CryptoCurrency and Blockchain (2)

金融科技導論

陳君明

jmchen@crypto.tw

國立臺灣大學 National Taiwan University

密碼貨幣市值

http://coinmarketcap.com 2018.12.10 11:00

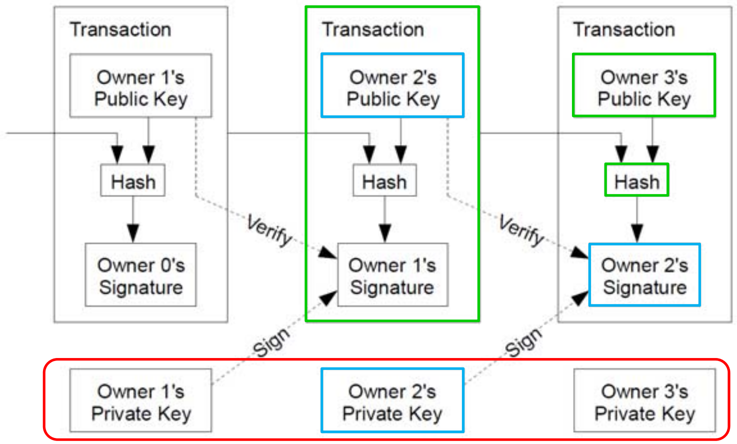
Cryptocurrencies: 2068 • Markets: 15712 • Market Cap: $114,297,957,903 • 24h Vol: $13,979,798,812 • BTC Dominance: 55.0%

CryptoCurrency Market Capitalizations

#	Name	Market Cap	Price	Volume (24h)	Circulating Supply	Change (24h)	Price Graph (7d)
1	Bitcoin	$62,874,805,945	$3,610.33	$4,944,445,835	17,415,250 BTC	3.27%	
2	XRP	$12,784,135,647	$0.312365	$429,530,185	40,926,963,305 XRP *	1.42%	
3	Ethereum	$9,765,721,168	$94.17	$1,890,208,918	103,704,450 ETH	1.57%	
4	Stellar	$2,378,240,524	$0.124094	$176,421,225	19,164,800,956 XLM *	3.87%	
5	Bitcoin Cash	$1,892,831,301	$108.15	$121,797,207	17,502,000 BCH	3.73%	
6	Tether	$1,888,454,617	$1.02	$3,253,068,343	1,856,421,736 USDT *	0.17%	
7	EOS	$1,801,902,222	$1.99	$834,928,229	906,245,118 EOS *	8.92%	
8	Bitcoin SV	$1,754,752,179	$100.26	$76,052,760	17,501,686 BSV	3.12%	
9	Litecoin	$1,521,556,117	$25.57	$466,782,230	59,504,167 LTC	1.59%	
10	TRON	$896,361,021	$0.013531	$60,183,036	66,245,700,961 TRX *	0.29%	

Core Concepts

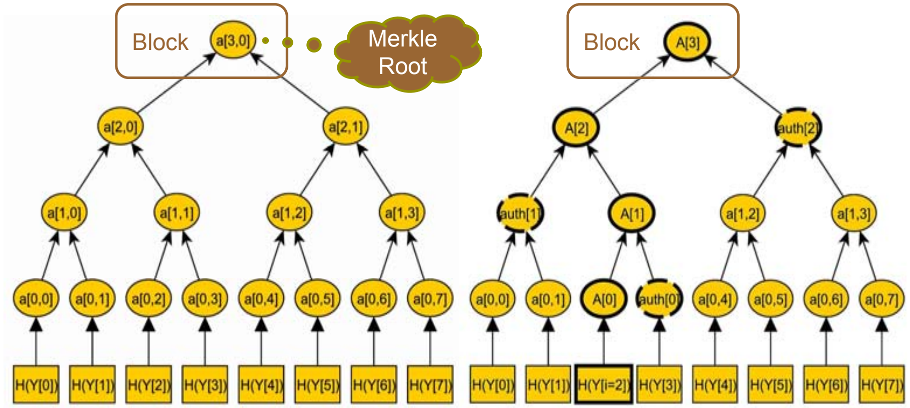
Bitcoin Transactions 交易



Must be protected very well!!!

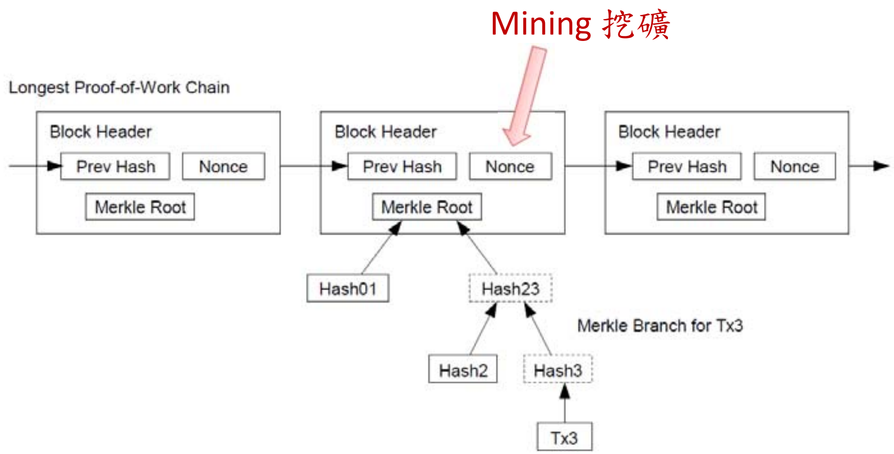
http://bitcoin.org/bitcoin.pdf 中本聰

Merkle Tree / Hash Tree



http://commons.wikimedia.org/wiki/File:MerkleTree1.jpg http://commons.wikimedia.org/wiki/File:MerkleTree2.jpg

