Goal of this lecture note

- Bitcoin Difficulty
- History of Bitcoin Difficulty
- Geometric vs Exponential Distribution
- Block Generation Speed
- Double Spending Attack Possibility
- Data Immutability

GIST

1 Bitcoin Difficulty

- Finding Good Block Summary
  - PoW Inequality is given by

$$F(\mathrm{BH}: nonce) < Target \qquad \text{PoW Ineq}$$

  - Find the first nonce that satisfies PoW Ineq.
  - Record it in to the block header, along with $Target$.
  - $Target$ specifies how difficult the puzzle was.

1 Bitcoin Difficulty

- Bitcoin Difficulty (*D*)

- The aim is to keep the average block generation time be 10 min.
  - Ex) The time span to mine 2016 blocks is set to take 2 weeks.
- Difficulty is adjusted for every 2016 block.
- Measure the time span, T [min], during which the past 2016 blocks were mined.
- Let T_D be 2 weeks [min], i.e., T_D = 2016x10 = 20160 [min].
- If T is different from T_D, adjust the Difficulty D :

$$D = D_{prev} \times \frac{T_D}{T}$$

In Bitcoin, initial D is set to 1 with 8 leading hexa zeros.

GIST

1 Bitcoin Difficulty

- *Target* is defined to be inversely proportional to Difficulty D.
  - The measured time T is used to update Difficulty D.
  - Finally, a new *Target* is thus given by

$$Target = Target_0 \frac{1}{D}$$

  - $Target_0$ is set to $2^{256-32} = 2^{224}$ the maximum allowed target.
  - With $Target = 2^{224}$, all good hashes are smaller than the target and have 32 leading zero bits.

GIST

1 Bitcoin Difficulty

- *Target* can be directly updated by combining the two equations:

$$Target = Target_{prev} \frac{T}{T_D}$$

- Target is any real number in the interval from 2^1 to 2^{224}.
- Minimum difficulty is 2^{224}.
- Maximum difficulty is 2^1.

1 Bitcoin Difficulty

- *Target* is inversely proportional to Difficulty.
  - The smaller *Target* is, the more difficult the puzzle is.

GIST

1 Bitcoin Difficulty

- What shall be *Target* if all good hashes has 10 leading hexadecimal zeros?
  - 40 binary zeros.
  - Target shall be $2^{256-40} = 2^{216}$.

GIST

1 Bitcoin Difficulty

- Conversion from Hashes to Hash Rate req'd.
  - Given a *Target*, one can calculate the number of hashes (avg) to make a single PoW success.
- Let $\mathrm{Log2Target} = \log_2(\mathit{Target})$.
  - Number of hashes needed per success is $2^{256\text{-log2target}}$.
  - The network hash rate req'd to keep 10 min per success is:

$$\text{Hash Rate Req'd} = \frac{2^{256-\log 2\mathrm{Target}}}{600}[\text{hashes / sec}]$$

1 Bitcoin Difficulty

- Given a Target, one can determine the network hash rate.
- Suppose you bring your own mining chip.
- You can determine your chance of winning a puzzle.
- It is the ratio of your hash rate to the total hash rate:

$$= \frac{\text{Your Hash Rate}}{\text{Your Hash Rate} + \text{Network Hash Rate}}$$

1 Bitcoin Difficulty

Example

- Suppose Target is 2^{204}.
- You want to join with 1 Tera hash/sec mining chip.
- What is your chance of winning a block?
  - The network hash power is $2^{256-\log 2\text{target}}/600 = 2^{52}/600 = 7.51e12$ [hash/sec].
  - The hash rate percentage is:

$$= \frac{\text{Your Hash Rate}}{\text{Your Hash Rate} + \text{Network Hash Rate}}$$

