



OST

Eastern Switzerland
University of Applied Sciences

Blockchain (BlCh)

Wallets and Seed Phrases

Thomas Bocek

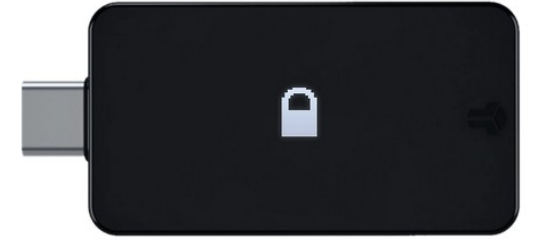
15.10.2026

Learning Goals

- Lecture 6
 - Understand wallet types and their security properties
 - Custodial vs. non-custodial
 - From private key to address
 - Explain HD wallets and BIP32/BIP39/BIP44 standards
 - Describe the path from mnemonic phrases to derived keys
 - Understand hardened vs. non-hardened key derivation
 - Know real attacks and what comes after the seed phrase

Introduction

- What are cryptocurrency wallets?
 - Store, send, and receive digital currencies
 - Stores keys, not coins (coins are on chain)
 - Analog to ~bank accounts
- Key functions
 - Storage of private and public keys
 - Sign transactions with private keys
 - Interaction with blockchain
 - Balance checking and transaction history
- Types of wallets
 - Hardware wallets, software wallets, paper wallets

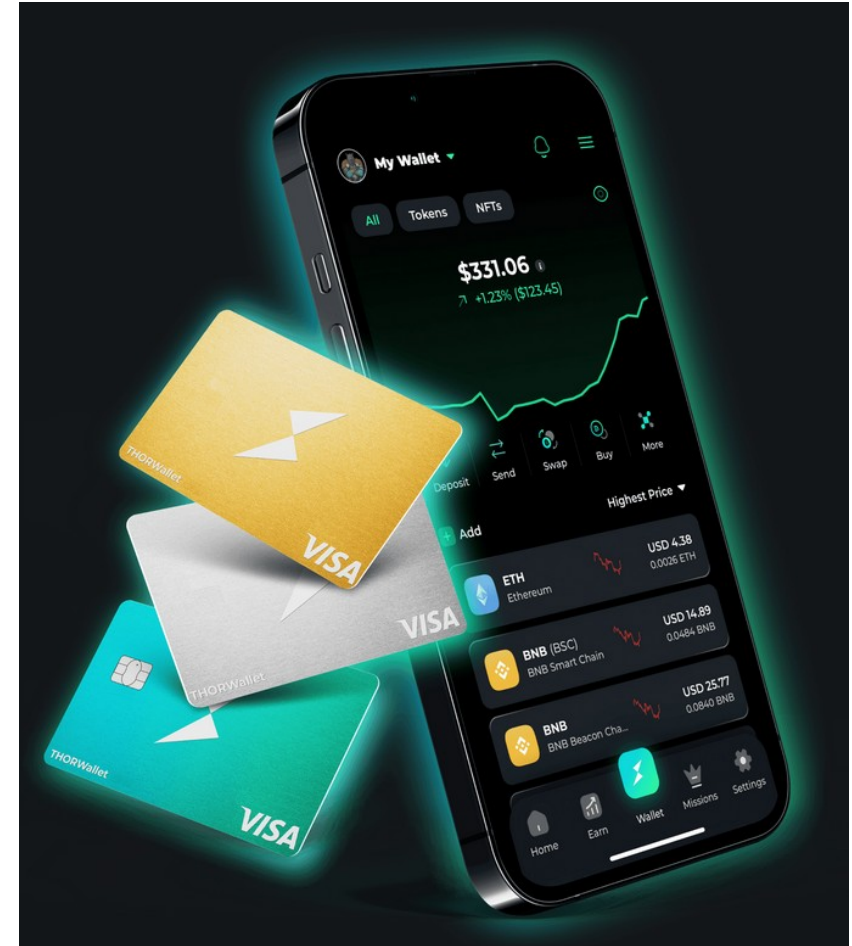


- Hardware wallets
 - [Trezor](#), [Ledger](#), [BitBox](#), ...
 - Specialized hardware (single purpose)
- Software wallets
 - [Metamask](#), [Trust Wallet](#)
- Paper wallet
 - Physical document with mnemonic words