Block Chain



http://bitcoin.org/bitcoin.pdf 中本聰

RSA

RSA Cryptosystem

- Ronald Rivest, Adi Shamir and Leonard Adleman proposed the asymmetric (public-key) RSA cryptosystem in 1977
- Until now, RSA is the most widely use asymmetric cryptosystem although elliptic curve cryptography (ECC) becomes increasingly popular
- RSA is mainly used for two applications
  - Transport of (symmetric) keys
  - Digital signatures

Encryption and Decryption

- RSA operations are done over the integer ring $\boldsymbol{Z}_n$ (i.e., arithmetic modulo n), where $n = p \times q$, with p, q being large primes

Definition
Given the public key $(n, e) = k_{pub}$ and the private key $d = k_{pr}$,

$$y = e_{k_{pub}}(x) = x^e \mod n$$
$$x = d_{k_{pr}}(y) = y^d \mod n$$

where $x, y \in \boldsymbol{Z}_n$.
Call $e_{k_{pub}}(\,)$ the encryption and $d_{k_{pr}}(\,)$ the decryption operation.

Key Generation

Algorithm: RSA Key Generation
Output: public key: $k_{pub} = (n, e)$ and private key $k_{pr} = d$
1. Choose two large primes p, q
2. Compute $n = p \times q$
3. Compute $\Phi(n) = (p-1)(q-1)$
4. Select the public exponent $e \in \{1, 2, \ldots, \Phi(n) - 1\}$ such that $\gcd(e, \Phi(n)) = 1$
5. Compute the private key d such that $d \times e \equiv 1 \mod \Phi(n)$
6. **RETURN** $k_{pub} = (n, e)$, $k_{pr} = d$

- Remarks:
  - Choosing two large, distinct primes p, q (in Step 1) is non-trivial
  - $\gcd(e, \Phi(n)) = 1$ ensures that e has an inverse and, thus, that there is always a private key d

Example

ALICE	**BOB**
Message $x = 4$	1. Choose $p = 3$ and $q = 11$
	2. Compute $n = p \times q = 33$
	3. $\Phi(n) = (3-1)(11-1) = 20$
	4. Choose $e = 3$
	5. $d \equiv e^{-1} \equiv 7 \mod 20$
$\xleftarrow{K_{pub} = (33, 3)}$	
$y = x^e \equiv 4^3 \equiv 31 \mod 33$	
$\xrightarrow{y = 31}$	
	$y^d = 31^7 \equiv 4 = x \mod 33$

Proof of Decryption

- There exists $k \in \boldsymbol{Z}$ such that $e\,d = 1 + k\,\phi(N)$
  - If $\gcd(x, p) = 1$
    - We have $x^{p-1} \equiv 1 \pmod p$ by Fermat's Little Theorem
    - Taking $k(q-1)$-th power and multiplying with x yields
      $$x^{1+k(p-1)(q-1)} \equiv x \pmod p \qquad (*)$$
  - If $\gcd(x, p) = p$, then $x \equiv 0 \pmod p$ and $(*)$ is valid again
- Hence $x^{ed} \equiv x \pmod p$ in both cases, and by a similar argument we have $x^{ed} \equiv x \pmod q$
- Since p and q are distinct primes, the CRT leads to
  $$y^d \equiv (x^e)^d = x^{ed} = x^{1+k(p-1)(q-1)} \equiv x \pmod N$$

Square-and-Multiply

- Computes x^{26} without modulo reduction
- Binary representation of exponent:

$$26 = (1, 1, 0, 1, 0)_2 = (h_4, h_3, h_2, h_1, h_0)_2$$

Step		Binary exponent	Op	Comment
1	$x = x^1$	$(1)_2$		Initial setting, h_4 processed
1a	$(x^1)^2 = x^2$	$(10)_2$	SQ	Processing h_3
1b	$x^2 \times x = x^3$	$(11)_2$	MUL	$h_3 = 1$
2a	$(x^3)^2 = x^6$	$(110)_2$	SQ	Processing h_2
2b	-	$(110)_2$	-	$h_0 = 0$
3a	$(x^6)^2 = x^{12}$	$(1100)_2$	SQ	Processing h_1
3b	$x^{12} \times x = x^{13}$	$(1101)_2$	MUL	$h_1 = 1$
4a	$(x^{13})^2 = x^{26}$	$(11010)_2$	SQ	Processing h_0
4b	-	$(11010)_2$	-	$h_0 = 0$

Square-and-Multiply

- Basic principle: Scan exponent bits from left to right and square/multiply operand accordingly