$$= \frac{1.00e12}{1.00e12 + 7.51e12}$$

$$= 11.8\%$$

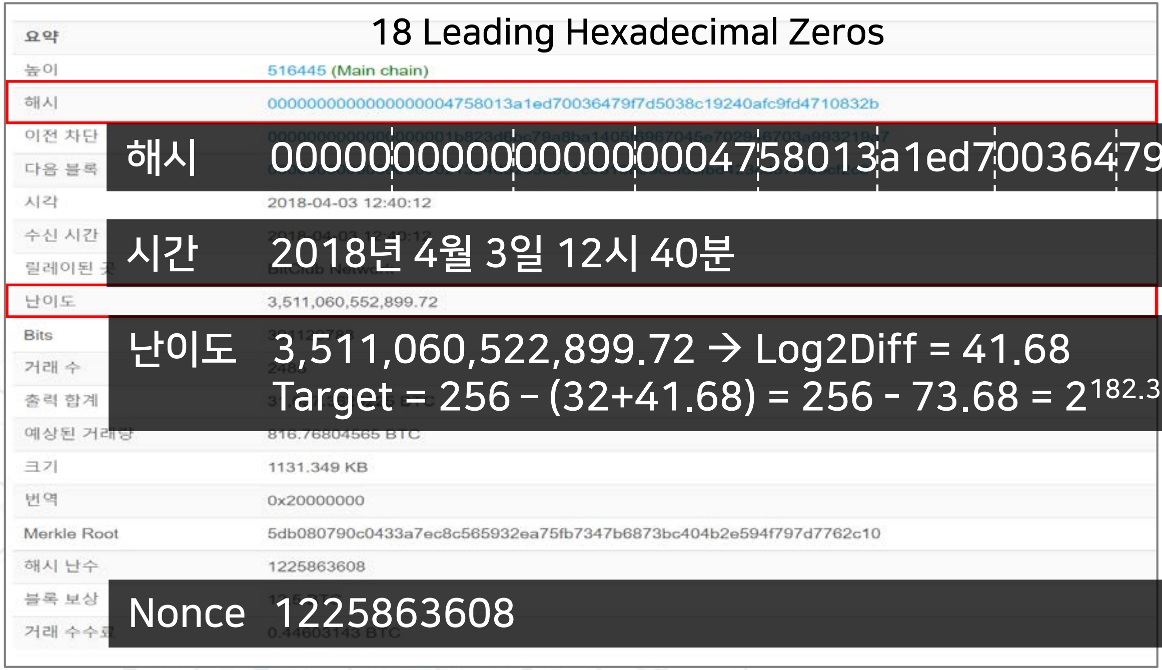
1 Bitcoin Difficulty

- *Target* specifies how *difficult* the puzzle was.
- It represents the number of hashes needed to solve the puzzle.
- It represents how many number of computers worked together at that time.
- Nonce is the proof.
- Nonce and Target are recorded in the block header along with the time-stamp.
- One can verify if the proof-of-work for the block was correctly done or not.

GIST

1 Bitcoin Difficulty

- 블록 높이 516445 비트코인 블록체인 내 깊이 값 513445에서의 블록들



18 Leading Hexadecimal Zeros

요약	
높이	516445 (Main chain)
해시	000000000000000000004758013a1ed70036479f7d5038c19240afc9fd4710832b
이전 차단	
다음 블록	
시각	2018-04-03 12:40:12
수신 시간	
릴레이된 곳	
난이도	3,511,060,552,899.72
Bits	
거래 수	
출력 합계	
예상된 거래량	816.76804565 BTC
크기	1131.349 KB
번역	0x20000000
Merkle Root	5db080790c0433a7ec8c565932ea75fb7347b6873bc404b2e594f797d7762c10
해시 난수	1225863608
블록 보상	
거래 수수료	

