho vyvíjať aplikácie s natívnym grafickým používateľským rozhraním pre používateľa operačného systému Windows. Využíva výhody .NET Frameworku. Výhodou programu je jednoduchosť použitia v operačnom systéme Windows. Jeho najväčšou silou je výkon a bezpečnosť.

Aplikácia šifruje vybraný súbor algoritmom Rijndael (AES) v móde CBC. Použitý kľúč sa zašifruje asymetrickou šifru RSA s dĺžkou kľúča 256-bitov. Zašifrovaný kľúč sa prenáša v súbore so zašifrovanými dátami. Pre zašifrovanie kľúča je potrebné použiť verejný kľúč a pre dešifrovanie kľúča (a tým pádom aj celého súboru) odpovedajúci privátny kľúč. Pár kľúčov vlastní iba používateľ, ktorý ich vytvoril. Používateľ môže verejný kľúč doručiť tretej osobe a súbory šifrované jeho verejným kľúčom bude môcť dešifrovať iba on.

Program TurboPowerCrypt implementuje kontrolu integrity údajov metódou HMAC (kľúčovaný autentifikačný hashkód správy). Po šifrovaní súboru sa vypočíta odtlačok správy. Kľúčom pre algoritmus je tajná zmes parametrov, teda bol použitý iný kľúč ako pre šifrovanie dát súboru. Proces dešifrovania sa spustí iba po pozitívnom výsledku porovnania deklarovaného HMAC odtlačku v súbore a vypočítaného HMAC.

1.2 Použité knižnice

Využili sme .NET knižnicu.
Program bol vyvíjaný a testovaný v prostredí s .NET verziou 4.5.2 . Používame štandardné triedy z menných priestorov .Net frameworku. Kryptografické procesy používajú implementáciu z menného priestoru: System.Security.Cryptography:

- RSACryptoServiceProvider
- CspParameters
- RijndaelManaged
- CipherMode
- ICryptoTransform
- CryptoStream
- HMACSHA256
- CryptographicException

1.3 Zdroje

Ako predlohu pre logickú štruktúru TurboPowerCrypt aplikácie a pre používateľské rozhranie sme použili príklad:

<https://msdn.microsoft.com/en-us/library/bb397867(v=vs.110).aspx>

Pri zabezpečení integrity údajov sme vychádzali z dokumentácie a ukážok:

<https://msdn.microsoft.com/en-us/library/system.security.cryptography.hmacsha256(v=vs.110).aspx>

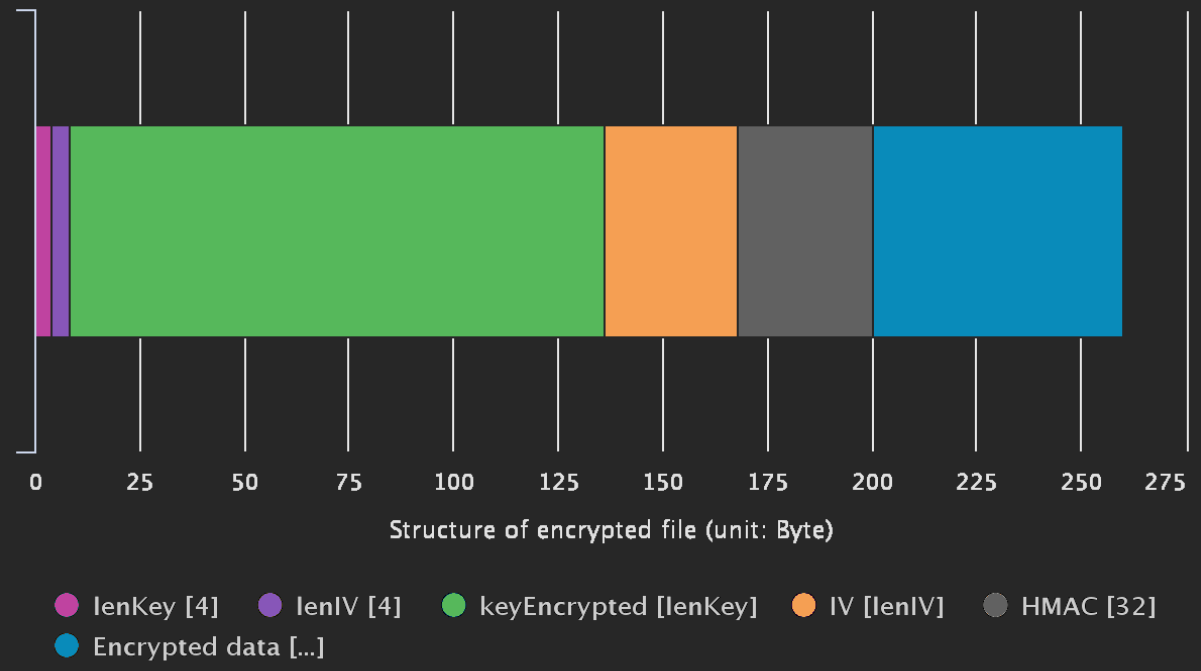
Inšpiráciu pri optimalizácii výkonu a zlepšení UX z aplikácie TurboPowerCrypt sme čerpali z dokumentácie:

<https://msdn.microsoft.com/en-us/library/system.componentmodel.backgroundworker.aspx>

2 Encrypted file

Zašifrovaný súbor obsahuje od začiatku:

- Dĺžka kľúča [4B]
- Dĺžka inicializačného vektora [4B]
- Zašifrovaný kľúč pomocou RSA [{dĺžka kľúča}B] //[128B]
- Inicializačný vektor [{dĺžka IV}B] //[32B]
- HMAC [32B]
- Zašifrované dáta [?B]



3 Power

3.1 Benchmark

Testovanie výkonu aplikácie TurboPowerCrypt prebiehalo šifrovaním a dešifrovaním súboru s veľkosťou 1 GB (1 073 741 824 bajtov).
Test prebiehal na osobnom počítači s parametrami:

- OS: Windows 10 Home
- Procesor: Intel(R) Core(TM) i5-3230M CPU @2.6GHz
- RAM: 8,0 GB
- HDD: ST1000LM024 HN-M101MBB 931.51 GB

V nasledujúcej tabuľke sú zhrnuté 3 výsledky testu na 1GB súbore:

	ŠIFROVANIE		DEŠIFROVANIE	
1.	03.11.2016 00:45:27	46.892 s	03.11.2016 00:47:26	1 min 5.534 s
2.	03.11.2016 00:53:27	47.299 s	03.11.2016 00:57:19	59.768 s
3.	03.11.2016 17:44:27	47.564 s	03.11.2016 17:46:48	58.799 s

Vysoký výkon pri zachovaní priateľskosti k používateľovi sme dosiahli vďaka viacvláknovej architektúre programu a buffrovaným I/O operáciám.

4 Compilation

Program TurboPowerCrypt bol vyvíjaný a zostavený v IDE Microsoft Visual Studio Enterprise 2015 Version 14.0.24720.00 Update 1. Po vytvorení Windows Form projektu volíme Build > Build Solution.

5 Installation

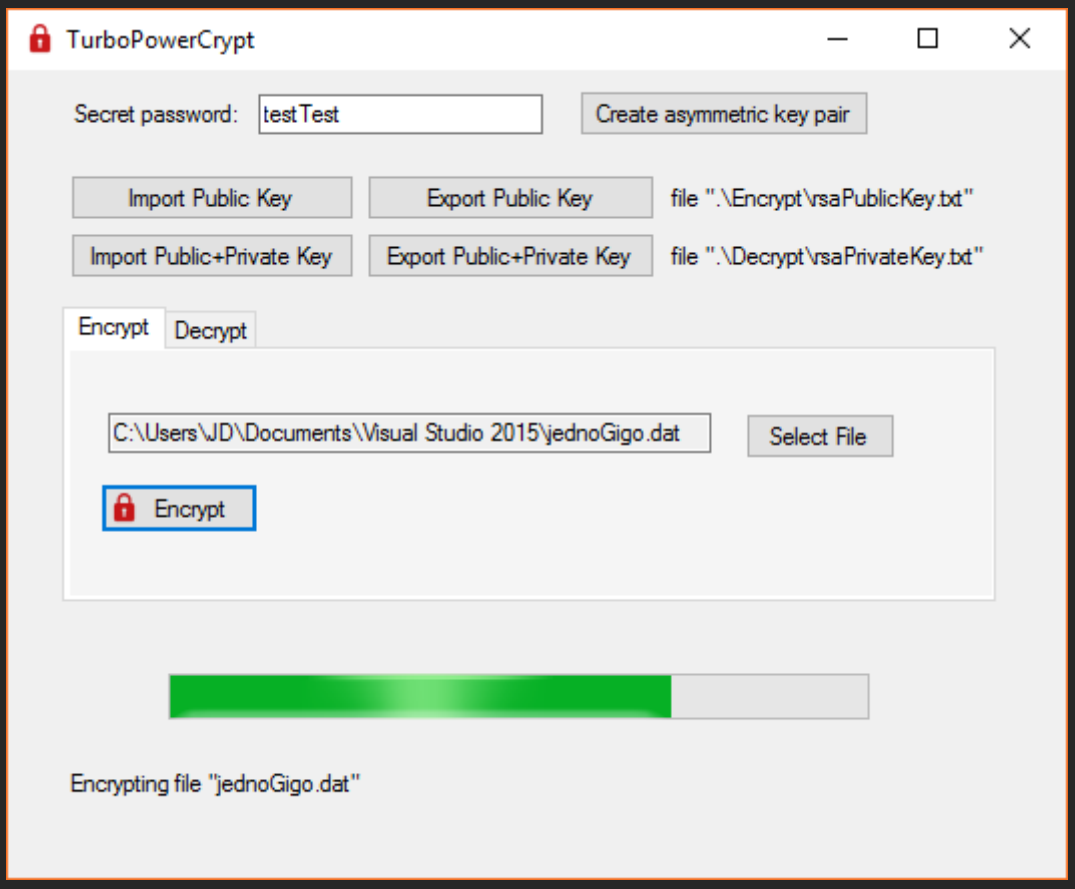
Otvoriť TurboPowerCrypt.exe.
Aplikácia vyžaduje nainštalovaný .NET framework.