Algorithm: Square-and-Multiply for x^H mod n

Input: Exponent H, base element x, modulus n

Output: $y = x^H$ mod n

1. Determine binary representation $H = (h_t, h_{t-1}, \dots, h_0)_2$
2. Initialization: $y = x$
3. **FOR** $i = t-1$ **TO** 0
4. $\quad y = y^2 \bmod n$
5. $\quad$ **IF** $h_i = 1$ **THEN**
6. $\quad\quad y = y \times x \bmod n$
7. **RETURN** y

Small Public Exponent

- Choosing a small public exponent e does not weaken the security of RSA
- A small public exponent improves the speed of the RSA encryption significantly

Public Key	e as binary string	#MUL + #SQ
$2^1 + 1 = 3$	$(11)_2$	1 + 1 = 2
$2^4 + 1 = 17$	$(1\,0001)_2$	4 + 1 = 5
$2^{16} + 1 = 65537$	$(1\,0000\,0000\,0000\,0001)_2$	16 + 1 = 17

Speed-Up Techniques

- Modular exponentiation is computationally intensive
- Even with the square-and-multiply algorithm, RSA can be quite slow on constrained devices such as smart cards
- Some important tricks
  - Short public exponent e
  - Chinese Remainder Theorem (CRT)
  - Exponentiation with pre-computation

RSA Signature Scheme

Alice

Bob

$K_{pr} = d$

$K_{pub} = (n, e)$

$\xleftarrow{K_{pub}}$

Compute signature:

$s = sig_{K_{pr}}(x) \equiv x^d \bmod n$

$\xleftarrow{(x,\ s)}$

Verify signature:

$x' = ver_{K_{pub}}(x) \equiv s^e \bmod n$

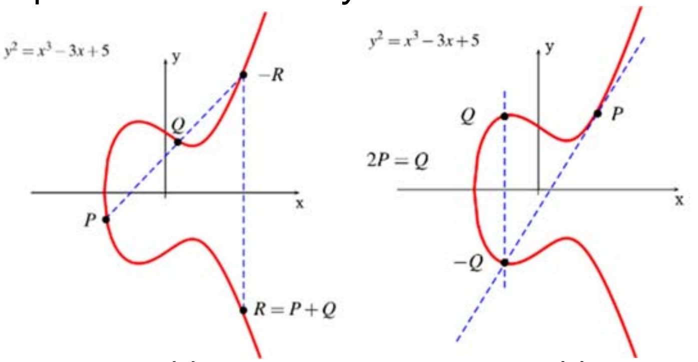
If $x' = x \ \rightarrow$ valid signature

If $x' \neq x \ \rightarrow$ invalid signature

Elliptic Curves

Elliptic Curve 橢圓曲線

- The rich and deep theory of Elliptic Curves has been studied by mathematicians over 150 years
- Elliptic Curve over $\boldsymbol{R}$: $y^2 = x^3 + ax + b$



Point Addition　　Point Doubling

Image Courtesy: http://www.embedded.com/design/safety-and-security/4396040/An-Introduction-to-Elliptic-Curve-Cryptography

質數體 (Prime Field) 上的曲線

Addition:

$(x_3, y_3) = (x_1, y_1) + (x_2, y_2)$

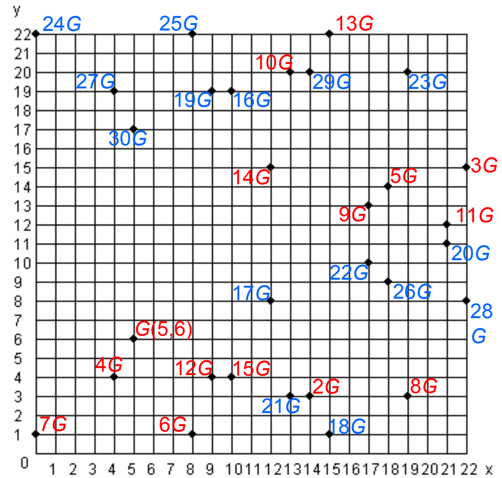
Doubling:

$(x_3, y_3) = [2]\ (x_1, y_1)$

$$s = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} \bmod p & \text{(addition)} \\ \dfrac{3x_1^2 + a}{2y_1} \bmod p & \text{(doubling)} \end{cases}$$

$$x_3 = s^2 - x_1 - x_2 \mod p$$

$$y_3 = s(x_1 - x_3) - y_1 \mod p$$



$y^2 = x^3 + 5x + 1$ over F_{23}

30 solutions

Example

- Given $E: y^2 = x^3 + 2x + 2 \mod 17$ and point $P = (5, 1)$

Goal: Compute $2P = P + P = (5, 1) + (5, 1) = (x_3, y_3)$

$$s = \frac{3x_1^2 + a}{2y_1} = (2 \cdot 1)^{-1}(3 \cdot 5^2 + 2) = 2^{-1} \cdot 9 \equiv 9 \cdot 9 \equiv 13 \bmod 17$$

$$x_3 = s^2 - x_1 - x_2 = 13^2 - 5 - 5 = 159 \equiv 6 \bmod 17$$

$$y_3 = s(x_1 - x_3) - y_1 = 13(5 - 6) - 1 = -14 \equiv 3 \bmod 17$$

Finally $2P = (5, 1) + (5, 1) = (6, 3)$

Example

- The points on an elliptic curve and the point at infinity $\mathcal{O}$ form cyclic subgroups

$2P = (5, 1) + (5, 1) = (6, 3)$	$11P = (13, 10)$
$3P = 2P + P = (10, 6)$	$12P = (0, 11)$
$4P = (3, 1)$	$13P = (16, 4)$
$5P = (9, 16)$	$14P = (9, 1)$
$6P = (16, 13)$	$15P = (3, 16)$
$7P = (0, 6)$	$16P = (10, 11)$
$8P = (13, 7)$	$17P = (6, 14)$
$9P = (7, 6)$	$18P = (5, 16)$
$10P = (7, 11)$	$19P = \mathcal{O}$

This elliptic curve has order $\#E = |E| = 19$ since it contains 19 points in its cyclic group.

