[이론] 블록체인 핵심기술

→ Consensus (합의) 알고리즘

→ 블록체인 P2P 네트워크

블록체인 (Blockchain)

08. Consensus 알고리즘 및 P2P 네트워크

소프트웨어 꼰대 강의

노기섭 교수

(kafa46@cju.ac.kr)

Consensus (합의) 알고리즘

비잔틴 내결함성 (Byzantine Fault Tolerance)

비잔틴 장군 문제

- 장군들은 하나의 성을 점령하고자 한다 ➔ 과반수 이상이 동시에 행동해야 작전 성공
- 장군들은 다수결 합의에 따라 동시에 공격 또는 후퇴 (단, 1대1 통신만 가능)
- 총사령관 1명, 그리고 장군 여러 명으로 구성
- 장군들 중 누구든지 반역자가 될 수 있다(총사령관 포함), 누가 배신자인지는 모름

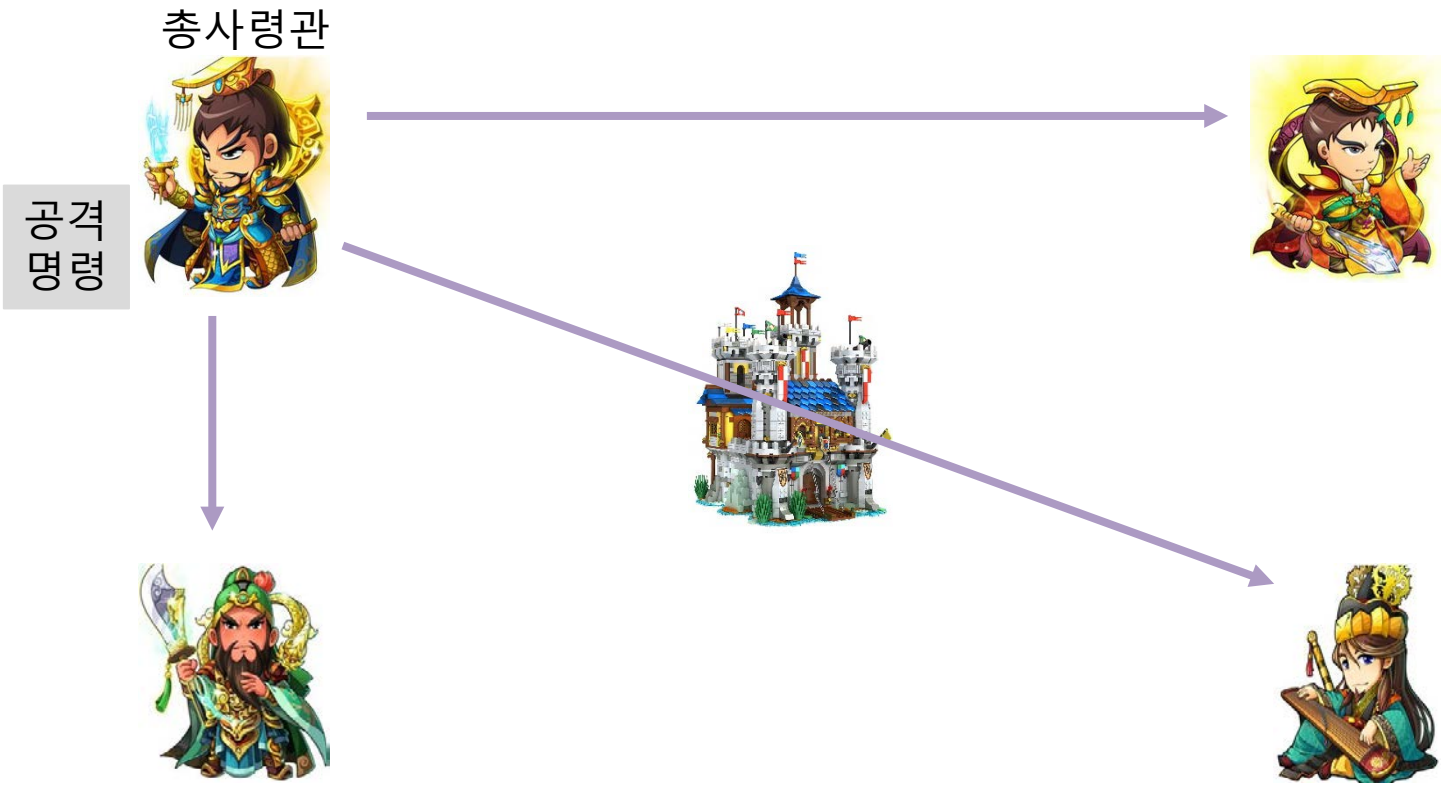
50% 이상의 장군이 총사령관의
명령에 따라야 작전 성공!

- 배신자가 몇 명 이하일 경우 작전 성공할까?
- 모든(n명) 장군들이 합의에 따라 공격/후퇴를 결정하는 알고리즘(프로토콜)은?

비잔틴 내결함성 - 배신자 없는 경우 1/2

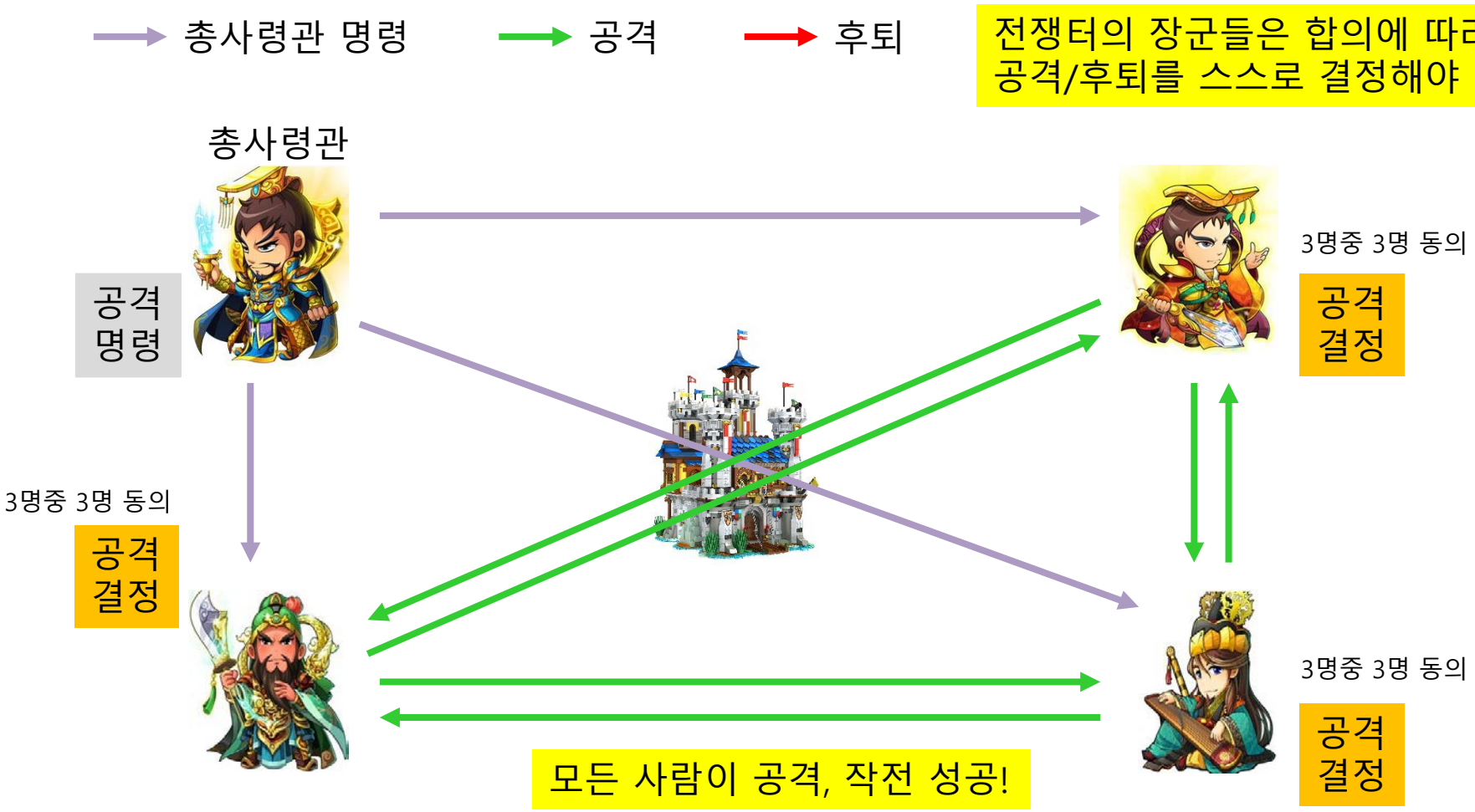
■ **합의 프로토콜: 장군들의 메시지 공유 및 결정 (정상 상황, 배신자 없음)**