	SW wallet	HW wallet	Paper wallet
Hot wallet	x		
Cold wallet		x	x

Introduction

- Importance of wallet security
 - Keeping assets safe from unauthorized access
 - Importance of backup and recovery methods
- Convenience and accessibility
 - Ease of use, mobile and desktop access
 - Importance for widespread adoption of cryptocurrencies
- Cryptocurrency wallets vs traditional banking
 - User-controlled security vs. bank-managed security
- Custodial vs. non-custodial
 - Exchange (Coinbase, Binance): login, no keys → promise to pay
 - FTX, Nov 2022: customer funds misused, bankrupt [\[link\]](#)
 - "Not your keys, not your coins"



From key to address

- Private key k
 - ~256-bit random number
- Public key $K = k \times G$
 - Curve secp256k1 (same as Bitcoin)
 - "x" is scalar multiplication on the elliptic curve
 - $k \rightarrow K$ fast, $K \rightarrow k$ practically impossible (discrete log)
- Ethereum address
 - keccak256(K) $\rightarrow$ last 20 bytes
 - Mixed case = checksum (EIP-55) [[link](#)]
 - Bitcoin: same key, other address format
- One private key, many chains
 - $k = 0x0000...0001$
 - Bitcoin (old):
1BgGZ9tcN4rm9KBzDn7KprQz87SZ26SAMH
 - Bitcoin (segwit):
bc1qw508d6qejxtdg4y5r3zarvary0c5xw7kv8f3t4
 - Ethereum:
0x7E5F4552091A69125d5DfCb7b8C2659029395Bdf
 - Tron:
TMVQGm1qAQYVdetCeGRRkTWYYrLXuHK2HC
- Same key $\rightarrow$ same public key $\rightarrow$ different hash + encoding

HD wallets

- Hierarchical Deterministic (HD) wallets
 - Many cryptocurrency wallet are HD wallets
 - Based on the [BIP32/BIP44](#) protocol (not restricted to Bitcoin)
 - Creation of derived keys from single master seed
- Key features
 - Generation of multiple addresses from a single seed
 - Simplifies management and backup
 - Each transaction can use a unique address
 - Enhanced privacy
- Understanding BIP32/BIP44
 - BIP32 (Bitcoin Improvement Proposal 32) introduces the concept of hierarchical deterministic wallets
 - BIP44 builds on BIP32, adding a structure for multiple coin types and accounts
- Mechanism of HD Wallets
 - Based on [BIP39](#) mnemonic phrase
 - Seed leads to the generation of private keys

HD wallets

- Benefits
 - Efficient backup: single seed backup is sufficient for multiple addresses and keys
 - Easy organization: Easy management of funds across various addresses/accounts
 - One seed, many accounts, BTC, ETH, ...
- Disadvantages
 - User Experience → most wallets ask you to write down the seed phrase
 - Unexperienced user: what is this? Is this important?
- BIP39 mnemonic phrase
 - Seed phrase: series of words from a defined [list](#)
 - Essential for wallet backup and restoration
 - If lost, your cryptos are lost
- Seed Phrase Composition
 - Typically a sequence of 12 or 24 words
 - Encoding of 128bit or 256bit
- Let's see how it works:

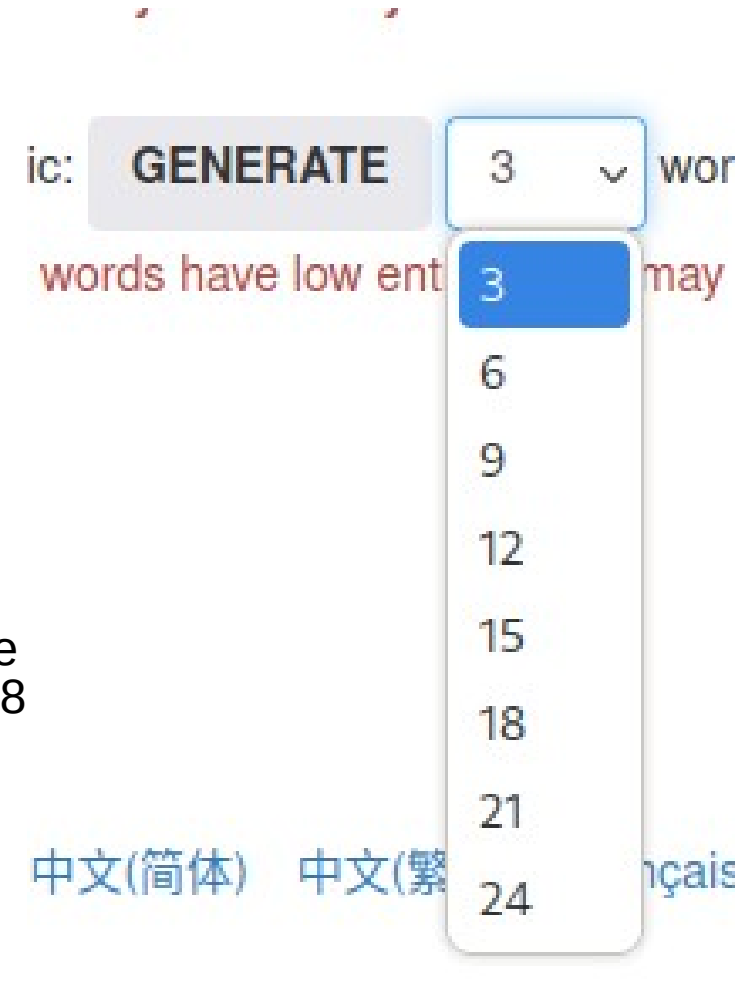
BIP39

- Generate a random number 128 bit or 256
 - Lets use 128bit for simplicity
 - Create random hex number (128bit)
 - `hex=$(hexdump -vn16 -e'16/1 "%02X"' /dev/urandom)`
 - Convert to binary
 - `hex_bin=$(echo "obase=2; ibase=16; ${hex}" | BC_LINE_LENGTH=0 bc)`
 - Word list has 2048 entries = 11bit
 - 12 words x 11 bit = 132bit, 4 bit wasted?
 - 4bit used as checksum – append first 4 bit of sha256(rand number), 24 words x 11 bit = 264bit, 8 bit checksum
 - `hash_hex=$(printf "%s" "$hex" | xxd -r -p | sha256sum | cut -d' ' -f1 | tr '[:lower:]' '[:upper:]')`
 - `checksum=$(echo "obase=2; ibase=16; $hash_hex" | BC_LINE_LENGTH=0 bc)`
 - `echo $(printf "%0128s" "$hex_bin" | tr ' ' 0)$(printf "%0256s" "$checksum" | tr ' ' 0 | cut -c1-4)`

- | | | |
|-------------|-------------|-------------|
| 11111100010 | 11100101100 | 11010001111 |
| 11111111000 | 00001010000 | 10000101011 |
| 10100111000 | 00110100110 | 01110000001 |
| 11111111111 | 01001000110 | 00111111110 |
- Take first 11 bit, lookup **word**
 - 11111100010 → 2018 → **wisdom**
- Take second 11 bit, lookup word
 - 11100101100 → 1836 → **tortoise**
- ...
- Take the last 11 bit, lookup word
 - 00111111110 → 510 → **divert**
- Wrong words = checksum won't match