해시 000000000000000000004758013a1ed70036479f7d5036c19240afc9fd4710832b

시간 2018년 4월 3일 12시 40분

난이도 3,511,060,522,899.72 → Log2Diff = 41.68

Target = 256 – (32+41.68) = 256 - 73.68 = $2^{182.32}$

Nonce 1225863608

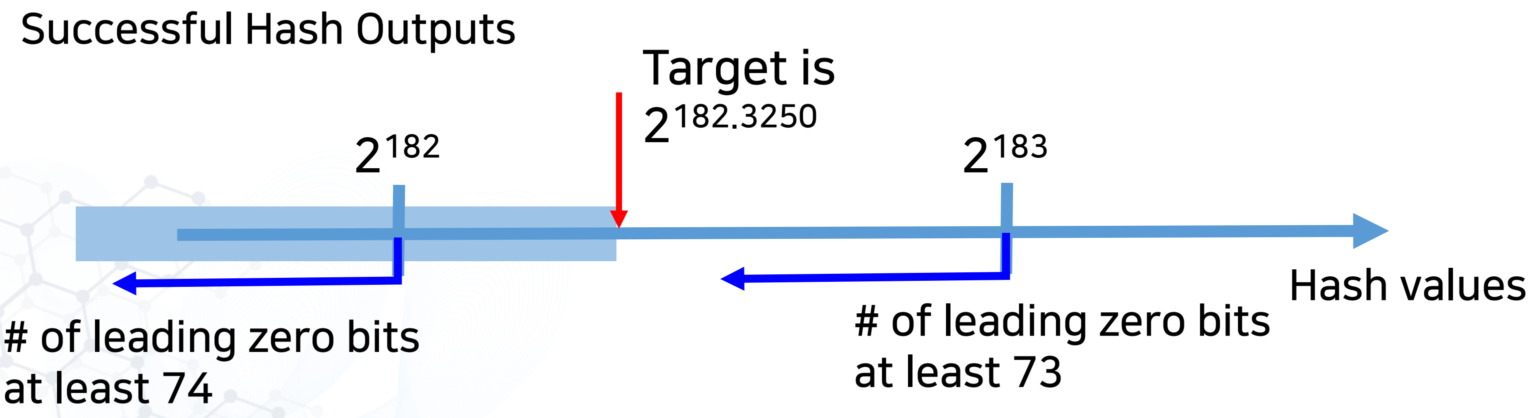
1 Bitcoin Difficulty

- Example of Difficulty and Target
  - Block #516445
  - BlockHash 0000 0000 0000 0000 0047 5801 … … … 832b
    - 18 hex zeros * 4 bits/hex + 1 bit = 72 + 1 = 73 bit zeros
  - Difficulty D is 3,511,060,552,899.7197 = 3.5e12
  - Target is Target_0 * (1/Difficulty)
    - Log2(D) = 41.68
    - Target = $2^{224.000}$ $2^{-41.675}$
      = $2^{182.325}$

GIST

1 Bitcoin Difficulty

Recall PoW Success is SHA Hash Output < Target



1 Bitcoin Difficulty

- Network Hash Rate : Block#516445
- With D=3.5e12, the probability p is about $2^{-(32+41.675)} = 2^{-73.6750}$.
- Then, it would take $1/p = 2^{73.6750}$ ~ 1.5080 e22 hashes to mine a single block.
- Dividing it by 10 min = 600 sec, the network hash rate is obtained, 25.1332 Exa hash/sec.

GIST

1 Bitcoin Difficulty

Example

- Calculate no. of Antminer S9s (14 Thps) you need bring to obtain hash power 0.01 %, given the network hash rate is 25 Exa hash/sec.
  - You need to bring at least 179 AS9 chips.

$$\text{Your Hash Rate} \geq \frac{0.01\% \ \ \text{Network Hash Rate}}{100\% - 0.01\%}$$

$$= \frac{1}{9999} 25\text{e}18 = 25\text{e}14$$

$$= 178.6(14\text{e}12)$$

GIST

1 Bitcoin Difficulty

Example

- Given the network hash rate is 25 Exa hash/sec, further questions can be asked.
  - What is the least number of mining chips working in the network?
  - How long does it take for a single mining chip to find a good PoW?