→ 총사령관 명령 → 공격 → 후퇴 총사령관이 공격 명령을 내림



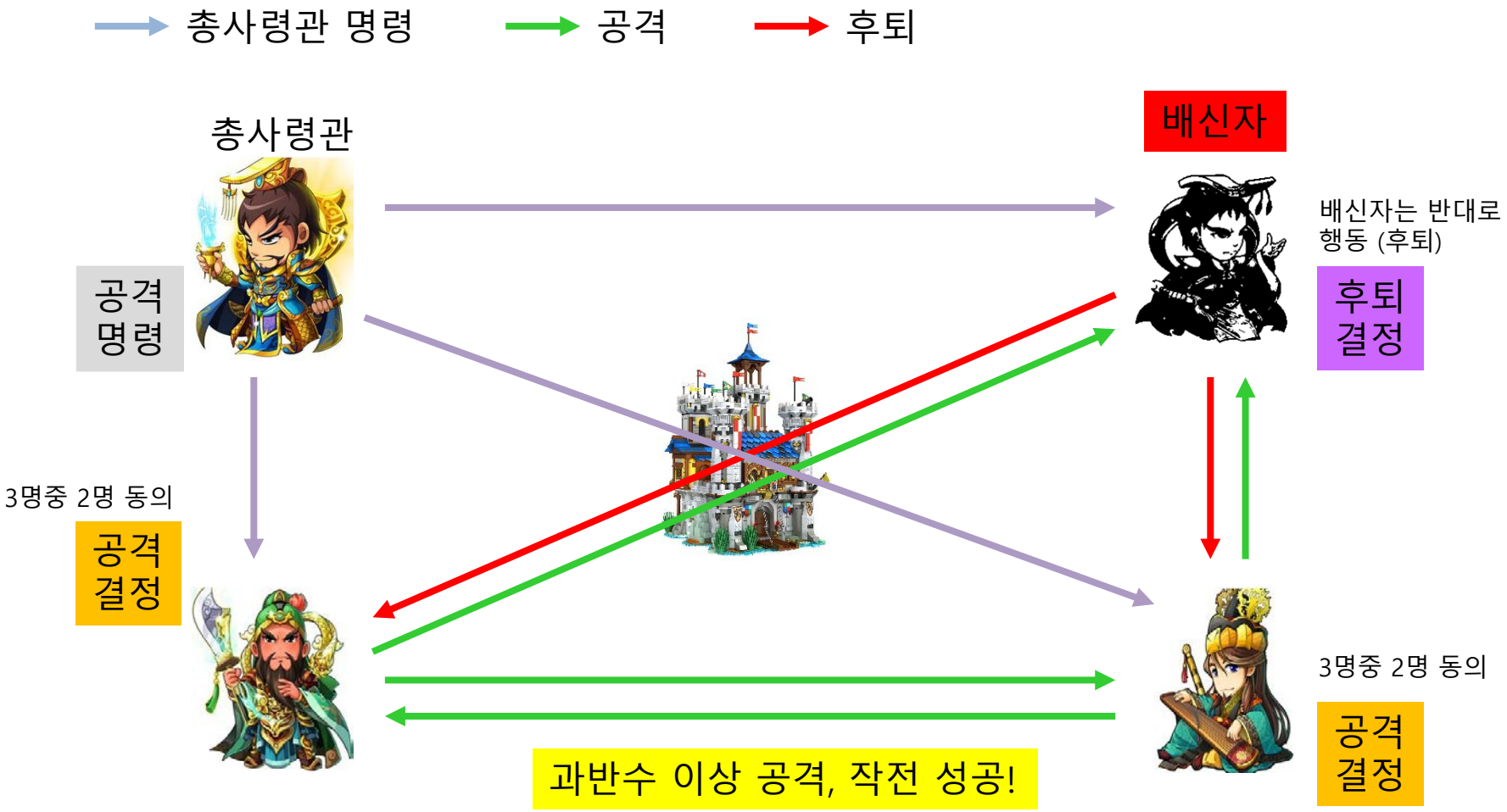
비잔틴 내결함성 - 배신자 없는 경우 2/2

■ **합의 프로토콜: 장군들의 메시지 공유 및 결정 (정상 상황, 배신자 없음)**



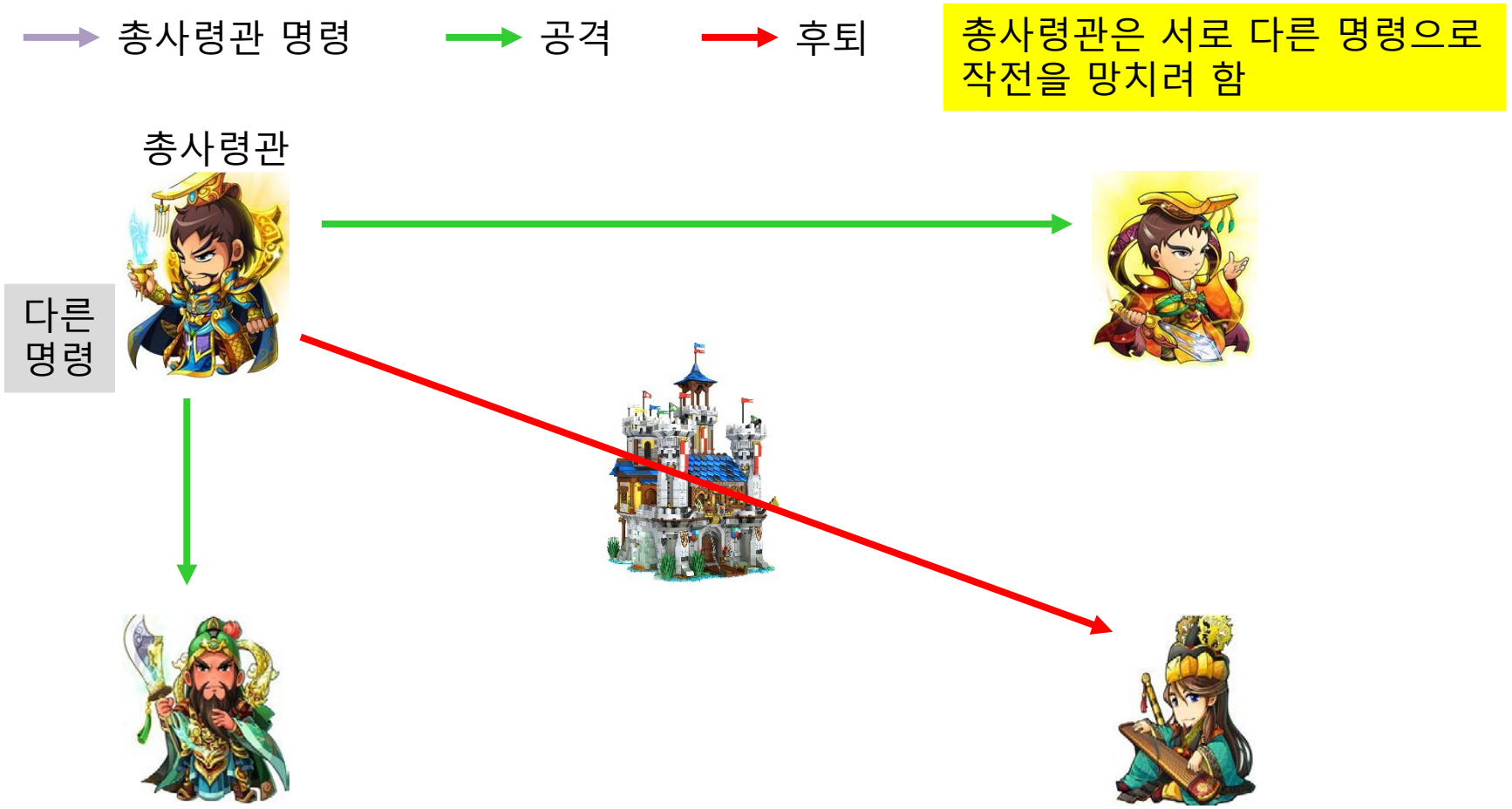
비잔틴 내결함성 - 배신자가 1명인 경우

■ **합의 프로토콜: 총 사령관이 공격 명령을 내린 경우 (배신자 1명)**



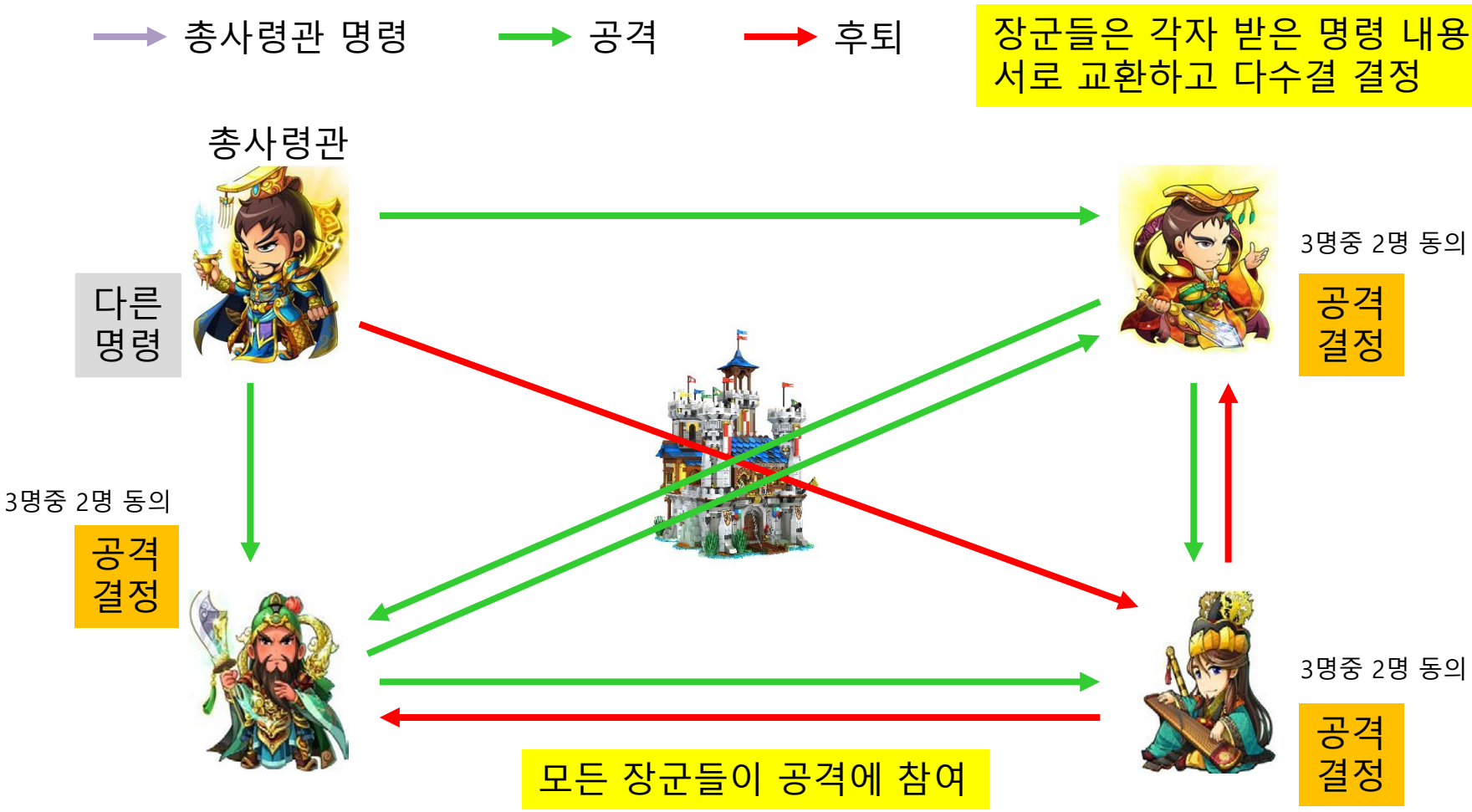
비잔틴 내결함성 - 총사령관이 배신자인 경우

