ITS3ILV Übung

C# Implementierung einer IT-Security Applikation zur Verschlüsselung bzw. Entschlüsselung von Dateien mit Windows-Forms

Betreuung der Übungen durch

Dr. Norbert Modl

INHALTSVERZEICHNIS

Inhaltsverzeichnis II

1 Einleitung 1

1.1 Ziel 1

1.2 Organisation 1

2 Technische Spezifikation 2

2.1 Einleitung 2

2.2 Datei- und Ordner-Struktur 2

2.3 Funktionalitäten 3

2.3.1 Windows Forms Design 3

2.3.2 Generierung der RSA-Schlüssel 3

2.3.3 Export des Öffentlichen RSA-Schlüssels 3

2.3.4 Datei verschlüsseln 4

2.3.5 Datei entschlüsseln 4

2.3.6 Aufräumen von Zwischen-Ergebnissen 5

2.3.7 Beenden der Applikation 5

3 Qualitative Anforderungen 6

3.1 Einleitung 6

3.2 Vorbereitung durch die Studenten 6

1 EINLEITUNG

1.1 ZIEL

Ziel der Übungen ist es den theoretischen Stoff aus der Vorlesung an einem praktischen Beispiel in die Tat um zu setzen.

Die im Kapitel 2 beschriebene technische Spezifikation soll als grundsätzliche Orientierung dienen. Geeignete Abweichungen sind nach Absprache mit dem betreuenden Dozenten erlaubt.

1.2 ORGANISATION

Je zwei Personen (max. 3) sollten diese Aufgabe gemeinsam bearbeiten. Innerhalb eines solchen Teams sollte darauf geachtet werden die Lasten möglichst geleichmäßig zu verteilen.

Die Ergebnisse sind in Form einer C#-Solution im Moodle in gezippter Form hochzuladen. Der Name der zip-Datei sollte die Autoren widerspiegeln.

2 TECHNISCHE SPEZIFIKATION

2.1 EINLEITUNG

Es ist eine Windows-Forms Applikation zu entwickeln in der mit Hilfe des ***RSA***-Verfahrens bzw. des ***AES256-Verfahrens*** eine Datei verschlüsselt und entschlüsselt werden sollte.

2.2 DATEI- UND ORDNER-STRUKTUR

Die Original-Dateien in unterschiedlichen Formaten, die verschlüsselte Datei bzw. die entschlüsselte Datei werden in dedizierten Ordnern abgelegt. Die geforderte Verzeichnis-Struktur lautet wie folgt:

```
$PATH/    00_Documents
          01_Encrypt
          02_Decrypt
```

Der Wert der Variablen `PATH` kann von Ihnen gewählt werden.

Die Speicherung des öffentlichen RSA-Schlüssels erfolgt in eine eigene Datei. Der Name dieser Datei kann beliebig gewählt werden. Der Pfad hierfür lautet:

```
$PATH/    03_Keys
```

Zur Anwendung des AES-Verfahrens sind ein symmetrischer Schlüssel und ein sogenannter Initialisierungsvektor (IV) erforderlich. Der symmetrische Schlüssel muss mit Hilfe des RSA-Verfahrens ver- bzw. entschlüsselt werden (hybrides Verfahren).

Nach der Verschlüsselung einer Klar-Text Datei müssen der symmetrische Schlüssel und der IV in einer eigenen Datei abgelegt werden. Vor den Entschlüsselung müssen diese beiden Informationen aus der Datei wieder gelesen werden. Zur Speicherung wählen Sie das Format `Base64`. Der Name dieser Datei kann beliebig gewählt werden. Der Pfad hierfür lautet:

```
$PATH/    03_Keys
```

2.3 FUNKTIONALITÄTEN

2.3.1 WINDOWS FORMS DESIGN

Die nachfolgende Abbildung zeigt eine beispielhafte Implementierung. Hinweis: Ihr Design kann hiervon abweichen.