GIST

2 History of Bitcoin difficulty

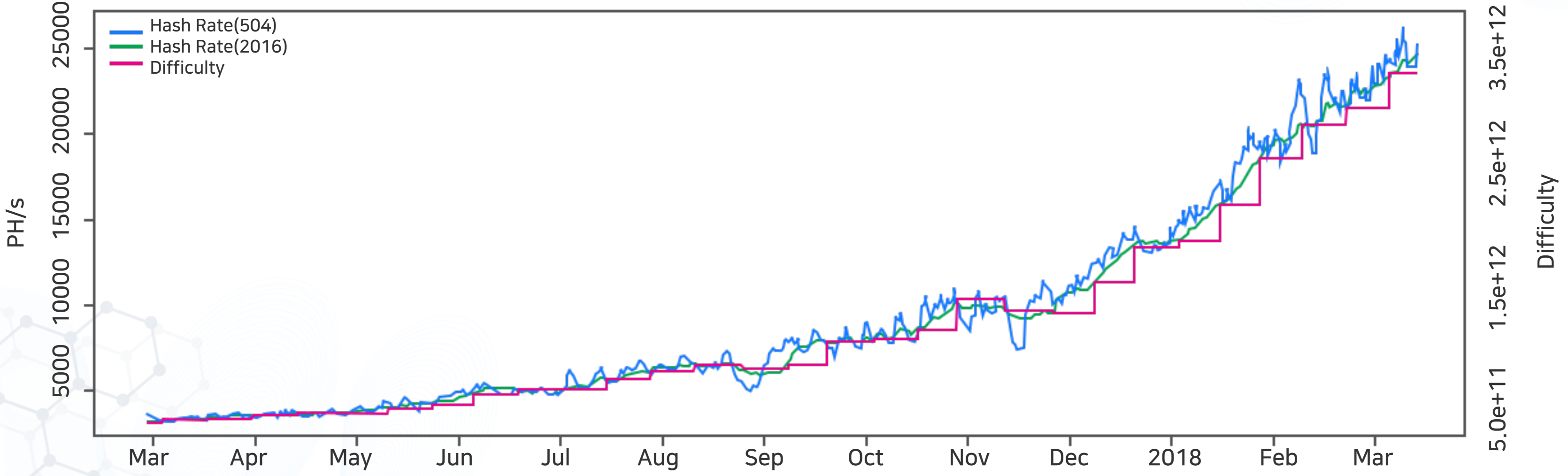
- Bitcoin Hash Rate vs Difficulty

Date	Difficulty	Hash Rate
Apr 01 2018	3,511,060,552,899	25,133,150,415 GH/s
Oct 04 2018	7,454,968,648,263	53,364,744,228 GH/s
Mar 24 2019	6,379,265,451,411	45,664,560,811 GH/s

출처: https://bitcoinwisdom.com/bitcoin/difficulty

2 History of Bitcoin difficulty

- Bitcoin Hash Rate vs Difficulty (Mar/17 ~ Apr 18)



출처: https://bitcoinwisdom.com/bitcoin/difficulty

3 Geometric vs Exponential Distribution

- Recall the Alone theorem, the probability of PoW success in k hashes is expressed with the per-hash success probability p.
- We now aim to improve it by embedding the concept of time into it.
- Then, we will get the *block generation* speed.
  - Given a unit time one can determine probabilistically the number of PoW successes or the number of blocks formed.

GIST

3 Geometric vs Exponential Distribution

- Theorem 1. (Alone) The CDF $P_{geom}(p, k)$, the probability of PoW successes in k hashes, can be expressed for k = 1, 2, 3, ⋯, as

$$P_p\{K \le k\} = 1 - P_p\{K > k\}$$
$$= 1 - (1 - p)^k$$

GIST

3 Geometric vs Exponential Distribution

- To the result of Theorem 1,
  we aim to put the time into consideration.
- For this, we define a new random variable S.
- *Recall* K is the random time index at which duration the PoW success occurs.

3 Geometric vs Exponential Distribution

- Geometric distribution(p) ~ Exponential distribution(p, T)
  - Let T here be the time-duration per single hash generation.
  - For a fast CPU, T be very small.
    - For example, 1 Tera hash/sec, T = 1e-12 sec/hash.

GIST

3 Geometric vs Exponential Distribution

- Geometric distribution(p) ~ Exponential distribution(p,T)
  - Let $S = KT$.
  - Then, S denotes the random time-epoch at which the PoW success occurs.