■ **합의 프로토콜: 총 사령관이 배신자인 경우 (서로 다른 명령을 내림)**



비잔틴 내결함성 (Byzantine Fault Tolerance)

■ **합의 프로토콜: 총 사령관이 배신자인 경우 (서로 다른 명령을 내림)**



비잔틴 내성결함 - Tolerance 한계?

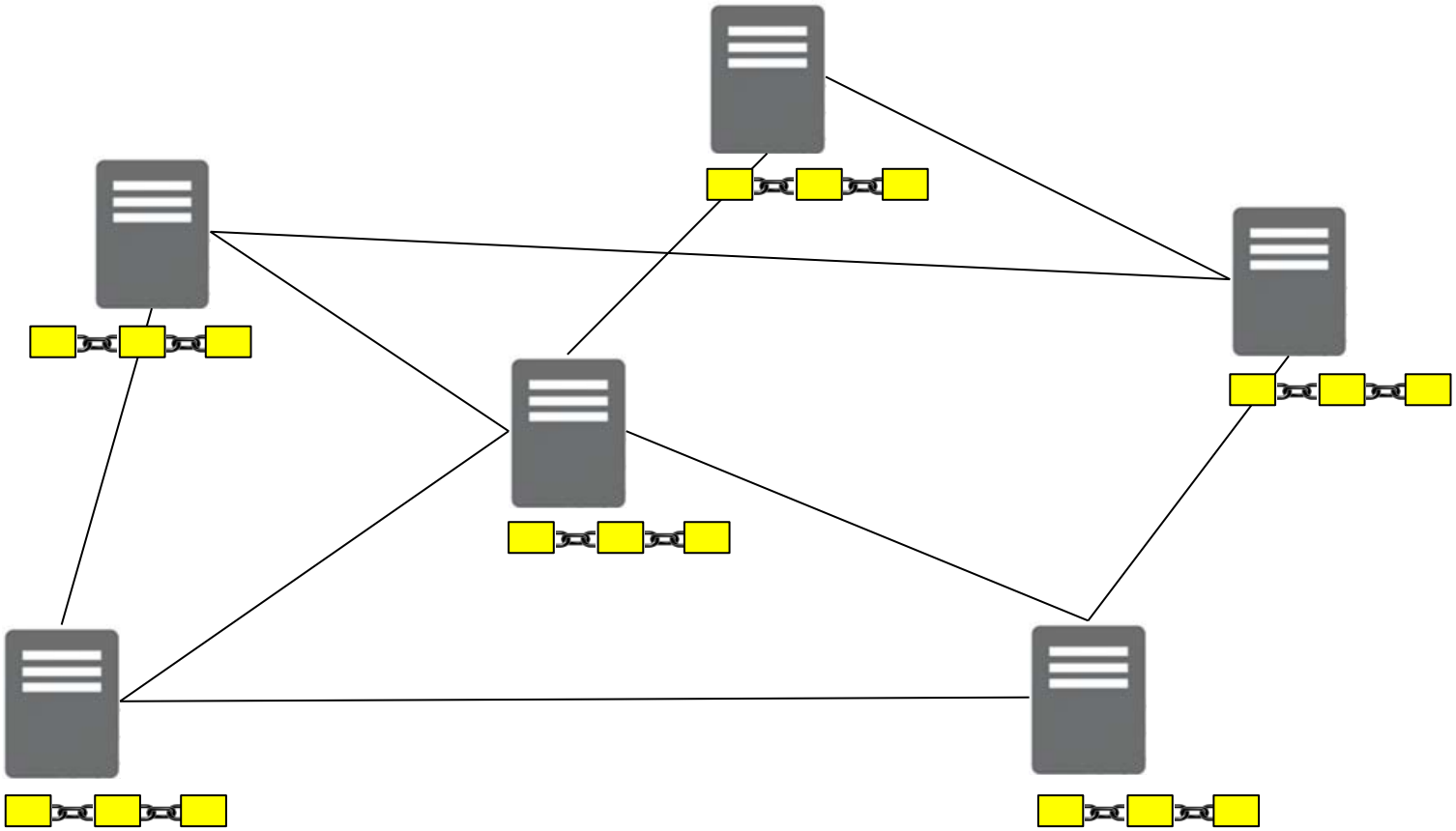
- **공격에 대한 내성(Tolerance) 한계는 얼마일까?**
  - 탈중앙화 시스템에서의 공격에 대한 안전 수준과 같은 문제
  - 이중화 시스템에서 안전 유지를 위한 다중회로를 몇 개 만들어야 하나와 같은 문제
  - 공격자가 몇 명까지 견디는가? 서버가 몇 % 까지 좀비가 되었을 때까지 견디는가?