Double and Add

Example: $26P = (11010_2)P = (d_4d_3d_2d_1d_0)_2\, P$.

Step		
#0	$P = \mathbf{1}_2 P$	inital setting
#1a	$P+P = 2P = \mathbf{10}_2 P$	DOUBLE (bit d_3)
#1b	$2P+P = 3P = 10^2P + 1_2P = \mathbf{11}_2 P$	ADD (bit $d_3 = 1$)
#2a	$3P+3P = 6P = 2(11_2P) = \mathbf{110}_2 P$	DOUBLE (bit d_2)
#2b		no ADD ($d_2 = 0$)
#3a	$6P+6P = 12P = 2(110_2P) = \mathbf{1100}_2 P$	DOUBLE (bit d_1)
#3b	$12P+P = 13P = 1100_2P + 1_2P = \mathbf{1101}_2 P$	ADD (bit d_1=1)
#4a	$13P+13P = 26P = 2(1101_2P) = \mathbf{11010}_2 P$	DOUBLE (bit d_0)
#4b		no ADD ($d_0 = 0$)

Bitcoin和Ethereum使用的曲線

The elliptic curve domain parameters over $\mathbb{F}_p$ associated with a Koblitz curve `secp256k1` are specified by the sextuple $T = (p, a, b, G, n, h)$ where the finite field $\mathbb{F}_p$ is defined by:

p = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE FFFFFC2F

$= 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$

256-bit prime

The curve E: $\boxed{y^2 = x^3 + ax + b}$ over $\mathbb{F}_p$ is defined by:

a = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

b = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000007

橢圓曲線 secp256k1

https://en.bitcoin.it/wiki/Secp256k1

The base point G in compressed form is:

G = 02 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B 16F81798

and in uncompressed form is:

G = 04 79BE667E F9DCBBAC 55A06295 CE870B07 029BFCDB 2DCE28D9 59F2815B 16F81798 483ADA77 26A3C465 5DA4FBFC 0E1108A8 FD17B448 A6855419 9C47D08F FB10D4B8

Finally the order n of G and the cofactor are:

n = FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFE BAAEDCE6 AF48A03B BFD25E8C D0364141

256-bit prime

h = 01

Key Pairs 金鑰對

- The base point G is fixed on the given Elliptic Curve
- $P = [m]\,G$
  - Given m, it is **easy and fast** to find the point P
    - Using "double and add" for scalar multiplication
  - Given P, it is **extremely hard** to find the integer m
    - Elliptic Curve Discrete Logarithm Problem (橢圓曲線離散對數問題)
  - A randomly generated integer m is a **private key**
    - A private key is used to sign Bitcoin transactions with ECDSA
  - The point P is the **public key** corresponding to m
    - A public key is used by other nodes to verify Bitcoin transactions
    - **A Bitcoin address is the hash value of a public key P**

NIST Curve Standards in FIPS 186

Table D-1: Bit Lengths of the Underlying Fields of the Recommended Curves

Bit Length of n	Prime Field	Binary Field
161 – 223	**len**(p) = 192	m = 163
224 – 255	**len**(p) = 224	m = 233
256 – 383	**len**(p) = 256	m = 283
384 – 511	**len**(p) = 384	m = 409
≥ 512	**len**(p) = 521	m = 571

NIST Curves over Prime Fields

D.1.2 Curves over Prime Fields

For each prime p, a pseudo-random curve

$$E: y^2 \equiv x^3 - 3x + b \pmod{p}$$

of prime order n is listed[4]. (Thus, for these curves, the cofactor is always $h = 1$.) The following parameters are given:

- The prime modulus p
- The order n
- The 160-bit input seed *SEED* to the SHA-1 based algorithm (i.e., the domain parameter seed)
- The output c of the SHA-1 based algorithm

[4] The selection $a \equiv -3$ for the coefficient of x was made for reasons of efficiency; see IEEE Std 1363-2000.

- The coefficient b (satisfying $b^2\, c \equiv -27 \pmod{p}$)
- The base point x coordinate G_x
- The base point y coordinate G_y

The integers p and n are given in decimal form; bit strings and field elements are given in hexadecimal.

