# Bitcoin Transactions

**Goal:** enable digital payments between untrusted parties with no central authority (no companies, governments, etc).

**Ingredients of a Bitcoin transaction:**

(1) sender
(2) receiver
} specified by "public key"

(3) amount to transfer (in BTC)
[currently 1BTC ≈ 10K USD]

(4) pointer to last transaction with these coins

(5) transaction fee

**Valid transaction:**

- cryptographically signed by sender
- sender = owner of coins

**P2P Network:** used to broadcast all transactions to everybody.

# The Blockchain

**Ledger**: history of all transactions authorized thus far. (grouped into "blocks")

**Ingredients of a block**: ① some transactions (typically 1000 - 2000) ② reference to preceding block ③ a "nonce"

**Blockchain**: 


**Key idea**: ① incentivize "miners" to add blocks (+ collect reward in BTC) BUT ② make it hard to do so ("proof of work")

how BTCs get "minted"!

# Mining

**Cryptographic hash function:** arbitrary sequence of information → [SHA-256] → gobbledygook (0s and 1s)

[in practice, SHA-256 indistinguishable from a random function]

Call a block b **eligible** if SHA-256(b) starts with 80 zeroes.

**Bitcoin mining:** ① try to find eligible block b ② broadcast it across P2P network
=> gets appended to blockchain

**Reward:** 6.25 BTC (+ transaction fees)

**Belief:** no algorithm better than random guessing. => on average, succeed every $2^{80}$ tries

**Why 80?:** want new block added every 10 minutes on average.

# Forks

**Issue:** two different eligible blocks discovered at roughly the same time ⇒ fork.


**Specified behavior:** interpret authorized transactions as those in the longest chain (break ties in favor of block you heard about it first).

**Consequence:** regard a transfer of funds as complete only after transaction added to block chain and extended by several more blocks (e.g., 6).

# Forking Attacks

if Bob waits for k blocks to be added, drops to $\alpha^{k+2}$

**Good news:** Sybil attacks (i.e., create multiple IDs) ineffective.

**Double-spend attack:**


success probability = $\alpha^3$
[$\alpha$ = fraction of overall computational power possessed by Alice]

- Alice pays Bob in block $b_1$
- block $b_2$ added after $b_1$
- Alice tries to extend $b_0$ 3 times before anyone extends $b_2$ (=) would orphan $b_1, b_2$)

**51% Attack:** if $\alpha > \frac{1}{2}$, miner can act like a centralized authority.

- e.g., can freeze assets of any user

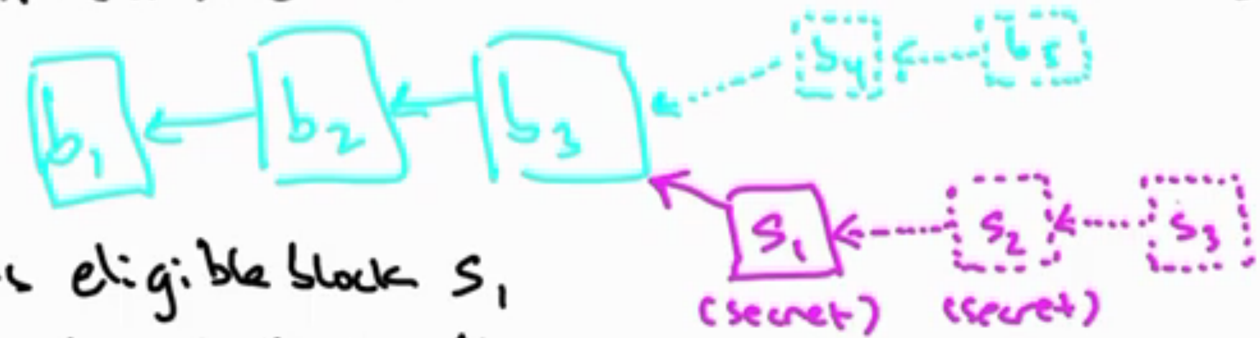
# Selfish Mining

(Eyal/Gün Sirer 2014)

**Second genre of attack:** block withholding. (don't tell other miners about your eligible block)

**Intuition:** withhold block ⇒ trick other miners into working on wrong cryptopuzzle.

**Strategy:**




- Alice finds eligible block $s_1$
- privately try to extend $s_1$ with another block $s_2$
- if $b_4$ (extending $b_3$) announced first, Alice restarts
- if $s_2$ found first: ① Alice mines secret chain until "lead" drops to 1
  ② announce entire secret chain

**Theorem:** Selfish mining better than honest mining when $\alpha > 1/3$.

(α: fraction of overall computational power controlled by miner)