- **Answer: n 명의 참가자가 있을 경우, 배신자는 최대 $1/3 \approx 33.3\%$ 이하일 경우 정상 작동**
  - L. Lamport, R. Shostak, and M. Pease, "The Byzantine Generals Problem," 1982.
    - 논문: https://lamport.azurewebsites.net/pubs/byz.pdf
    - 버클리대학 강의자료: https://people.eecs.berkeley.edu/~kubitron/cs262/lectures/lec19-Byzantine.pdf
    - 블로그: https://medium.com/loom-network/understanding-blockchain-fundamentals-part-1-byzantine-fault-tolerance-245f46fe8419

- **블록체인 역시 탈중앙화 시스템...**
  - 어떻게 블록체인에 대한 신뢰도 합의를 할 것인지에 대한 프로토콜 필요
  - 블록체인은 비잔틴 프로토콜과 다른 프로토콜을 사용

블록체인 합의 알고리즘 내성(Tolerance)

- 모든 블록체인 서버가 동일한 블록체인을 공유

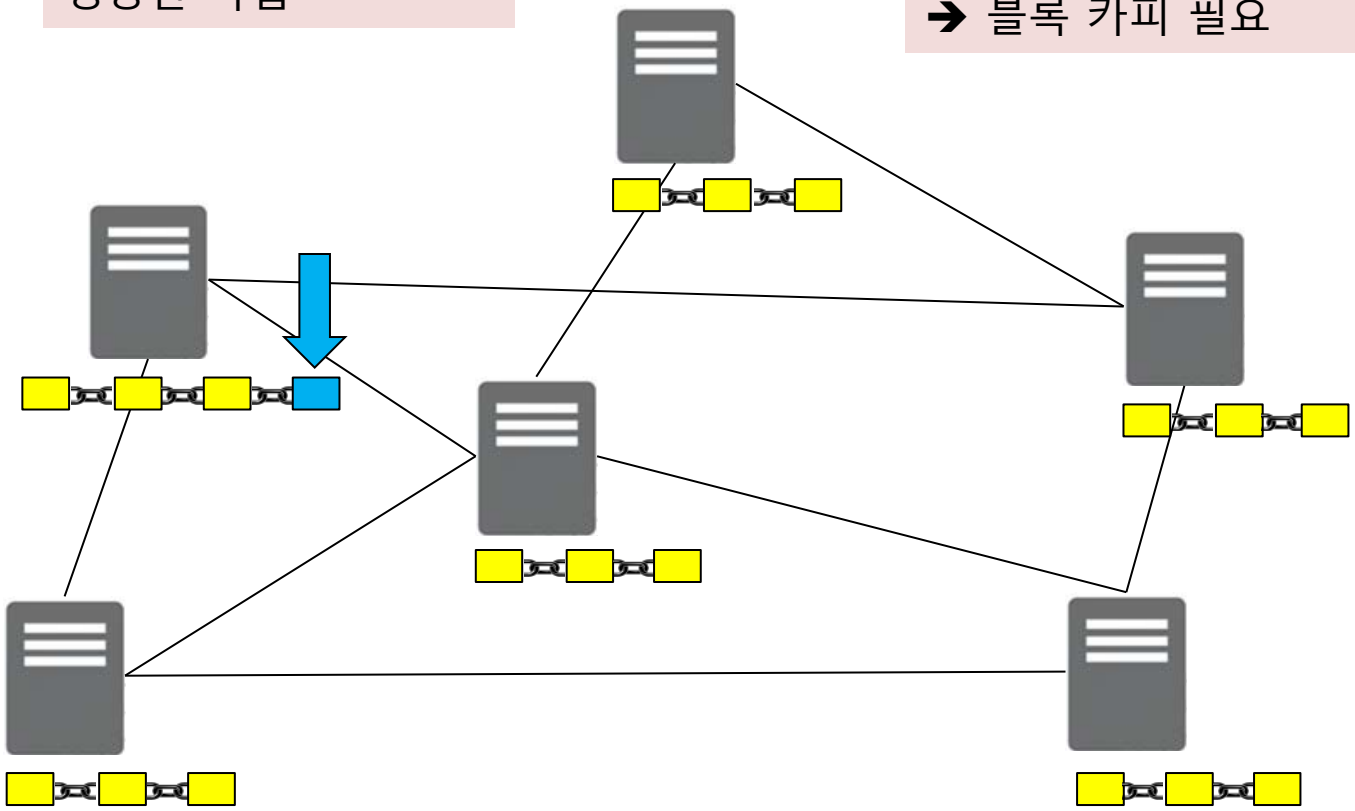


블록체인 합의 알고리즘 내성(Tolerance)