Curve P-256

D.1.2.3 Curve P-256

p = 115792089210356248762697446949407573530086143415290314195533631308867097853951

n = 115792089210356248762697446949407573529996955224135760342422259061068512044369

SEED = c49d3608 86e70493 6a6678e1 139d26b7 819f7e90

c = 7efba166 2985be94 03cb055c 75d4f7e0 ce8d84a9 c5114abc af317768 0104fa0d

b = 5ac635d8 aa3a93e7 b3ebbd55 769886bc 651d06b0 cc53b0f6 3bce3c3e 27d2604b

G_x = 6b17d1f2 e12c4247 f8bce6e5 63a440f2 77037d81 2deb33a0 f4a13945 d898c296

G_y = 4fe342e2 fe1a7f9b 8ee7eb4a 7c0f9e16 2bce3357 6b315ece cbb64068 37bf51f5

NIST Curves over Prime Fields

P-192: $p = 2^{192} - 2^{64} - 1$, $a = -3$, $h = 1$,
$b =$ 0x 64210519 E59C80E7 0FA7E9AB 72243049 FEB8DEEC C146B9B1
$n =$ 0x FFFFFFFF FFFFFFFF FFFFFFFF 99DEF836 146BC9B1 B4D22831

P-224: $p = 2^{224} - 2^{96} + 1$, $a = -3$, $h = 1$,
$b =$ 0x B4050A85 0C04B3AB F5413256 5044B0B7 D7BFD8BA 270B3943 2355FFB4
$n =$ 0x FFFFFFFF FFFFFFFF FFFFFFFF FFFF16A2 E0B8F03E 13DD2945 5C5C2A3D

P-256: $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$, $a = -3$, $h = 1$,
$b =$ 0x 5AC635D8 AA3A93E7 B3EBBD55 769886BC 651D06B0 CC53B0F6 3BCE3C3E 27D2604B
$n =$ 0x FFFFFFFF 00000000 FFFFFFFF FFFFFFFF BCE6FAAD A7179E84 F3B9CAC2 FC632551

P-384: $p = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$, $a = -3$, $h = 1$,
$b =$ 0x B3312FA7 E23EE7E4 988E056B E3F82D19 181D9C6E FE814112 0314088F 5013875A C656398D 8A2ED19D 2A85C8ED D3EC2AEF
$n =$ 0x FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF C7634D81 F4372DDF 581A0DB2 48B0A77A ECEC196A CCC52973

P-521: $p = 2^{521} - 1$, $a = -3$, $h = 1$,
$b =$ 0x 00000051 953EB961 8E1C9A1F 929A21A0 B68540EE A2DA725B 99B315F3 B8B48991 8EF109E1 56193951 EC7E937B 1652C0BD 3BB1BF07 3573DF88 3D2C34F1 EF451FD4 6B503F00
$n =$ 0x 000001FF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFA 51868783 BF2F966B 7FCC0148 F709A5D0 3BB5C9B8 899C47AE BB6FB71E 91386409

Security Level

(NIST) SP 800-57 Part 1

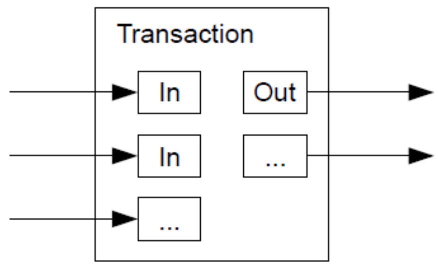
Bits of security	Symmetric key algorithms	Finite Field Cryptography (FFC, e.g., DSA, D-H)	Integer Factorization Cryptography (IFC, e.g., RSA)	Elliptic Curve Cryptography (ECC, e.g., ECDSA)
80	2TDEA*	$L = 1024$ $N = 160$	$k = 1024$	$f = 160\text{-}223$
112	3TDEA	$L = 2048$ $N = 224$	$k = 2048$	$f = 224\text{-}255$
128	AES-128	$L = 3072$ $N = 256$	$k = 3072$	$f = 256\text{-}383$
192	AES-192	$L = 7680$ $N = 384$	$k = 7680$	$f = 384\text{-}511$
256	AES-256	$L = 15360$ $N = 512$	$k = 15360$	$f = 512+$

* The assessment of at least 80-bits of security for 2TDEA is based on the assumption that an attacker has no more than 2^{40} matched plaintext and ciphertext blocks ([ANSX9.52], Annex B).

Transactions

Combining & Splitting Value

- “To allow value to be split and combined, transactions contain multiple inputs and outputs.”



