Grazer Linuxtage 2021

Bitcoin sicher aufbewahren

Marco Horn

Intro: Was soll dieser Vortrag?

Bitcoin und OpenSource

- Als Entwicklerin/Entwickler
  - Viel über Kryptographie lernen
  - Spenden via Bitcoin annehmen
- Als Unterstützerin/Unterstützer
  - Bitcoin spenden für coole Projekte
- Allgemein als Linuxerin/Linuxer
  - “Du hast doch Ahnung… Was muss ich beachten?”

Worum geht es hier NICHT?

- Werbung für Bitcoin – das können andere besser
- Kursversprechen und Anlagetips
- Die Schattenseiten von Bitcoin (z.B. Energieverbrauch)
- Gesellschaftliche Bedeutung von Crypto-Coins

Worum geht es dann?

- Direktes Ziel: Eigene Risiko-Abschätzung
  - Sich Bitcoin nicht stehlen lassen
  - Sich aber auch nicht selbst aussperren
  - Notfall-Vorsorge: wie kommen Freunde / Familie im Falle des Falles an den Schatz?
- Indirektes Ziel: Etwas lernen
  - Verstehen der Bitcoin-Blockchain
  - Ideen für eigene Experimente / Projekte

Teil 1: Wallet-Arten

Was sind Wallets? Was tun sie?

- Verwalten der Adressen und deren private/public Keys
- Analyse der Blockchain: Welche Amounts liegen aktuell auf welchen Adressen der Wallet?
- Interagieren mit dem Bitcoin-Network
  Ausgeben von Bitcoin: Erstellen und Signieren von Transaktionen

Custody-Wallets

- Vertrauenswürdiger Anbieter mit hübscher Web-UI
- **Pro**: Der Anbieter kümmert sich um alles
- **Contra**: (jede Menge gute Gründe)
- Merke: ***Not your keys – not your coins***

Smartphone-Wallets

- Wallet-App (echte Wallet mit Secrets in der App)
- **Pro**: Immer dabei, wenn man sie braucht
- **Contra**: Smartphone ist ein sehr unsicheres Gerät
- Denkanstoß: Wie oft gibt man **spontan** Bitcoin aus?

Wallet am (Linux)PC

- Auf Wunsch mit eigener Blockchain-Kopie (Platz ist da!)
- **Pro**: Lässt sich viel besser absichern (Docker, QubeOS)
- **Contra**: Noch immer oft online – und nicht immer dabei
- Merke: Die Wallet muss nicht 24/7 laufen

Hardware-Wallets

- Eigene Geräte im USB-Stick-Format
- **Pro**: (fast) nie online, je nach Lösung
- **Contra**: Umständlich zu bedienen
- Hinweis: Produkte für jedes Level an Paranoia

Paper-Wallets

- Total offline: Auf Papier die wichtigsten Daten
- **Pro**: Vollständige Trennung vom Internet
- **Contra**: Bitcoin ausgeben kostet etwas Zeit
- Anmerkung: Seit HD-Wallets praktisch ausgestorben

Teil 2: Was genau ist zu sichern?

Wie funktioniert Bitcoin?

- Blockchain => Blöcke => Transactions
- Transaction: 1-n Inputs, 1-n Outputs
- Input: Verweist auf einen vorherigen Output (UTXO)
- Output: BTC-Adresse, enthält sogenanntes Lock-Script

- Input = “Ausgeben von Bitcoin”, nur mit private Key
- Output = “Empfang von Bitcoin” - public Key reicht

Was muss man sich also merken?

- Private Key
  - Wird für das Ausgeben von Bitcoin benötigt
  - Public Key und damit Adresse lässt sich ableiten

- Wie sieht der private Key aus?
  - 256Bit = 32 Bytes
  - Als Hex-Darstellung String mit 64 Zeichen
  - Alternativen: QR-Code, WIF (Wallet Import Format)