■ 1개 서버가 블록을 추가한 경우

각 노드 입장에서는
블록체인 규칙에 따른
정당한 작업

다른 노드들?
정당한 블록이 생성된 것으로 간주
모든 노드가 동일한 블록을 보유
➔ 블록 카피 필요



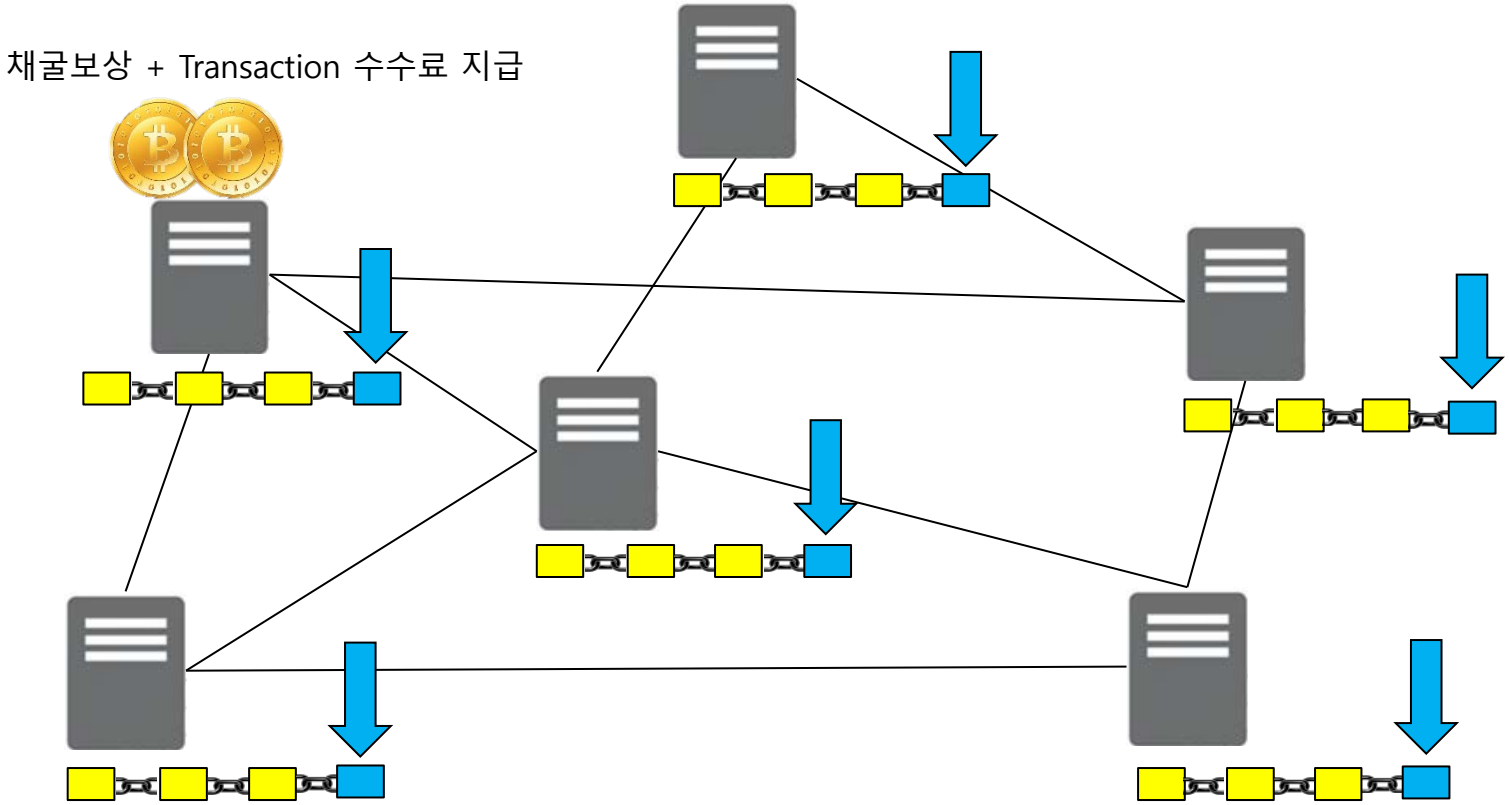
블록체인 합의 알고리즘 내성(Tolerance)

Proof of Work 방식을 사용하자!

- 마이닝 과정에서 정당한 Nonce 값을 찾았다면

  → 정당한 작업을 수행하여 블록을 생성한 것으로 인정!

채굴자에게 보수 지급 후
모든 노드의 체인을 업데이트!

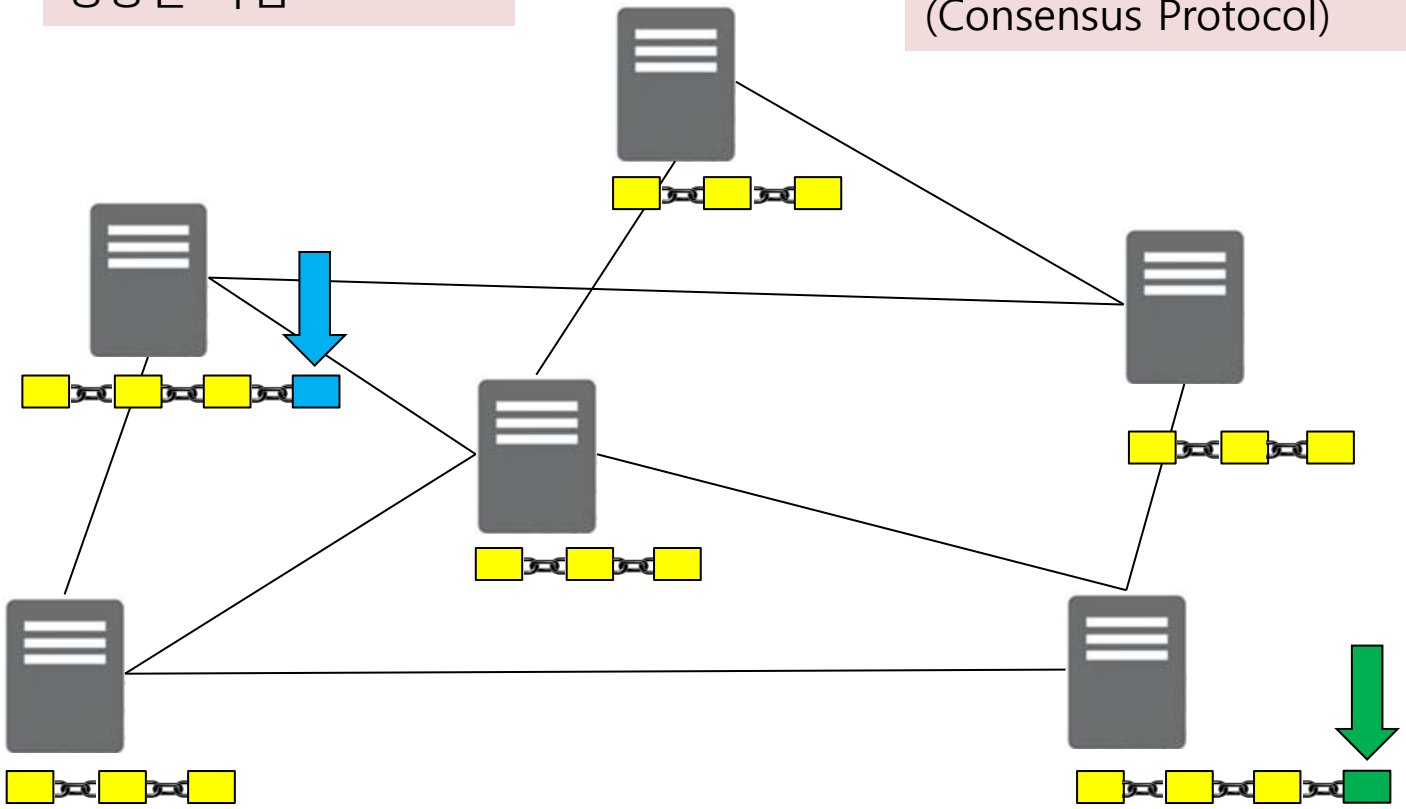


블록체인 합의 알고리즘 내성(Tolerance)