http://bitcoin.org/bitcoin.pdf 中本聰

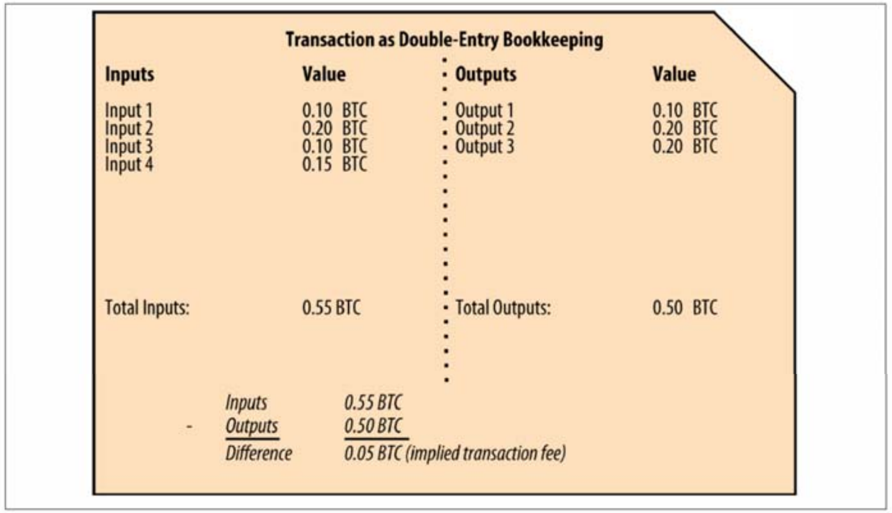


Figure 2-3. Transaction as double-entry bookkeeping

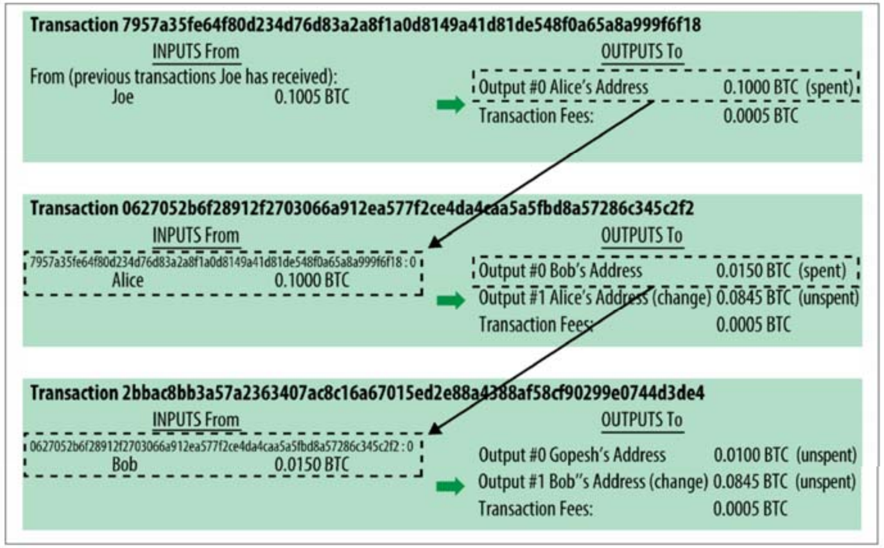


Figure 2-4. A chain of transactions, where the output of one transaction is the input of the next transaction

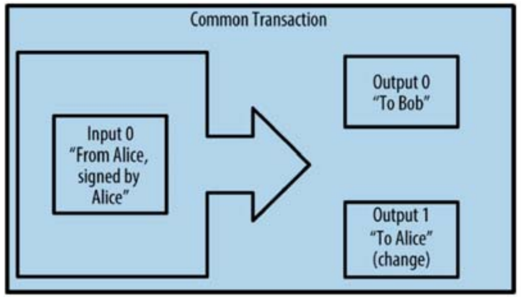


Figure 2-5. Most common transaction

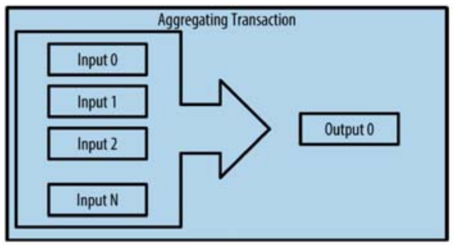


Figure 2-6. Transaction aggregating funds

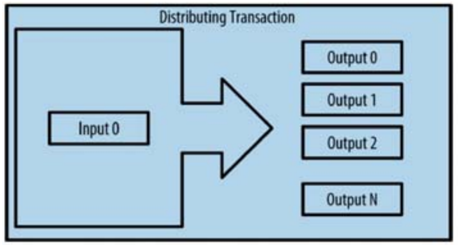


Figure 2-7. Transaction distributing funds

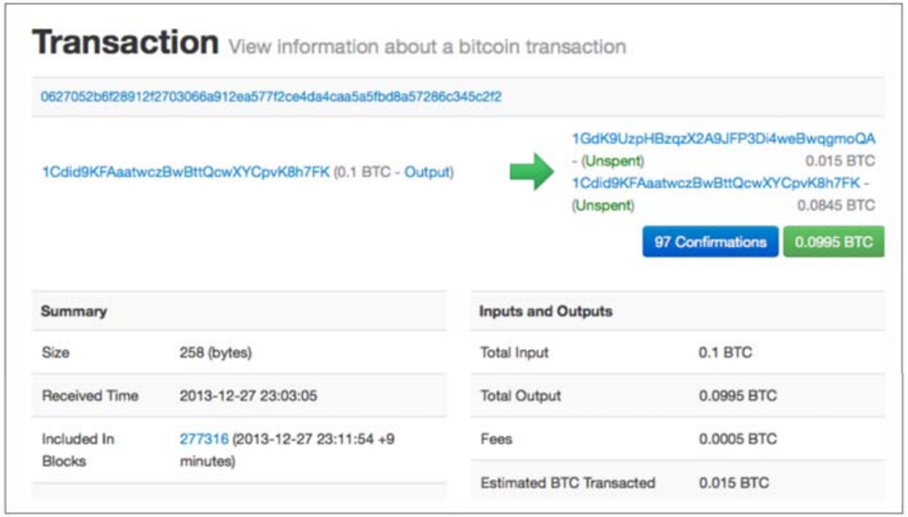


Figure 2-8. Alice's transaction to Bob's Cafe

Transaction Data

- {"hash":"7c4025... "
  - the hash of the remainder of the transaction (Data)
- "ver":1,
  - version 1 of the Bitcoin protocol
- "vin_sz":1,
  - one input
- "vout_sz":1,
  - one output
- "lock_time":0,
  - transaction is finalized immediately
- "size":224,
  - size (in bytes) of the transaction
  - not transaction amount