6 Use

6.1 GUI prostredie

Pri prvom spustení aplikácie je potrebné vytvoriť pár kľúčov pre asymetrickú šifru. Zadajte heslo pre vytvorenie páru kľúčov a stlačte "Create asymmetric key pair." Po vygenerovaní kľúčov je

možné vyexportovať verejný kľúč a privátny kľúč do automaticky vytvorených priečinkov Encrypt a Decrypt. Verejný kľúč je možné poslať druhému používateľovi, s ktorým chceme bezpečne komunikovať. Ten po spustení aplikácie vloží verejný kľúč do priečinka Encrypt, importuje verejný kľúč do programu, zvolí si súbor a zašifruje ho. Súbor sa uloží do priečinku Encrypt. Zašifrovaný súbor môže odoslať prvému používateľovi. Následne prvý používateľ spustí aplikáciu, importuje svoj privátny kľúč z priečinku Decrypt, vyberie súbor na dešifrovanie a dešifruje ho. Dešifrovaný súbor sa ukladá do priečinku Decrypt.



Create asymmetric key pair - vygeneruje pár asymetrických kľúčov použitím algoritmu RSA, ktoré sú uchované v programe a je ich možné exportovať do súboru.

Export public key - zapíše do súboru rsaPublicKey.txt verejný kľúč

Export public + private key - zapíše do súboru rsaPrivateKey.txt privátny kľúč + verejný kľúč (súbor rsaPrivateKey.txt obsahuje všetky informácie pre dešifrovanie aj pre šifrovanie)

Import public key - naimportuje do programu verejný kľúč zo súboru rsaPublicKey.txt z priečinka Encrypt, následne je možné program použiť na šifrovanie súborov.

Import public + private key - naimportuje do programu privátny kľúč + verejný kľúč zo súboru rsaPrivateKey.txt z priečinka Decrypt, následne je možne program TurboPowerCrypt použiť na šifrovanie súborov, aj na dešifrovanie súborov, ktoré boli zašifrované s použitím obsiahnutého verejného kľúča.

6.2 Kontrola integrity

Na výpočet autentifikačného kódu sme použili objekt .NET frameworku HMACSHA256. Inicializujeme ho poľom 36 Bytov, ktorého prvých 32 Bytov tvorí inicializačný vektor použitý pri šifrovaní dát a posledné 4 Byty sú binárne vyjadrené kladné celé číslo 9.

Po šifrovaní sa z vytvorených dát vytvorí odtlačok pomocou inicializovaného HMACSHA256 objektu. Odtlačok sa zapíše na definované miesto v súbore s šifrovanými dátami. Pred pokusom o dešifrovanie sa najprv inicializuje HMACSHA256 objekt rovnakým parametrom ako pri šifrovaní a vytvára sa odtlačok zo zašifrovaných dát. Iba ak sa zhoduje vypočítaný HMAC s HMAC deklarovaným v súbore, proces pokračuje dešifrovaním dát.

Bitovú identickosť pôvodného a dešifrovaného súboru sme overili pomocou nástroja fc vo windowsovom príkazovom riadku:

```
C:\Users\JD>fc /b "C:\Users\JD\Documents\Visual Studio
2015\Projects\upbz2\upbz2RSA\upbz2RSA\bin\Debug\Decrypt\jednoGigo.dat"
"C:\Users\JD\Documents\Visual Studio 2015\jednoGigo.dat"

Comparing files C:\USERS\JD\DOCUMENTS\VISUAL STUDIO
2015\PROJECTS\UPBZ2\UPBZ2RSA\UPBZ2RSA\BIN\DEBUG\DECRYPT\jednoGigo.dat and
C:\USERS\JD\DOCUMENTS\VISUAL STUDIO 2015\JEDNOGIGO.DAT

FC: no differences encountered

C:\Users\JD>
```

7 Security comments

Po vytvorení asymetrického kľúča je potrebné privátny kľúč bezpečne uchovať (exportovať) a chrániť pred vyzradením, čím by sa znehodnotil kľúč a vyzradili ním šifrované súbory. Verejný kľúč je možné odoslať druhému používateľovi.

Každý používateľ by si mal vygenerovať a používať svoj vlastný pár verejného a privátneho kľúča.

Pre výmenu šifrovaných súborov medzi dvoma používateľmi sa odporúča, aby si každý vygeneroval jeden vlastný pár kľúčov. Verejné

kľúče si vzájomne zverejnia. Následne pre ďalšiu komunikáciu medzi nimi už nie je potrebné vytvárať nový pár kľúčov.

Program TurboPowerCrypt uchováva v zašifrovaných súboroch metadáta potrebné pre dešifrovanie súboru a kontrolu integrity. Tieto metadáta sú uložené v prvých 200 Bytoch súboru. Modifikácia zašifrovaného súboru treťou stranou v tejto oblasti môže viesť k stavu, že súbor bude nezvratne poškodený. Nebude ho možné dešifrovať ani pri použití správneho privátneho kľúča.