■ 2개 서버가 동시에 블록을 추가(동시에 채굴 성공)한 경우

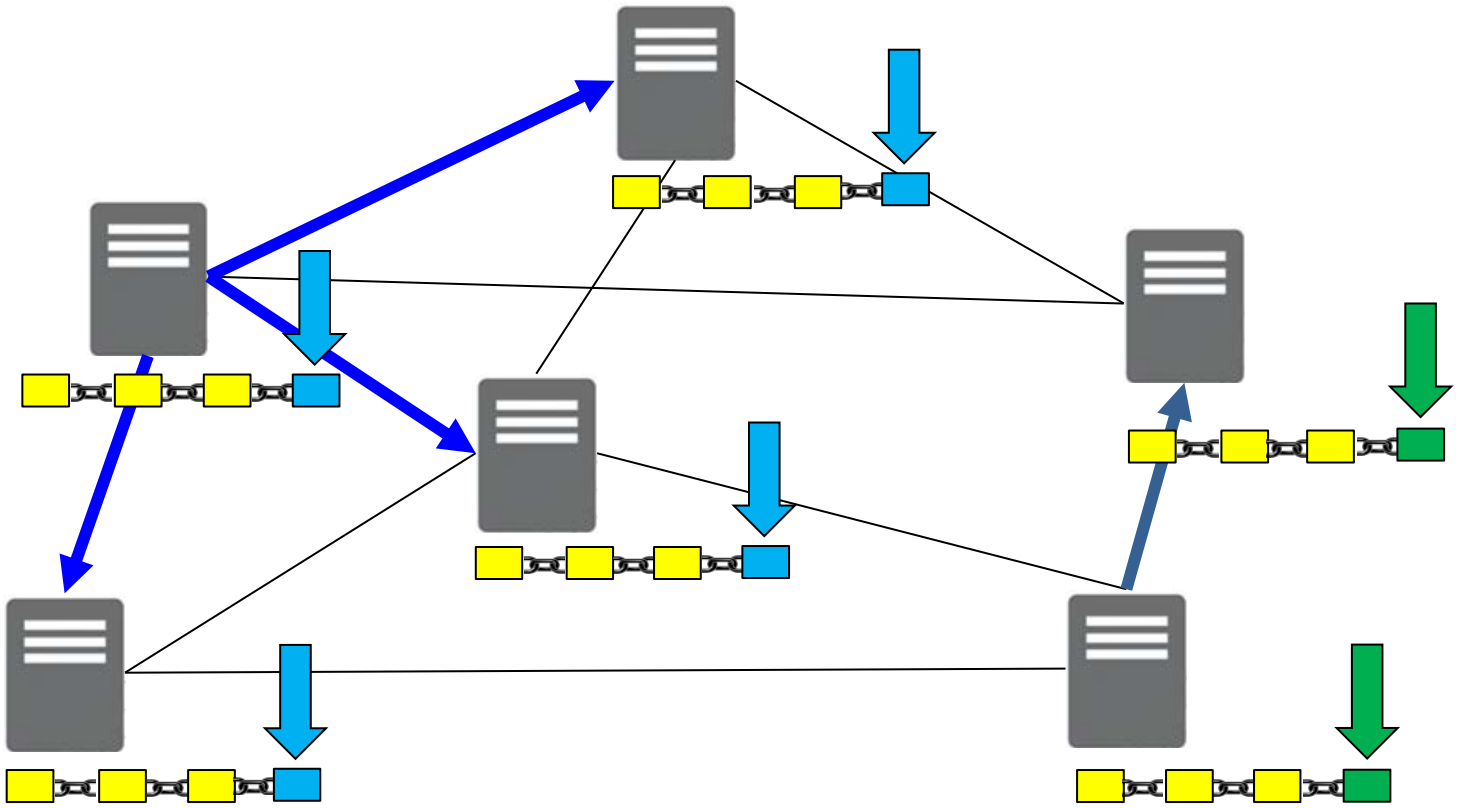
각 노드 입장에서는
블록체인 규칙에 따른
정당한 작업

하지만, 나머지 노드들은
둘 중 하나를 선택해야 하는 상황
비잔틴 장군 문제와 유사
(Consensus Protocol)



블록체인 합의 알고리즘 내성(Tolerance)

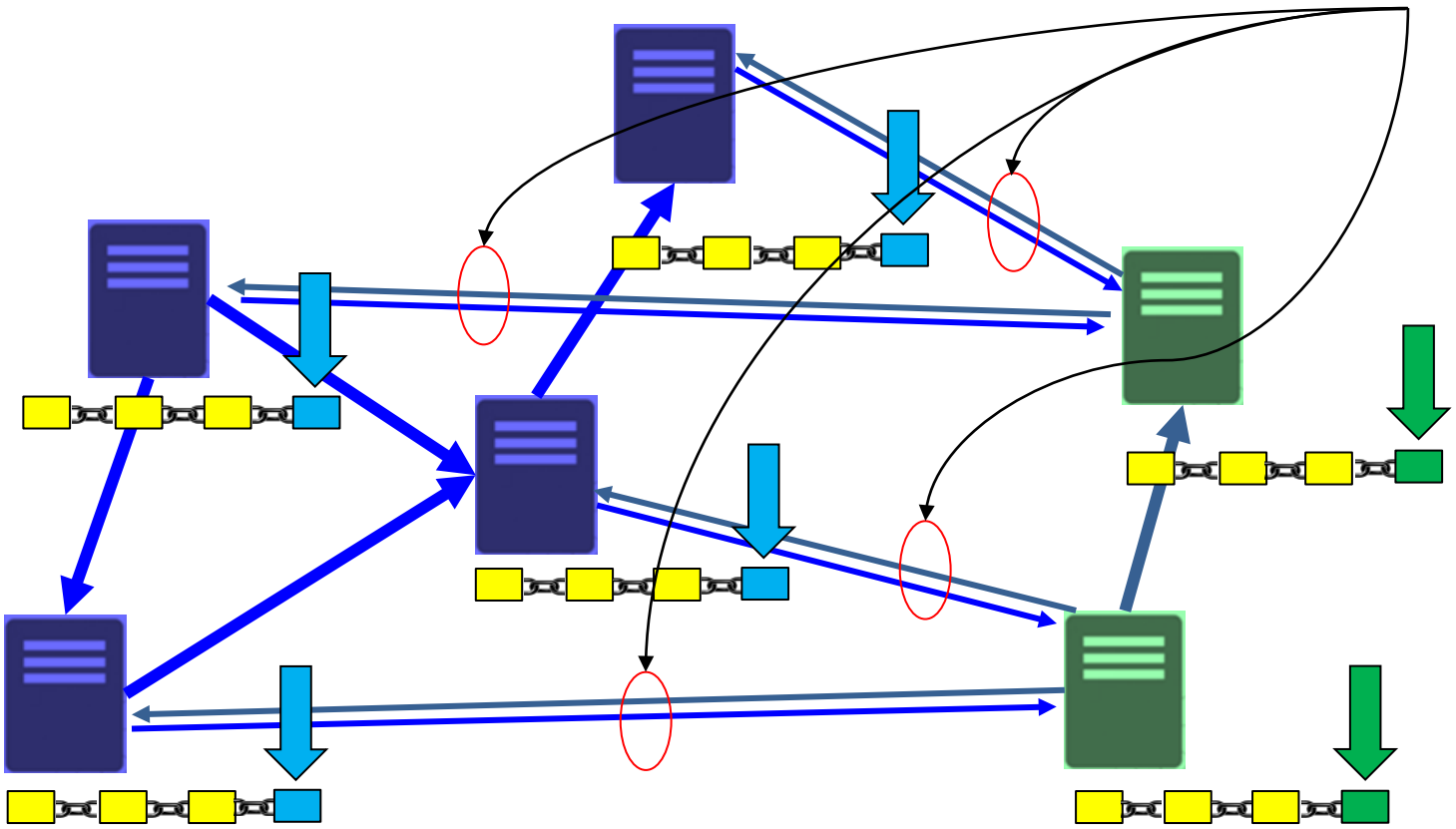
- **2개 서버가 동시에 블록을 추가(동시에 채굴 성공)한 경우**
  - 네트워크 연결 상태에 따라 업데이트 노드 수가 다름
  - 네트워크가 빠르면 보다 많은 노드가 수용, 느리면 적은 수의 노드가 수용



블록체인 합의 알고리즘 내성(Tolerance)

- **네트워크 상태에 따른 블록 업데이트 이후**
  - 네트워크 연결 상태에 따라 업데이트 노드 수가 다름
  - 서로 연결된 노드들 사이에 다른 의견 발생 (비잔틴 장군 문제와 동일 상황)

더 긴 체인이 생길 때까지 기다린다!