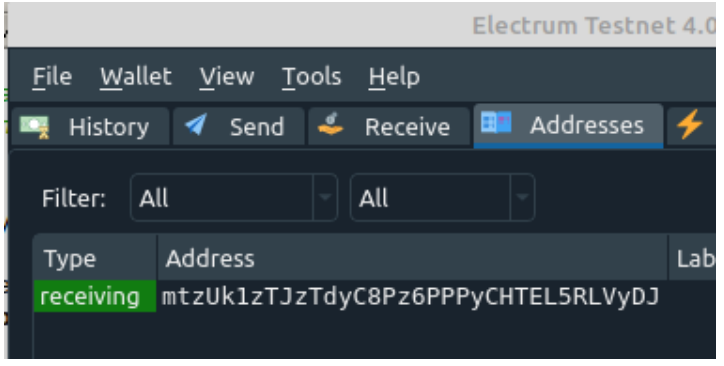
BTC-Address erstellen (python)

```python
# Convert a private key (256bit) into P2PKH BTC address

import pybtc
from pybtc.functions.hash import hash160, double_sha256
from pybtc.constants import *
from pybtc.functions.encode import encode_base58
from pybtc.functions.tools import bytes_from_hex

private_key = "ef235aacf90d9f4aadd8c92e4b2562e1d9eb97f0df9ba3b508258739cb013db2"
public_key = pybtc.PublicKey(private_key).key          # ECDSA curve secp256k1
public_key_hash = hash160(public_key)                  # RIPEMD160 + SHA256
prefix = TESTNET_ADDRESS_BYTE_PREFIX                   # 0x80: Main, 0xEF: Test
address_raw = b"%s%s" % (prefix, public_key_hash)      # Add prefix to hash
address_raw += double_sha256(address_raw)[:4]          # Add the first 4 byte of checksum
print(encode_base58(address_raw))                      # Encode bytes to BASE_58
```

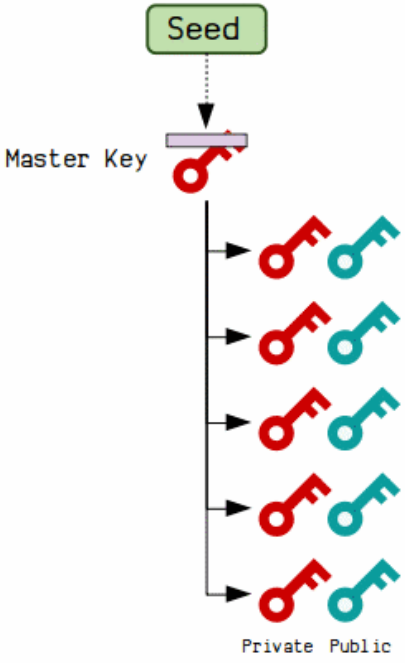
Wie prüfe ich das?

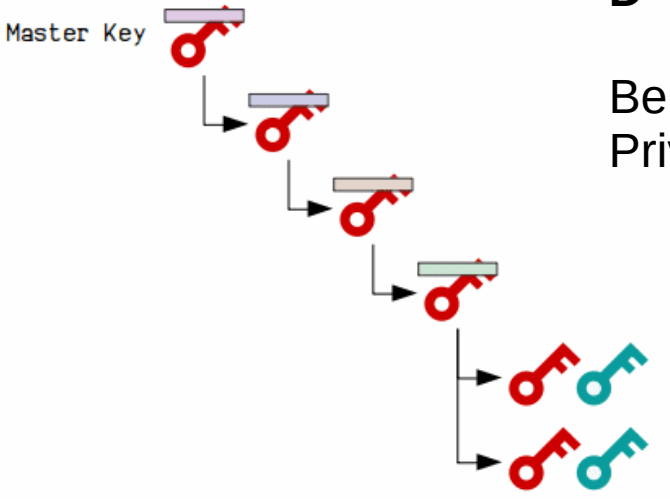
- Private Key => WIF
- Wallet: Import WIF

```python
# Convert private key into WIF
# see https://en.bitcoin.it/wiki/Wallet_import_format

from pybtc.functions.key import private_key_to_wif

private_key = "ef235aacf90d9f4aadd8c92e4b2562e1d9eb97f0df9ba3b508258739cb013db2"
wif = private_key_to_wif(private_key, True, True) # compressed and testnet

print(wif)
```



HD-Wallets (1)





H – Hierarchical
D – Deterministic

Beliebig tiefe Struktur
Private Keys wiederherstellbar

HD Wallets (2)