BIP39

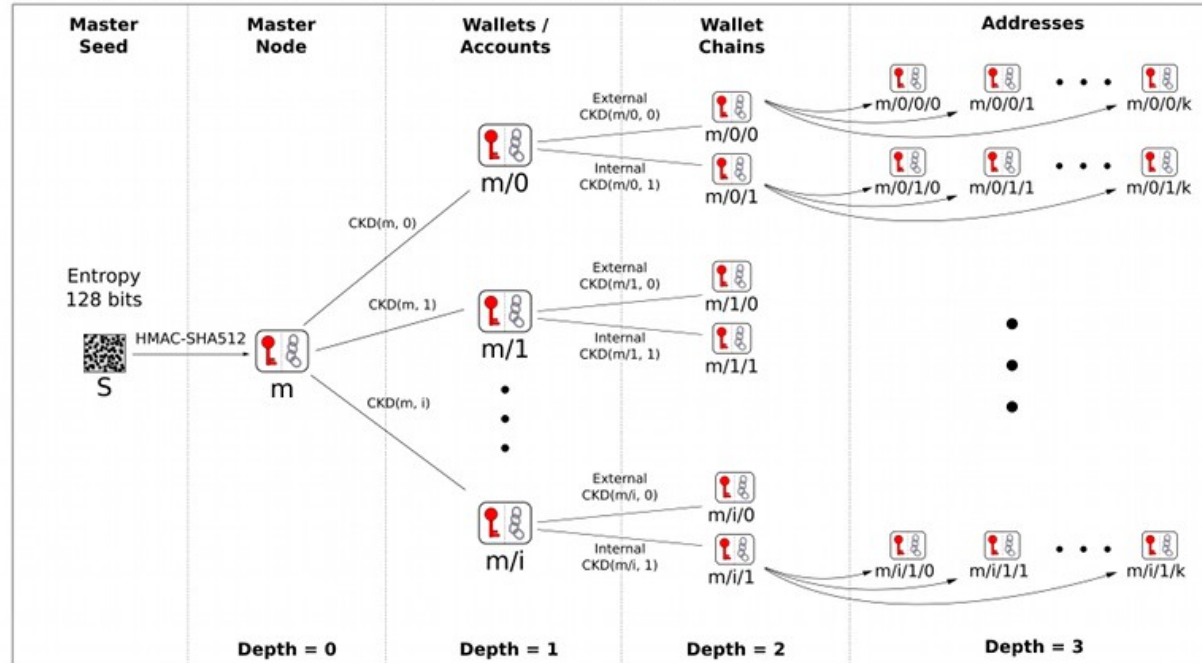
- 256 bit, same, but 8bit checksum
 - Mnemonic Code Converter [\[link\]](#)
 - Never enter a real seed online
- Seed extension
 - 13th/25th word, any text/passphrase
- From mnemonic to seed
 - [PBKDF2-HMAC-SHA512](#) function with mnemonic sentence as password, string "mnemonic" + passphrase as salt, 2048 rounds, 512-bit seed
 - Seed = PBKDF2("wisdom tortoise spin...", "mnemonicyourpassphrase", 2048, ...)
- Seed can be used for BIP-32



BIP32/BIP44

- BIP 32

BIP 32 - Hierarchical Deterministic Wallets



Child Key Derivation Function $\sim$ $CKD(x,n) = \text{HMAC-SHA512}(x_{\text{Chain}}, x_{\text{PubKey}} || n)$

- BIP 44

- m / purpose' / coin_type' / account' / change / address_index
 - Purpose $\rightarrow$ 44
 - Coin type
 - Bitcoin: m/44'/0'/2'/0/1
 - Ethereum: m/44'/60'/2'/0/1
 - Account $\rightarrow$ Account 2
 - Change (Bitcoin specific – resp. UTXO)
 - Address_index $\rightarrow$ Index 1
- Hardened vs. non-hardened
 - Hardened: $\text{hash}(\text{parent private key} + \text{index})$
 - Non: $\text{hash}(\text{parent public key} + \text{index})$
 - Risk: xpub + one non-hardened child private key $\rightarrow$ parent private key
 - $k_{\text{parent}} = k_{\text{child}} - a \rightarrow a$ is computable from xpub
 - $\rightarrow$ all keys exposed, hardening stops this