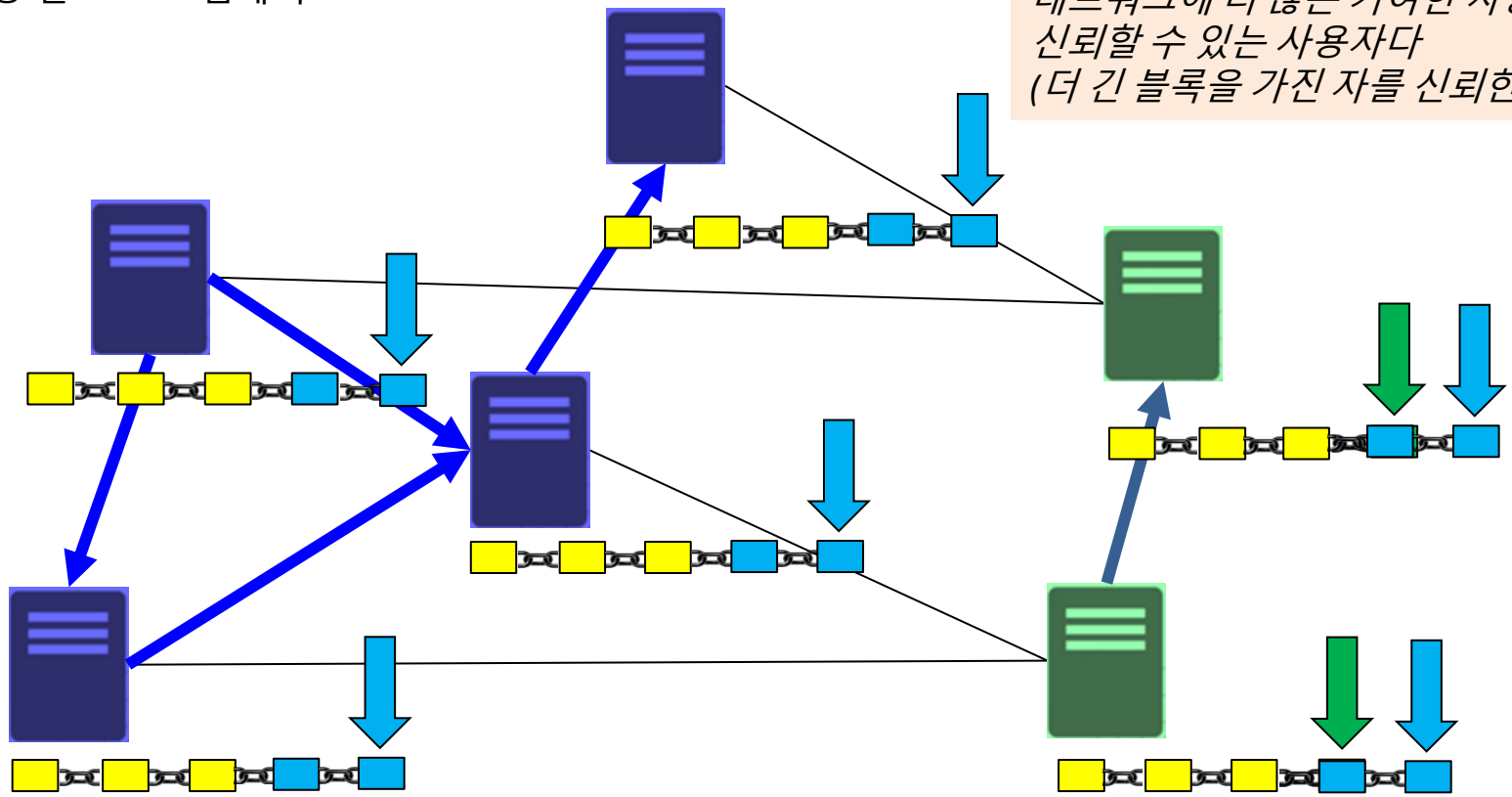


블록체인 합의 알고리즘 내성(Tolerance)

네트워크 상태에 따른 블록 업데이트 이후

- 더 긴 체인이 발생 ➔ 가장 긴 체인을 중심으로 모든 노드의 체인을 업데이트
- 기존의 동일 길이 체인에 추가된 정보는 버리고(고아 블록, orphan node),
- 가장 긴 노드로 업데이트

네트워크에 더 많은 기여한 사용자는 신뢰할 수 있는 사용자다
(더 긴 블록을 가진 자를 신뢰한다)



51% 공격이란?

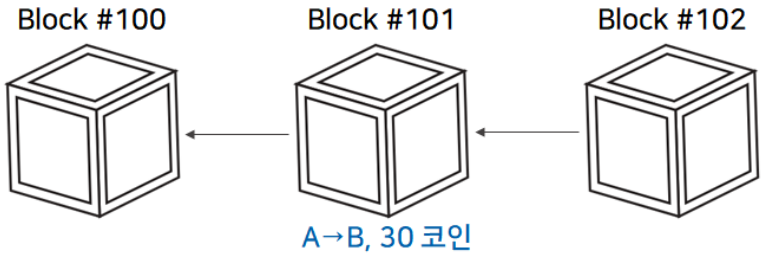
- **블록체인 네트워크를 공격하려면?**
  - 공격자가 50% 이상의 해싱 파워(hash power)를 가진다면 가능한 일
    - 비잔틴 프로토콜의 경우 33% 이상을 제어한다면 성공
    - 블록체인은 비잔틴 프로토콜 보다 더 많은 노력 필요
  - 그러나 전 세계에 흩어져 있는 블록체인 컴퓨터(서버)는 수만~수십만 개 (추측)
    - 실질적으로 해커가 51% 이상의 컴퓨팅 파워를 제어하는 것은 어려운 문제
    - 비트 코인 인기 상승 → 더 많은 채굴자 → 51% 확보는 더욱 어려워 짐

- **이러한 블록체인의 취약성에 대한 공격 개념을**

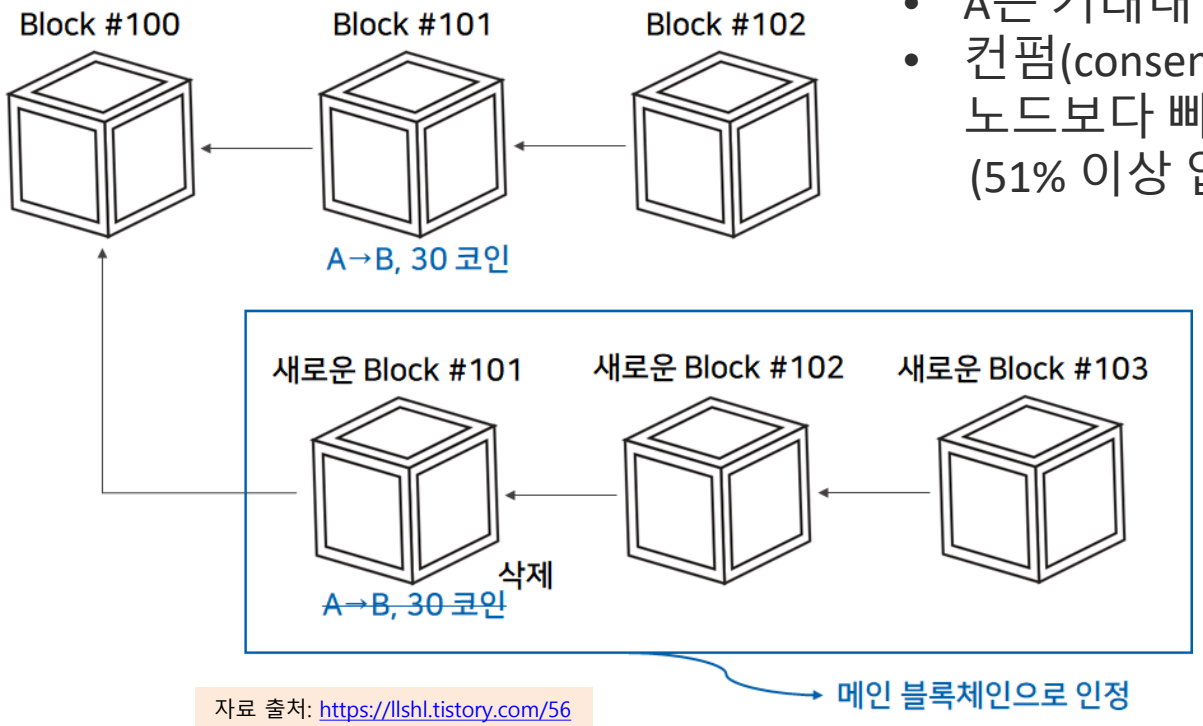
  "51% 공격"이라고 합니다.

