Übungen zu Systemsicherheit

Jürgen Kleinöder,
Michael Gernoth, Reinhard Tartler
Universität Erlangen-Nürnberg, Informatik 4

SYSSEC

D.1 Besprechung der Aufgabe ’findpw’

- Wieviele Passwörter pro Sekunde?
- Reihenfolge des Suchraums?
- Passwort gefunden?

D.2 Password-Based Key Derivation

- "Langsame" Funktion zur Ableitung von Schlüsseln anhand von Passwörtern
- PBKDF2
  - Standardisiert in RFC2898
  - Ersetzt PBKDF1
  - Kann beliebige Schlüssellängen erzeugen
  - Salt
  - Iterationszahl einstellbar
- Einsatzorte
  - WPA/WPA2
  - OpenOffice.org
  - LUKS (cryptsetup/dm-crypt unter Linux)
  - ....

D.3 Implementierung in OpenSSL

- OpenSSL bringt eine Implementierung von PBKDF2 mit
  - nur SHA1
  - undokumentiert
- Quellen von Openssl in `/proj/i4syssec/openssl/openssl-0.9.8g/`
- Definition in `/usr/include/openssl/evp.h`

```
int PKCS5_PBKDF2_HMAC_SHA1(const char *pass, int passlen,
        const unsigned char *salt, int saltlen, int iter,
        int keylen, unsigned char *out);
```

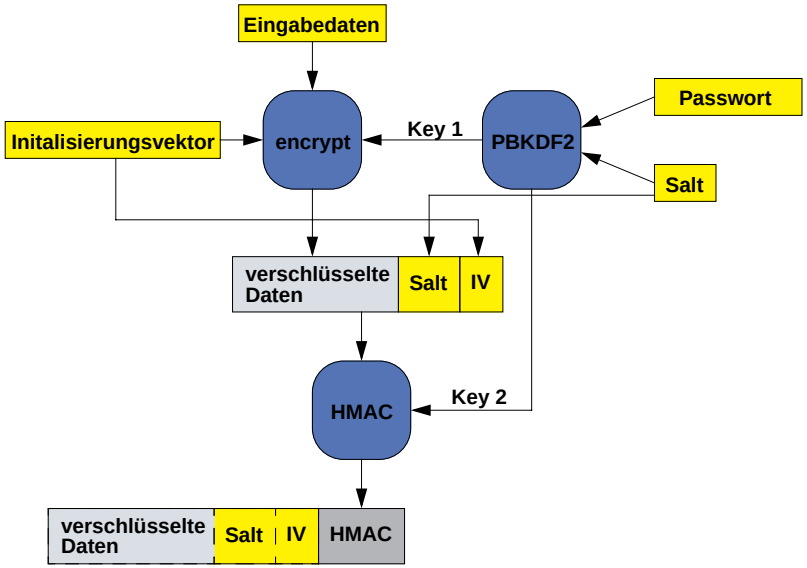
- Aus HMAC(3ssl)

```
unsigned char *HMAC(const EVP_MD *evp_md, const void *key,
       int key_len, const unsigned char *d, int n,
       unsigned char *md, unsigned int *md_len);
```

D.4 Aufgabe: filecrypt

- Dateiverschlüsselung mit Passwort
- Überprüfung des Passwort bevor die Datei entschlüsselt wird
- Parameter:
  - Verschlüsselung mit AES, 256 bit, Output-Feedback Mode (NICHT cbc!)
  - zufälliger Initialisierungsvektor (128 bit)
  - HMAC mit sha1
  - 160 bit Salt
  - PBKDF2
    - SHA1
    - 2000 Iterationen
    - 2 x 256 bit Schlüssellänge

D.5 Überblick



D.6 Hinweise

- Einlesen des Passworts mit OpenSSL-Funktionen:

```
/* read in password from stdin */
EVP_read_pw_string(pw, BUFLEN, "Enter Password: ", 0);
```

- Registrieren der vorhandenen Cipher und Digests:

```
OpenSSL_add_ssl_algorithms()
```

- Verwendung der **EVP_CIPHER**-Schnittstelle:

```
EVP_CIPHER_CTX cctx;
EVP_CIPHER_CTX_init(&cctx);
EVP_CipherInit_ex(&cctx, EVP_aes_256_ofb(), NULL, key, iv,
       enc_dec);
EVP_CipherUpdate(...);
EVP_CIPHER_CTX_cleanup(&cctx);
```