3 Geometric vs Exponential Distribution

- Geometric distribution(p) ~ Exponential distribution(p, T)
  - Let $S = KT$.
  - Then, S denotes the random time-epoch at which the PoW success occurs.

$$\Pr_p\{K \le k\} = \Pr_p\{KT \le kT\}$$

$$=: \Pr_p\{S \le kT\}$$

$$= \Pr_p\{S \le t\}$$

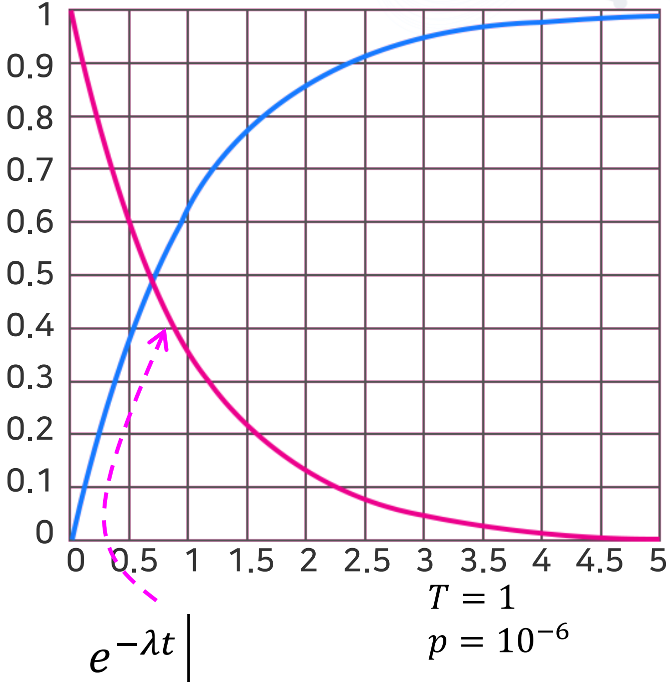
GIST

3 Geometric vs Exponential Distribution

- $\Pr(S > t) = e^{-\lambda t}$ where $\lambda = \frac{p}{T}$

$$P_p\{K > k\} = (1-p)^{\frac{1}{T}kT}$$
$$= (1-p)^{\frac{1}{p}\frac{p}{T}kT}$$
$$= \left\{(1-p)^{\frac{1}{p}}\right\}^{\frac{p}{T}kT}$$
$$= e^{-\frac{p}{T}kT}$$
$$= e^{-\lambda t}\Big|_{t=kT}$$

Geometric distribution
& Exponential distribution



GIST

3 Geometric vs Exponential Distribution

- $\Pr(S > t) = e^{-\lambda t}$ where $\lambda = \frac{p}{T}$
- We now aim to determine lambda.
  - Suppose a mining chip with hash rate $R_{chip}=10^{12}$ [hashes/sec].
  - The time duration per hash is 1 pico $T = 10^{-12}$ [sec/hash].

  - We now treat the time is continuous.

GIST

3 Geometric vs Exponential Distribution

- $\Pr(S > t) = e^{-\lambda t}$ where $\lambda = \frac{p}{T}$
- We now aim to determine lambda.
  - Recall the average number of hashes for a PoW success or a block generation is $E\{K\}=1/p=10^{20}$ [hash/block].
  - Thus, lambda's unit is [block/hash]/[sec/hash] = [block/sec].

GIST

3 Geometric vs Exponential Distribution

- Lambda is the block generation speed.
- Recall

$$T_{block} = \mathrm{E}\{K\} / R_{chip}$$
$$= 10^{20} / 10^{12} \text{ [sec/block]}$$
$$= 10^{8} \text{ [sec/block]}$$
$$= 3.15 \text{ [year/block]}$$
$$= 1/\lambda$$

Thus, lambda is block generation speed!

4 Block Generation Speed

- Network Hash Power vs. Block Generation Speed
  - A Bitcoin network's hash rate is the total mining rate of all online nodes.
  - Suppose the whole network is divided into two pools of computers, say pool A and pool B.
  - The hash rate of pool A is twice that of pool B.
  - What is the block generation speed of A?

