

```
Aufgabe 4 -- zsh -- 128x30
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
KEY=$(od -t x4 /dev/urandom | head -1 | cut -c 17- | sed -e "s/ //g")
echo "Alice's key: $KEY"
Alice's key: 6c0c01c48587c673dd87b435803c987c
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % openssl dgst -sha256 -mac HMAC -macopt hexkey:$KEY /Users/xudongzhang/services_copy
HMAC-SHA256(/Users/xudongzhang/services_copy)= 4eddec88c45b618e2d025e435014898be7268727a41c240260f905fa67d82712
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % CALCULATED_HMAC=$(openssl dgst -sha256 -mac HMAC -macopt hexkey:$KEY /Users/xudongzhang/services_copy | cut -d ' ' -f 2)
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % echo "calculated key: $CALCULATED_HMAC"
calculated key: 4eddec88c45b618e2d025e435014898be7268727a41c240260f905fa67d82712
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % CALCULATED_HMAC=$(openssl dgst -sha256 -mac HMAC -macopt hexkey:$KEY /Users/xudongzhang/services_copy | cut -d ' ' -f 2)
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % echo "calculated key after change: $CALCULATED_HMAC"
calculated key after change: dda3e9bf73ef58cf2a47c8b9c6756e505d782c3ae9f1be804cb7fa7c328c4d13
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % CALCULATED_HMAC=$(openssl dgst -sha256 -mac HMAC -macopt hexkey:$KEY /Users/xudongzhang/services_copy | cut -d ' ' -f 2)
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 % echo "calculated key after change it back: $CALCULATED_HMAC"
calculated key after change it back: 4eddec88c45b618e2d025e435014898be7268727a41c240260f905fa67d82712
(base) xudongzhang@Xudongs-MBP-2 Aufgabe 4 %
```

Alice generiert 128-Bit-Schlüssel

Alice erzeugt HMAC mit OpenSSL für
"/Users/xudongzhang/services_copy"

Bob berechnet den HMAC der empfangenen
"/Users/xudongzhang/services_copy"

Nachdem „services_copy“ geändert wurde,
Bob berechnet wieder den HMAC, bekommt er ein
ander HMAC

Nachdem „services_copy“ auf das Original zurück
geändert wurde, erhielt er den Original HMAC

1. `od -t x4 /dev/urandom` :

- `od` : Steht für "octal dump". Es ist ein Programm, das die Daten von einer Datei oder von der Standardeingabe liest und sie in verschiedenen Formaten darstellt.
- `t x4` : Gibt das Format an, in dem die Daten angezeigt werden sollen. `x4` bedeutet, dass die Daten als 4-Byte-Hexadezimalwerte angezeigt werden.
- `/dev/urandom` : Ist eine spezielle Datei, die einen unendlichen Strom von zufälligen Bytes liefert. Hier wird sie als Eingabedatenquelle verwendet.

2. `head -1` :

- `head` : Ein Befehl, der die ersten Zeilen einer Datei oder der Ausgabe eines Befehls anzeigt.
- `1` : Zeigt nur die erste Zeile der Ausgabe von `od` an.

3. `cut -c 17-` :

- `cut` : Ein Befehl, der Teile von jeder Zeile der Eingabe entfernt und anzeigt.
- `c 17-` : Schneidet die Zeichen von Spalte 17 bis zum Ende jeder Zeile heraus. Dies wird verwendet, um nur den Teil der Ausgabe zu extrahieren, der die Zufallswerte enthält.

4. `sed -e "s/ //g"` :

- `sed` : Ein Stream-Editor, der Textbearbeitungen auf Zeilen ausführen kann.
- `e "s/ //g"` : Ein `sed` Ausdruck, der alle Leerzeichen (" ") durch nichts (//) ersetzt, also entfernt. Das `g` am Ende steht für "global", was bedeutet, dass alle Vorkommen von Leerzeichen in der Zeile entfernt werden.

Zusammengefasst generiert die Sequenz eine zufällige Zahl aus `/dev/urandom`, formatiert sie als 4-Byte-Hexadezimalwerte, wählt die relevante Zeile und Zeichen aus, und entfernt schließlich alle Leerzeichen, um einen fortlaufenden 128-Bit-Hexadezimalwert zu erhalten.