- "in":[
- {"prev_out":
- {"hash":"2007ae...",
  - where the money from
  - hash of previous transaction
- "n":0},
  - it is the first output from that transaction
- "scriptSig":"304502... 042b2d..."}],
  - signature of the person sending the money
  - the corresponding public key followed by a space
- "out":[
- {"value":"0.31900000",
  - the value of the output
- "scriptPubKey":"OP_DUP OP_HASH160 a7db6f OP_EQUAL
- VERIFY OP_CHECKSIG"}]}
  - Bitcoin's scripting language
  - Bitcoin address of the intended recipient (a7db6f)

http://www.michaelnielsen.org/ddi/how-the-bitcoin-protocol-actually-works

Keys, Addresses, Wallets

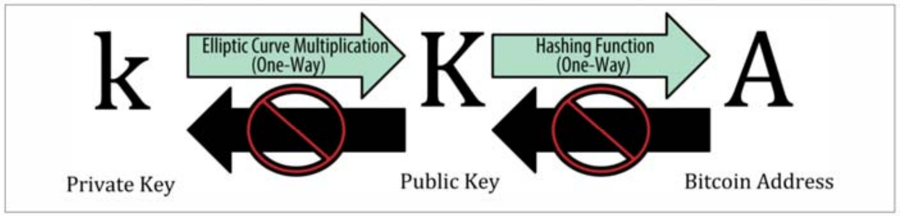


Figure 4-1. Private key, public key, and bitcoin address

The size of bitcoin's private key space, 2^{256} is an unfathomably large number. It is approximately 10^{77} in decimal. The visible universe is estimated to contain 10^{80} atoms.

"Mastering Bitcoin" by Andreas M. Antonopoulos

Bitcoin Address

- Address = RIPEMD160(SHA256(public key representation))
- Example
  - ECDSA private key = 18E14A7B6A307F426A94F8114701E7C8E774E7F9A47E2C2035DB29A206321725
  - Public key *P* = 04 50863AD64A87AE8A2FE83C1AF1A8403CB53F53E486D8511DAD8A04887E5B235
    22CD470243453A299FA9E77237716103ABC11A1DF38855ED6F2EE187E9C582BA6
  - SHA256(*P*) = 600FFE422B4E00731A59557A5CCA46CC183944191006324A447BDB2D98D4B408
  - RIPEMD160(SHA256(*P*)) = 010966776006953D5567439E5E39F86A0D273BEE
  - Address (Base58Check encoded): 16UwLL9Risc3QfPqBUvKofHmBQ7wMtjvM
  - https://en.bitcoin.it/wiki/Technical_background_of_version_1_Bitcoin_addresses#How_to_create_Bitcoin_Address
- Base58 is a set of lower and capital letters and numbers without (0, O, l, I), i.e., 0 (number zero), O (capital o), l (lower L), I (capital i)

Paper Wallets

Figure 4-14. An example of a simple paper wallet from bitaddress.org

Paper Wallets

Figure 4-15. An example of an encrypted paper wallet from bitaddress.org. The passphrase is "test."

CoolWallet

第一代 CoolWallet :
英飛凌 Infineon
SLE97 安全晶片

CoolBitX獲得日本金融集團SBI策略性投資

f 分享 分享 留言 列印 存新聞 A- A+

2018-03-02 15:10 經濟日報 邱世婷 讚 0 分享

台灣區塊鏈新創公司–庫幣科技(CoolBitX) 宣佈獲得日本金融集團SBI策略性投資。庫幣科技(CoolBitX)主要研發與產製可離線儲存加密數位貨幣的軟硬體，透過實體卡片與手機APP，協助加密數位貨幣持有者，降低駭客襲擊的損失風險。庫幣科技(CoolBitX)的產品CoolWallet是全球最薄的實體比特幣錢包，於全球銷售已超過10萬張，遍及歐美亞洲等地區，其技術與產品令日本SBI金融集團驚豔，受邀加入SBI加密數位貨幣生態系，成為生態系中主要提供離線儲存加密貨幣解決方案的要角；此外，SBI亦宣佈完成對庫幣科技(CoolBitX)的策略性投資。