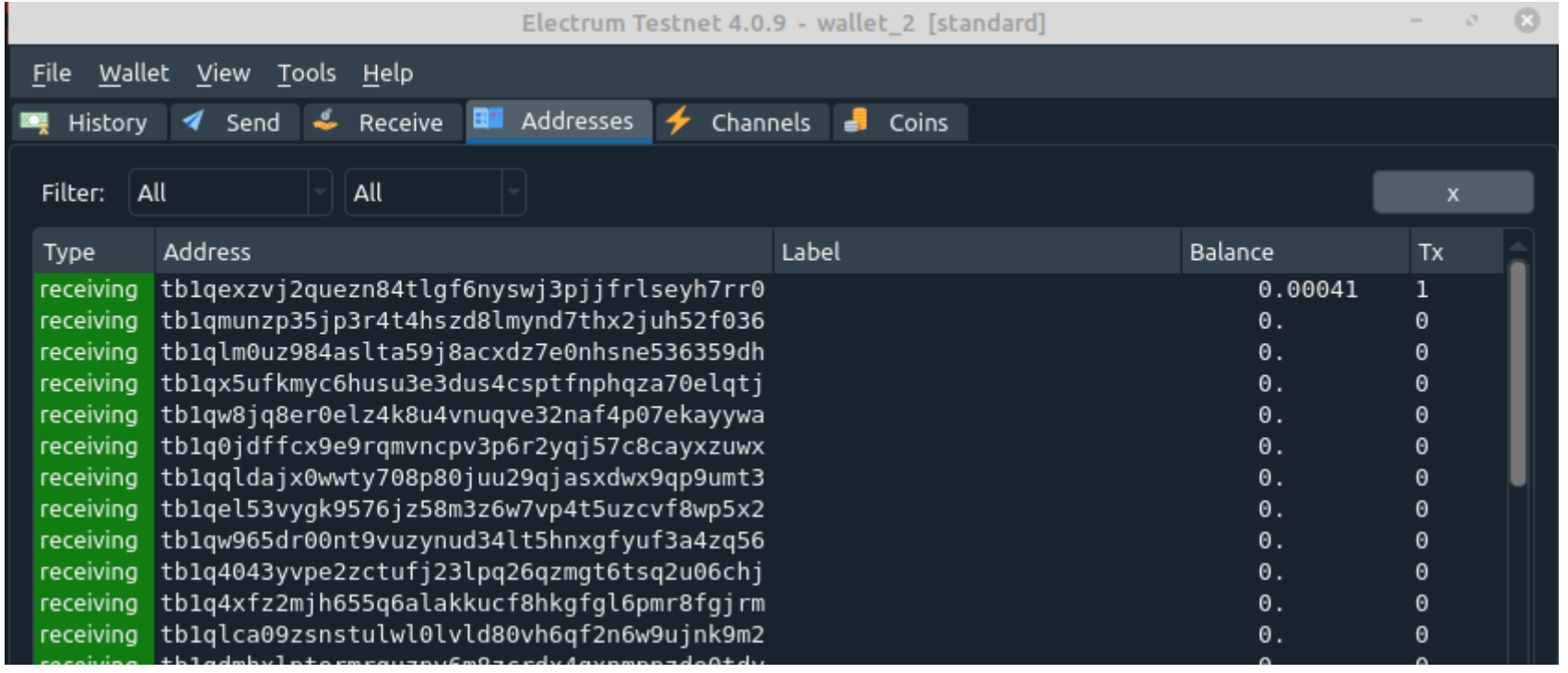
- Seed: 12 bis 24 Worte (2048 fixe Worte)
- Master-Key (512 Bits = 64 Bytes)
- Private-Keys ergeben sich aus
  - Master-Key (den hat man mit dem Seed)
  - Derivation-Path (hier gibt es Standards)
  - Nummer des Private-Keys: beginnend bei 0

HD-Wallet erstellen (python)

```python
import pybtc

# seed of 24 words
seed = ("loan melody service repeat engage spy today category "
        "sword twice youth jaguar purpose guitar now cream "
        "weird bid cushion correct negative enrich oblige similar")

# P2WPKH: Pay 2 Witness Public Key Hash (native SegWit)
w = pybtc.Wallet(init_vector=seed, path_type="BIP84", testnet=True)

# First addresses
print(w.get_address(0, chain="external")["address"])     # payment address
print(w.get_address(1)["address"])                       # "external" is default
print(w.get_address(0, chain="internal")["address"])     # internal: change address
```

HD-Wallet in Electrum importieren

- Address: tb1qexzvj2quezn84tlgf6nyswj3pjjfrlseyh7rr0



Zusammenfassung

- Private Key
  - Man kann Public Key + BTC-Adresse daraus generieren
  - Amount steht als UTXO in der Public Blockchain
  - Private Key wird benötigt, um die UTXO als Input zu nutzen
- Variante 1: Für jede Adresse eigenen Private Key merken
- Variante 2: Seed für Master-Key merken

Teil 3: Eigenes Security-Konzept

Gefahr: Hacking

- Custody-Anbieter sind begehrte Angriffs-Ziele
- Eigenes Smartphone sehr anfällig
- 24/7-PC mit laufender Wallet evtl mal “Beifang”
- Copy & Paste: Viele Apps lesen die Zwischenablage aus!