4 Block Generation Speed

- Note that

$$\lambda_A + \lambda_B = 1 \;\; [\text{block} / 10 \text{ min}].$$

- Since the hash power of pool A is twice that of pool B, the block generation speed of pool A is twice faster than that of pool B, i.e.,

$$\lambda_A = 2\lambda_B.$$

GIST

4 Block Generation Speed

- Thus, the block generation speed of B is

$$\lambda_{\mathrm{B}} = 1/3 \text{ [block / 10 min].}$$

- Then, that of A is

$$\lambda_{\mathrm{A}} = 2/3 \text{ [block / 10 min].}$$

5 Double Spending Attack Possibility

- 51% Double Spending Attack and its Possibility
  - Recall our subnet example where Bitcoin network is divided into subnet A and subset B.
  - Suppose the hash power of A is greater than that of B.
  - And, the attacker took the control of A.
  - The honest nodes are in B.
  - In this case, the Double Spending attacks launched by A are possible.
  - The probability of DS success can be calculated exactly.

5 Double Spending Attack Possibility

- Immutable File Keeping Technology
  - It is, according to the Bitcoin white paper, an unlikely event to have such an attacker with a sizeable pool of computers working for him in the network of decentralized and independent participants.

6 Data Immutability

- Proof of Work and Data Immutability
  - Proof of work(작업증명 in Korean) is to have a large set of miners find a solution satisfying the PoW with the given difficulty.
  - The first miner which succeeds in solving it obtains the right to produce a certain amount of new coins minted and paid to himself.

6 Data Immutability

- Proof of Work and Data Immutability
  - It is the key mechanism for enforcing integrity of data stored inside the blockchain.
  - Blockchain can be considered as a very large stone everyone can see!
  - Each and every transaction is checked for validity and scribed into the stone.
  - How can it be done with digital file?

6 Data Immutability

- Proof of Work and Data Immutability

Answer is simple!

- Let a large number of computers work together simultaneously.
  Let the first computer which is successful at finding a good answer get rewarded.
- Have a new race begin by having the computers work on a new problem (new block) and reward the new winner.
- The proof of work is an evidence that a large number of computers have worked together.
- If any computer, or a group of computers, aims to change the block content, then the same amount of work needs to be redone.

GIST

6 Data Immutability

- Immutable File Keeping Technology
  - The problem can almost never be solved alone, but it is designed in such a way that it can be solved within a desired time span when many computers come and compete to find a solution.
  - It also has a means to measure the total amount of work done in probabilistic sense.
  - If the difficulty level of the problem is increased, the number of computers in competition has to increase as well.

GIST

6 Data Immutability

- Immutable File Keeping Technology
  - This is used to protect the integrity of the data stored in the Blockchain. Because of the Al-Im-To-Po result, a small group cannot fool the majority.
  - PoW is to find the nonce or the block header (BH) which matches with the block content and have this nonce written into the block header.
  - Why those transactions once scribed inside blockchain are not alterable?
  - The block contents are locked with the nonce.

6 Data Immutability

- Immutable File Keeping Technology
  - When the block content is changed somehow, the content no longer matches with the nonce found.
  - Such blocks are easily detectable and thus a chain containing such block are also easily detectable and thrown away.
  - Thus, anybody who aims to launch an attack of changing the content, the person needs to redo the PoW again and find a new nonce reflecting the changed block content.

GIST

6 Data Immutability

- Immutable File Keeping Technology

  But it is not the end

  - The hash value of the previous block, F(block, nonce) in (PoW), is written inside the header of the next block.
  - Blocks are connected in a serial fashion with these hash values.
  - Thus, if an attacker aims to change the content of a block, he has to re-do all the block headers subsequent to the altered one.
  - This requires the attacker to redo all the PoWs for the subsequent blocks.

6 Data Immutability

- Immutable File Keeping Technology
  - Recalling that it is very difficult to find the nonce for a single block alone, it becomes almost impossible for a single computer to find all the nonces again for the subsequent series of blocks.
  - In addition, the honest nodes are continuously making new blocks.
  - Thus, if an attack wants to be successful, he needs to recruit computing resource with a hash power greater than that of honest nodes.