庫幣科技(CoolBitX)技術領先全球 首創最輕薄加密數位貨幣硬體錢包裝置 兼顧安全與便利性

https://money.udn.com/money/story/10860/3008668

資訊安全 區塊鏈

搭區塊鏈熱潮，台灣駭客年會HITCON Community推年會限定代幣、硬體錢包

by 張庭瑜 2018.07.27

https://www.bnext.com.tw/article/50035/hitcon-cmt-2018-blockchain

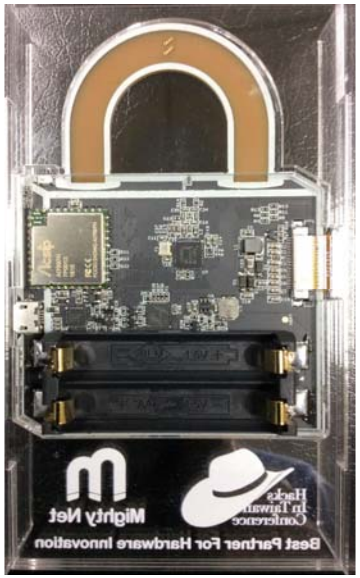
第14屆HITCON社群場新嘗試，專用數位貨幣及實境挑戰遊戲

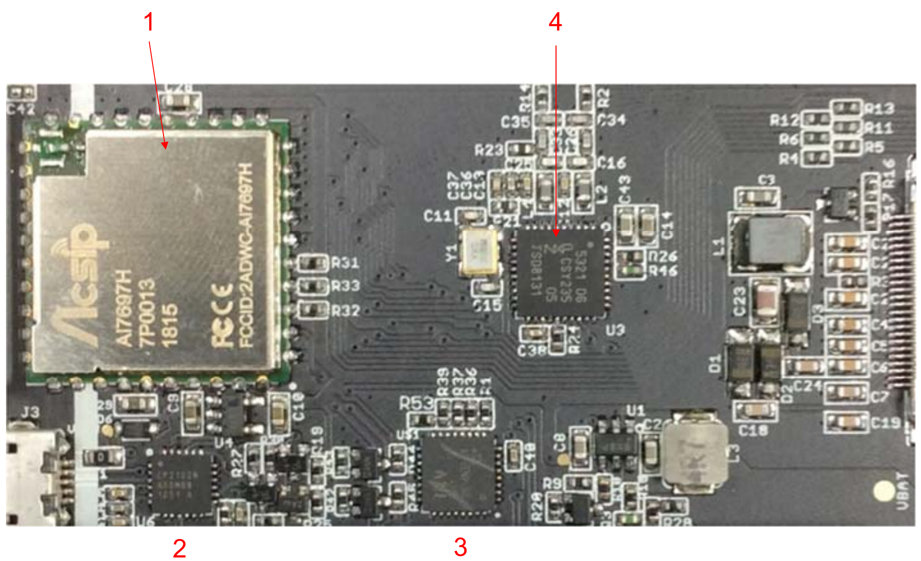
臺灣駭客年會社群場邁入第14屆，不只搭上區塊鏈議題，更強調了當中技術的儲存、驗證、傳遞，都與資訊安全息息相關。本屆活動還設計了專用加密貨幣HITCON Token，並推出實境挑戰遊戲HITCON Hackdoor。

讚 4.9萬 按讚加入iThome粉絲團 讚 416 分享 G+

文/ 羅正漢 | 2018-07-30 發表

https://www.ithome.com.tw/news/124861





HITCON Enterprise 2014
台灣駭客年會 企業場

私鑰數據庫?


比特币 (Bitcoin)

比特币「私钥数据库」是怎么回事？

1：All bitcoin private keys

2：比特币私钥数据库

2 条评论 分享

查看全部 4 个回答

知乎用户

10 人赞同

转载自贴吧 原地址 那些说比特币算法可以被轻易破解的同学

先说比特币地址和私钥，你必须要明白比特币的加密学原理是基于椭圆曲线加密算法的，具体来说是secp256k1

比特币地址和私钥是由ECDSA椭圆曲线加密算法计算出来的，由ECDSA私钥计算出我们常用的Bitcoin-qt格式比特币地址需要有十个步骤

https://www.zhihu.com/question/23608006/answer/25141783

誤解：加密?!

- Bitcoin protocol 沒有「加密」，僅數位簽章
  - 中本聰論文的全文無任何 encrypt / encryption，而 sign / signing / signature 出現 12 次
- 許多文章強調 Bitcoin 以橢圓曲線密碼系統對交易資料進行加密保護，此為錯誤敘述
- 保護私鑰可能需使用加密，但它不屬於比特幣協定，由使用者錢包自行處理對私鑰的保護
- CryptoCurrency 的適當翻譯是「密碼貨幣」

金融科技發展策略白皮書 p.93

區塊鏈加密技術是數種技術集合的統稱，最底層的帳冊記錄數位化的資產，自創始後無縫且持續增加的交易資料，通過公私鑰簽章加解密方法，讓數位資產可以在不同持有人之間移轉並記入帳冊，交易無需在任何第三方的主持下發生，結合密碼學加密技術，依時間序定期或定量將交易資料寫入資料區塊（block）內，再通過驗證程序確認，最新驗證過的區塊，會附加到先前已驗證過的區塊之後，形成區塊鏈帳冊，由所有參與成員構成的網路節點內電腦協同一致維護及儲存，共識即確保成員同意那些交易是根據什麼程序來運作，這些數位資產將無法與帳冊分割使用，意即不能離鏈交易。

https://www.fsc.gov.tw/uploaddowndoc?file=/201605181357350.pdf&filedisplay=1050518%E9%87%91%E8%9E%8D%E7%A7%91%E6%8A%80%E7%99%BC%E5%B1%95%E7%AD%96%E7%95%A5%E7%99%BD%E7%9A%AE%E6%9B%B8.pdf&flag=doc