ECC

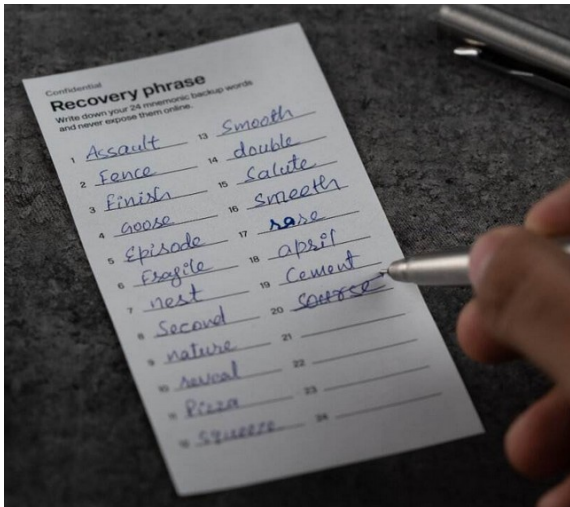
- $K = k \times G$
 - Private Key (k)
 - Base Point (G)
 - Parent Public Key (K) $\rightarrow K = k \times G$
 - "x" is scalar multiplication on the elliptic curve
- Key derivation
 - $k_{\text{child}} = k + a$
 - $K_{\text{child}} = K + a \times G$
 - Works because $(k + a) \times G = K + a \times G$
- a based on
 - $\text{hash}(\text{parent private key} + \text{index}) \rightarrow \text{hardened}$
 - $\text{hash}(\text{parent public key} + \text{index}) \rightarrow \text{non-hardened}$
- Be aware:
 - Single point of failure
 - User responsibility
 - Awareness: users need to know the importance of securing seed phrase

When It Goes Wrong

- 17B USD stolen in 10 years, largest share: compromised keys [\[link\]](#)
- Bad randomness
 - Coldcard, July 2026 [\[link\]](#)
 - Bug since 2021: software random generator instead of hardware
 - ~40–72 bits instead of 256 → ~1,600–2,050 BTC stolen in waves
 - Fix does not repair old seeds → new seed, move funds
 - Milk Sad 2023 / Profanity 2022: same pattern, 32 bits [\[link\]](#), [\[link\]](#)
- Oct 2026: Tampered Ledgers [\[link\]](#), [\[link\]](#)
 - CryptoBilis: authorized reseller, Malaysia / Indonesia / Philippines
 - ~72 - 86M USD stolen, hundreds of wallets (on-chain estimates)
 - Teardown (Mark Karpelès), second circuit board with antenna, cellular module, data eSIM
 - Reads the screen, recognizes the Ledger font, sends seed phrase out during setup
 - Secure Element protects the key – but not the screen
- Lessons: buy directly from the manufacturer

Best Practices Mnemonic

- When showing Metamask, I actually showed how not to do it
 - Write It Down: always write down the seed phrase, avoid digital storage unless it's encrypted. Maybe even: [addition](#)



- Use [metal](#) backups: durability against physical damage, store seed phrase on metal plate
- Maintain multiple backups: prevent loss due to accidents or natural disasters
 - Or split with Shamir Secret Sharing (e.g., 2-of-3)
- Never ever type the seed into a website, no legitimate support ever asks for seed phrase
- (New) Buy hardware from the manufacturer
- Separate wallets: hot (daily), cold (savings), burner (testnet / CT)
- Test with small funds, test recovery before depositing; revoke old approvals
- **Educate Yourself Continuously**