[∞]Übungsblatt 5

[∞] Aufgabe 2: HOTP (Praxisaufgabe)

HMAC-SHA1 ist in hmac.rs implementiert.

HOTP ist in hotp.rs implementiert.

main.rs implementiert einen "YubiKey-Simulator" zum generieren von HOTP-Werten und einen kommandozeilen-basierten HOTP Validator.

Eine kompilierte Linux x86-64 Binary kann von hier heruntergeladen werden.

Zum selbst kompilieren ist es ausreichend, cargo build --release auszuführen.

☼ Kommandozeilen-Parameter

Das Programm kann mit Kommandozeilen-Argumenten ausgeführt werden, um es zu konfigurieren. Valide Schreibweisen zum Konfigurieren der HOTP-Wert-Länge sind beispielweise --length=6 und -16.

Die folgenden Parameter werden akzeptiert. Unbekannte Parameter-Namen/Kurzformen werden ignoriert, ungültige Werte führen zur Ausgabe eines Fehlers und zum Ausführungsabbruch.

Name	Kurzform	Beschreibung	Default
mode	-m	Ausführungsmodus. Erlaubte Werte: "key", "validator"	"key"
counter	-C	Initialer Wert des HOTP Moving Factors. Integer >= 0	0
key	-k	HOTP Secret Key. Bytes als Hex-String	00000 (40 Zeichen)
length	-1	HOTP-Wert-Länge. Integer >= 0	6
window	-W	Fenster-Größe für den Validator	20

[™] Ausführungmodi

☼ Als YubiKey-Simulator (HOTP-Generator)

Im key -Modus gibt das Programm für jede Eingabe eines Zeilenumbruchs einen HOTP-Wert aus. Es beginnt mit dem konfigurierten Counter-Wert und erhöht den Wert dann stets um 1. Ein mit

```
ykpersonalize \
-1 -a1212121212121212121212121212121212 -ooath-hotp \
-ooath-imf=29344 -ofixed=ccccccc -oserial-api-visible -oappend-cr
```

konfigurierter YubiKey ist äquivalent zu dem mit

```
./hotp_linux_x86-64 \
--length=6 --mode=key --counter=29344 \
--key=12121212121212121212121212121212
```

ausgeführten Programm.

```
YubiKey Output
-ooath-imf=29344 -ofixed=ccccccc -oserial-api-visible -oappend-cr
Firmware version 3.4.9 Touch level 1029 Program sequence 8
                                                                                                                  G
Configuration data to be written to key configuration 1:
OATH id: 000000000000
Ш
00000000738204
ticket_flags: APPEND_CR|OATH_HOTP
config_flags:
extended_flags: SERIAL_API_VISIBLE
Commit? (y/n) [n]: y
  Executing as HOTP generator. Enter a newline to start generating values.
HOTP value "486073" at counter value 29344
HOTP value "528312" at counter value 29345
HOTP value "976603" at counter value 29346
HOTP value "278233" at counter value 29347
HOIP value "310690" at counter value 29348
HOTP value "310690" at counter value 29348
HOTP value "544536" at counter value 29349
HOTP value "3049456" at counter value 29350
HOTP value "738204" at counter value 29351
HOTP value "537214" at counter value 29352
HOTP value "962535" at counter value 29353
```

െ Als HOTP Validator

Im validator -Modus überprüft das Programm eingegebene HOTP-Werte auf Validität über die konfigurierte Fenstergröße. Alle Zeichen einer Eingabezeile bis auf die letzten Length Zeichen werden dabei ignoriert, sodass das Programm direkt gegen einen "echten" YubiKey getestet werden kann.

Ein mit

```
ykpersonalize \
-1 -a121212121212121212121212121212121212 -ooath-hotp \
-ooath-imf=29344 -ofixed=ccccccc -oserial-api-visible -oappend-cr
```

konfigurierter YubiKey erzeugt mit

```
./hotp_linux_x86-64 \
--length=6 --mode=validator --counter=29344 --window=20 \
--key=1212121212121212121212121212121212
```

validierbare HOTP-Werte.