Gefahr: Diebstahl

- ALLE Kopien sichern
  - Eine Kopie in falschen Händen reicht
- Seed gut verstecken
  - Auch eine Fotografie des Seeds reicht
  - Bei Einbruch / Hausdurchsuchung evtl “kopiert”
- Smartphone + Laptop sind schnell geklaut

Gefahr: Verlust des Keys/Seeds

- USB-Stick kann unleserlich werden
- Papier kann feucht werden / verbrennen / aus Versehen entsorgt werden (beim Siedeln weggeschmissen etc)
- Plastik-Karten können bei Hitze zerstört werden

Gefahr: Sich selbst aussperren

- Seed merken (“Brain Wallet”) - keine gute Idee
- Kopie vergraben und den Ort vergessen
- Kopie bei Freunden hinterlegen – und die entsorgen die Kopie aus Unwissen
- Kopie verschlüsseln – und Passwort vergessen

Gefahr: Vendor Lock-In

- Externer Anbieter: man liefert sich quasi aus
- Software-Wallet: Geänderte Lizenz, Abandom-Ware
- Hardware-Wallet: Hersteller in Insolvenz
- Merke: Nutze standardisierte Formate und **Open Source**

Gefahr: Plötzlicher Tod

- Eine Kopie dort ablegen, wo sie im Fall des Falles gefunden wird
- Rechtzeitig das Wissen verbreiten: “Wenn mir mal was passiert, ruf Marco an!”
- Der Wert kann unerwartet anziehen
  - September 2020 ca 10.000€/BTC, heute: 5x soviel

Persönliche Empfehlung

- BIP39 Seed mit 24 Worten
- Seed aufschreiben zusammen mit klarer Info:
  in 5 Jahren kann man viel vergessen!
- Einmal jährlich die Funds prüfen (Weihnachten, Geburtstag, Staatsfeiertag): Seed importieren in neuer Wallet (aktuellste Version)
- Seed an zwei Locations aufbewahren – und NIE auf einem Rechner als Textfile

Weitere Ideen

- MultiSig-Adressen
  - z.B. zwei von drei Private Keys nötig zum “Einlösen”
  - Die drei Private Keys an verschiedenen Stellen aufbewahren
- Seed mit Zusatzwort
  - Seed allein reicht nicht – Zusatzwort allein auch nicht
- Plausible Deniability
  - “Köder-Wallet” und “hidden wallet” bei manchen HW-Wallets

Teil 4: Maker Ideen

Maker Ideen - Software

- Entropy-Shaker
  - Smartphone nutzen um einen Seed zu würfeln
  - Nutzt Bewegungs-/Magnetfeld-Sensoren für echten Zufall
- Address-Info
  - Smartphone importiert reine Adressen
  - Kann zu diesen Adressen QR-Codes generieren
  - Kann mit HD-Wallets umgehen (über Master-Public-Key)

Maker Ideen: Hardware

- Private-Key als “Serien-Nr” unverdächtig verstecken
  - Als reine Hex-Nr nicht sofort erkennbar (im Gegensatz zu den Worten aus der 2048-Wort-List)

- Bitcoin-Wall-Info: Als Wand-Installation mit Raspi
  - Arbeitet auf dem xpub der HD-Wallet
  - Summiert die BTC aller Adressen zusammen
  - Rechnet in aktuellen Tauschkurs um
  - Ausgabe auf e-Ink / mit Chart-Verlauf / mit Hintergrundfarbe

Happy Coding!

- Doku-Einstieg: https://en.bitcoin.it/
- BIP: https://github.com/bitcoin/bips
- Gute Online-Doku: https://learnmeabitcoin.com/
- pybtc: https://github.com/bitaps-com/pybtc
- 21Bitcoin Podcast: https://einundzwanzig.space/podcast/ (Folge #9: Nachlass-Verwaltung)