2.3.2 GENERIERUNG DER RSA-SCHLÜSSEL

Mit Hilfe eines eigenen Controls (Button) sollte die Generierung der RSA-Schlüssel angestoßen werden. Verwenden Sie hierzu die Klasse: ***RSACryptoServiceProvider***

2.3.3 EXPORT DES ÖFFENTLICHEN RSA-SCHLÜSSELS

Mit Hilfe eines eigenen Controls (Button) sollte der Export des öffentliche RSA-Schlüssels angestoßen werden. Lesen Sie den RSA-Schlüssel aus dem ***RSACryptoServiceProvider*** Objekt aus und schreiben Sie die Werte in lesbarer Form in die Datei (XML-Format). Der Dateiname hierfür lautet z.B. `rsaPublicKey.txt`. Der Ziel-Ordner lautet: `$PATH/03_ Keys`

2.3.4 DATEI VERSCHLÜSSELN

Mit Hilfe eines eigenen Controls (Button) sollte diese Aktion angestoßen werden. Die Auswahl der Original-Datei sollte mit Hilfe des Controls ***OpenFileDialog*** implementieret werden. Der Quell-Pfad lautet: `$PATH/00_Documents`

Die Verschlüsselung sollte mit Hilfe des AES256-Verfahrens durchgeführt werden. Verwenden Sie hierfür die Klasse: ***RijndaelManaged***. Studieren Sie diese sorgfältig. Einige Eigenschaften bzw. Methoden dieser Klasse müssen recherchiert und geeignet angewendet werden.

Die verschlüsselte Datei sollte im Ordner: `$PATH/01_Encrypt` abgelegt werden. Bei gleichem Datei-Namen lautet die Datei-Endung `.enc`.

IV und symmetrischer Schlüssel für das AES-Verfahren müssen Sie ebenfalls in einer eigenen Datei geeignet ablegen.

2.3.5 DATEI ENTSCHLÜSSELN

Mit Hilfe eines eigenen Controls (Button) sollte diese Aktion angestoßen werden. Die Auswahl der verschlüsselten Datei sollte mit Hilfe des Controls ***OpenFileDialog*** implementieret werden. Der Quell-Pfad lautet: `$PATH/01_Encrypt`

In einem ersten Schritte müssen der IV und der symmetrische Schlüssel aus der hierfür vorher erzeugten Datei ausgelesen werden. Danach erfolgt die Entschlüsselung der `.enc`-Datei.

Diese Ergebnis wird in den Ordner `$PATH/02_Decrypt` geschrieben.

Nach dem Entschlüsselungsvorgang sollte der Windows-Explorer automatisch gestartet werden und den Inhalt des Ordners: `$PATH/02_Decrypt` zeigen.

Hinweis: Überprüfen Sie das finale Ergebnis aller Aktivitäten indem Sie die entschlüsselte Datei öffnen und den Inhalt „optisch“ überprüfen.

2.3.6 AUFRÄUMEN VON ZWISCHEN-ERGEBNISSEN

Mit Hilfe eines eigenen Controls (Button) sollte die Zwischen-Ergebnisse aus den Ordnern:

```
$PATH/    01_Encrypt
          02_Decrypt
          03_Keys
```

gelöscht werden.

2.3.7 BEENDEN DER APPLIKATION

Mit Hilfe eines eigenen Controls (Button) sollte die Applikation kontrolliert beendet werden.

3 QUALITATIVE ANFORDERUNGEN

3.1 EINLEITUNG

Es sollten alle Anforderungen aus Kapitel 2 umgesetzt werden. Ein grundsätzliches Exception-Handling sollte implementiert sein.

Zur Vermeidung unverhältnismäßig großer Komplexität des C#-Codes muss nicht die Behandlung jeder potentieller Exception implementiert werden.

Design und Detail-Ausgestaltung der Lösung bleibt dem Team überlassen.

Eine Demo-Implementierung wird durch den Dozenten vorab gezeigt.

3.2 VORBEREITUNG DURCH DIE STUDENTEN

Die Übung ist nicht trivial, aber bei entsprechender Vorbereitung auf jeden Fall in der zur Verfügung stehenden Zeit (2 Blöcke zu je 5 EH) machbar.

- Ein Studium der ***C#-Folien*** von Frau Prof. Hallewell ist empfehlenswert.
- Ebenso sollte der ***Umgang*** mit dem ***Visual Studio*** klar sein.
- Machen Sie sich vertraut mit ***MSDN Klassen-Recherchen*** über das Internet.
- Die Wirkungsweise ***Symmetrischer*** und ***Asymmetrischer Verschlüsselungsverfahren*** sollt bekannt sein. Gleiches gilt für die ***Verwendung*** von ***öffentlichen Schlüssel***. (siehe DM05-Folien von Dr. Modl).