Übungsaufgabe #2: FILECRYPT

2.12.2009

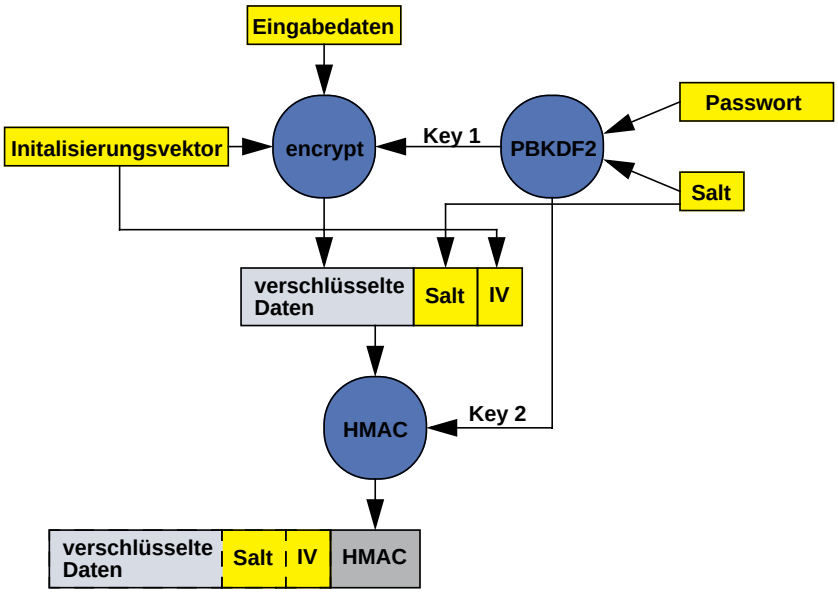
In dieser Aufgabe soll eine passwortbasierte Dateiverschüsselung implementiert werden.

Hierbei soll aus einem Passwort und einem zufälligen 160 Bit Salt durch Anwenden des in OpenSSL implementierten PBKDF2-Algorithmus mit 2000 Iterationen ein 512 Bit langer Schlüssel erzeugt werden, welcher zur weiteren Verwendung in zwei 256 Bit Schlüssel (`Key1` und `Key2`) aufgeteilt wird. `Key1` und eine zufälliger 128 Bit langer Initialisierungsvektor (`IV`) dienen als Eingabeparameter des `aes-256-ofb` Algorithmus, mit welchem der Dateiinhalt verschlüsselt wird. Das Tripel {verschlüsselte Daten, `Salt`, `IV`} soll durch einen 160 Bit langen `SHA-1` HMAC geschützt werden, wobei der durch PBKDF2 generierte `Key2` als Schlüssel dienen soll.

Als Ausgabe soll das Programm das Quad {verschlüsselte Daten, `Salt`, `IV`, `HMAC`} in der Zieldatei ablegen.

Bei der Entschlüsselung soll das Passwort überprüft werden, bevor der Entschlüsselungsalgorithmus initiiert wird (es sollen also nur PBKDF2 und HMAC berechnet werden). Sollte hierbei der selbe HMAC berechnet werden, ist die Datei durch Anwenden von `aes-256-ofb` zu entschlüsseln.

```
$./filecrypt enc test.txt test.enc.txt
Enter Password: geheim
$./filecrypt dec test.enc.txt text.txt
Enter Password: geheim
```



Parameter im Überblick (auch zu finden in `/proj/i4syssec/aufgabe2/parameter.h`):

```
#define AESKEYLEN 32     /* Key length of aes-256-ofb */
#define HMACKEYLEN 32    /* Key length for hmac */
#define SALTLEN 20       /* Salt for PBKDF2 to derive key from pw */
#define IVLEN 16         /* IV for aes-256-ofb, 128 bit */
#define HMACLEN 20       /* HMACLEN is 160 bit, as it shall base on SHA1 */
#define PBKDF2ITERATIONS 2000 /* Number of iterations */
```

Die Abgabe der bearbeiteten Aufgabe erfolgt in den Übungen am 9. oder 11.12. Die Bearbeitung kann in Gruppen zu zwei oder drei Personen erfolgen.