

Security Engineering

1. Übung

Vorbemerkung:

- Empfohlen ist, in den Praktikumsraum 6303 (ISL) ein eigenes Gerät (Notebook/Laptop) mit installiertem Linux oder FreeBSD (nativ oder VM) mitzubringen.
- Falls Sie kein eigenes Gerät mitbringen, können Sie bearbeitete Übungen auf einem USB-Stick mitbringen und diese mit Kommandos `usbmount` und `cp` in Ihr Homeverzeichnis kopieren. Hierfür sind im 6303 die FreeBSD-Maschinen `isl-1-01`, ..., `isl-1-08` installiert.

Aufgabe 1 (C Programm)

Diese Aufgabe dient dazu, dass Sie auf Ihrem Arbeitsgerät über eine funktionsfähige Umgebung mit der Programmiersprache C verfügen. Falls dies die ISL-Rechner sind, sind dort Editoren (`ee`, `vim`) und der Compiler `clang` vorhanden.

Das Shell-Kommando `date` gibt das aktuelle Datum mit Uhrzeit aus.

Schreiben Sie ein C-Programm, das den Zeitpunkt ebenfalls in ähnlicher Weise ausgibt:

```
Mon Apr  7 11:56:23 CEST 2025
```

- a) mit Hilfe von `time()`, gefolgt von `ctime()`
- b) mit Hilfe von `time()`, gefolgt von `localtime()` und `strftime()`

Falls Sie Informationen zu den C-Funktionen benötigen, hilft Ihnen das `man`-Kommando:

```
man 3 time
```

```
man 3 strftime
```

Aufgabe 2 (Hashfunktionen zur Prüfen der Integrität von Dateien)

Es gibt innerhalb der Systemkommandos auf Linux und FreeBSD-Systemen Hashfunktionen: in der Reihenfolge ihrer Wichtigkeit

sha256, sha1, sha384, md5 (FreeBSD) und sha256sum etc (Linux)

Wenden Sie diese Hashfunktionen an, um Hashwerte von Dateien zu bestimmen.

```
SHA256 (/etc/services) = ccda4683295b09834e17b1cce0c3c1945ec197...
SHA1 (/etc/services) = c42cb3105eac07d79fecb69976c7204818ee5415
SHA384 (/etc/services) = ab9487cfced4a262384de746430fdbfc0f8c97...
MD5 (/etc/services) = 89ad32116c62bee2a1eb3798d2583c96
```

Kopieren Sie die Datei `/etc/services` in Ihr Homeverzeichnis und verändern Sie einen Eintrag. Stellen Sie fest, dass dadurch der Hashwert verändert wird.

Geben Sie den Hashwert auch mittels `openssl` aus.

```
openssl dgst -sha256 ...
```

OpenSSL stellt noch weitere sichere Hashfunktionen zur Verfügung, probieren Sie insbesondere den neuen Hashstandard SHA-3 aus und auch SHA512, whirlpool, RIPEMD160.

Aufgabe 3 (Zufallsexperimente Pre-Image, Collision)

[Bezug: Folie 31 aus der Vorlesung]

Generieren Sie für eine eingegebene Obergrenze n in einer beliebigen Programmiersprache zufällige Zahlen zwischen 0 und $n - 1$, bis ...

- a) ein vorgegebener Wert y ausgegeben wird (simuliert Pre-Image)
- b) ein Wert y ausgegeben wird, der mit einem der vorherigen Werte übereinstimmt (simuliert Collision)

Zählen Sie die Anzahl der generierten Zahlen, bis das Ereignis eintritt.

Verifizieren Sie durch Mittelung der Experimente, dass für $n = 10000$ im Falle a) im Mittel 10000 und im Falle b) im Mittel ca. 125 Zahlen generiert werden müssen.

Aufgabe 4 (Wortlisten, Dictionaries)

Verifizieren Sie die Entropieangabe des XKCD-Comics aus den Vorlesungsfolien.

Dazu laden Sie sich Wortlisten in englischer und deutscher Sprache herunter und verifizieren Sie, dass der Sprachumfang näherungsweise zwischen 11 und 17 Bits Entropie besitzen kann.