해시 파워:
채굴 시 Nonce를 찾는데 필요한 컴퓨팅 능력(계산 속도)

51% 공격 (예시)



A가 B에게 30코인을 줬다는
거래 내역이 101번 블록에 기록
(여기까지는 정상 ~)



- A는 거래내역을 삭제한 새로운 블록 생성
- 컨펌(consensus)을 위한 블록을 다른 어떤 노드보다 빠르게 생성
  (51% 이상 압도적인 컴퓨팅 파워를 사용)

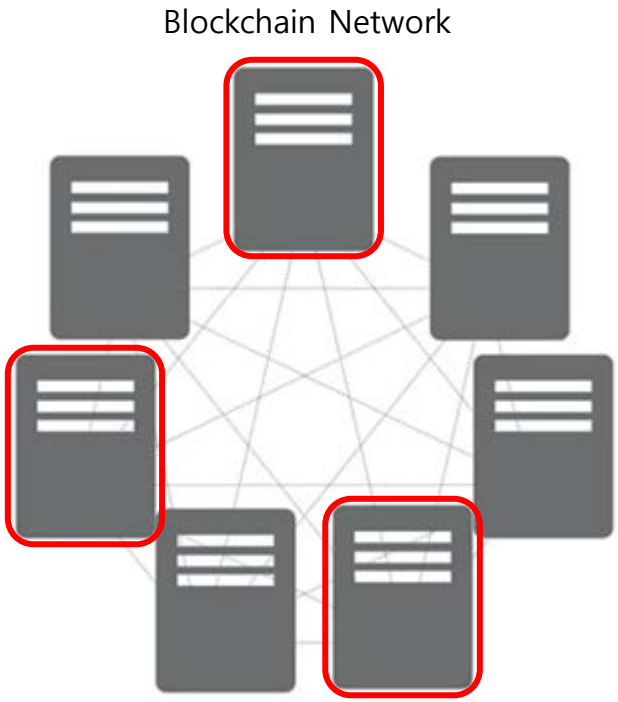
A가 조작한 블록이 속한
체인을 메인 체인으로 인정

자료 출처: https://llshl.tistory.com/56

P2P 네트워크

How to find neighbors?

이미지 출처: https://post.naver.com/viewer/postView.nhn?volumeNo=28855823&memberNo=12161421



친구를 알아야 블록체인 정보를 보낼 텐데....

Bitcoin wallet은 어떻게 노드를 찾을까?

When a new client starts up, it will attempt to connect to one or more of these seed nodes to bootstrap its connection to the Bitcoin network.

Once the client has established a connection to a seed node, it can exchange information about other nodes on the network and attempt to connect to them as well.

이미지 출처:
https://www.linkedin.com/pulse/how-does-each-bitcoin-wallet-find-other-nodes-connect-esmaeilzadeh

How does each bitcoin wallet find other nodes to connect?

Mohammadreza Esmaeilzadeh
System/Data Engineer, Distributed real-time stream processing with Apache Flink, Kafka, Scala ZIO stream, ...
발행일: 2023년 3월 30일

+ 팔로우

Bitcoin is a peer-to-peer network and dosen't have a centralized server. Maybe you also wonder how each client (wallet) chooses a node to connect and send and receive transactions.

each client has a list of hardcoded seed nodes:

The hardcoded seed nodes of Bitcoin are a set of IP addresses and domain names that are built into the Bitcoin Core client software. These seed nodes serve as a starting point for new clients to connect to the Bitcoin network since they are known to be reliable and always available.

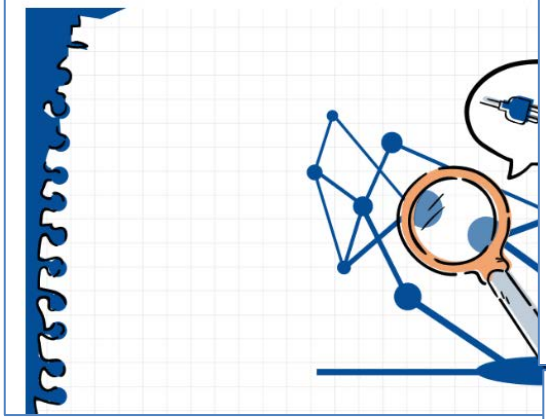
Bitcoin wallet은 어떻게 노드를 찾을까?

A seed node is a special node that allows the incorporation of new nodes to the network and maintains the strength of the network at all times, by allowing them to synchronize and obtain a copy of the data from the blockchain, replicating it and adding resistance and security to it.

What is a seed node?

9 June, 2020

▸ Advanced ⏲ 6 min reading



이미지 출처:
https://academy.bit2me.com/en/what-is-a-seed-node/

How does a seed node work in Bitcoin?

Bitcoin has a series of seed nodes that are used to locate active nodes. From these nodes, another new node that wants to enter the network, can connect. That is, the seed nodes are used only to locate or find complete nodes that are running the Bitcoin client. Sort of like an address book that tells other nodes who to go to in order to be part of the network. Surprising isn't it? This means that Bitcoin organizes its network in such a way that even new nodes find it easy to start being part of the Bitcoin network.

Thus when a new node wants to join the Bitcoin network, it must briefly connect to a seed node. This will indicate or provide you with a list of the IP addresses of the nodes that are active within the Bitcoin network, and through which you can connect to the system.

DNS seed servers

In Bitcoin when the nodes connect for the first time, they do not know the IP addresses of any of the full nodes that are active on the network. So they require connecting to a seed node to obtain these IP addresses. Only then can they connect to at least one of the active complete nodes regardless of where it is geographically located.

P2P 네트워크 구축

P2P 네트워크 목적

- 사용자 PC는 (node 또는 peer)는 어느 블록체인 노드로 접속해야 할지 모른다.
- 최근 접속한 노드의 정보를 가지고 있어야 함.
- 적용: 접속할 노드 정보가 필요한 경우
  - wallet server: 트랜잭션 정보를 전송할 노드를 알아야 함
  - blockchain server
    - 자신이 새로운 블록을 만들었을 경우 다른 노드로 전달
    - 다른 노드로부터 블록 정보 수신 → 가장 긴 블록을 찾아서 업데이트

구현 방법

- 기존 패키지를 활용하는 방법
  - https://github.com/macsnoeren/python-p2p-network --> 참고하여 새롭게 작성
- 직접 구현하는 방법
  - 우리가 적용할 방법

P2P 네트워크 구현 전략

■ 블록체인 네트워크를 유지할 최소 1개

- P2P "**Seed Node**" 지정
  - P2P 노드의 PORT는 22901로 단일화
- 코인 네트워크 유지를 위해 항상 작동
- 모든 노드는 P2P 서버를 갖도록 구현
  - P2P 노드는 이웃 노드의 IP, PORT, Timestamp 정보를 DB에 저장

■ 모든 블록체인 노드에서 P2P 작동

- 기본적으로 seed node 정보를 보유(seed node의 IP, PORT)
- 채굴(mining)에 성공하면 내가 보유한 모든 이웃 노드 정보를 반영하고 업데이트 요청(seed node 포함)
- 모든 이웃 노드에게 합의 알고리즘을 수행하도록 요청
  - blockchain_node/mining/mining.py 업그레이드

Public vs. Private IP Address

공인 IP (Public IP) vs. 사설 IP (Private IP) 비교

공인 IP (Public IP) vs. 사설 IP (Private IP) 차이점

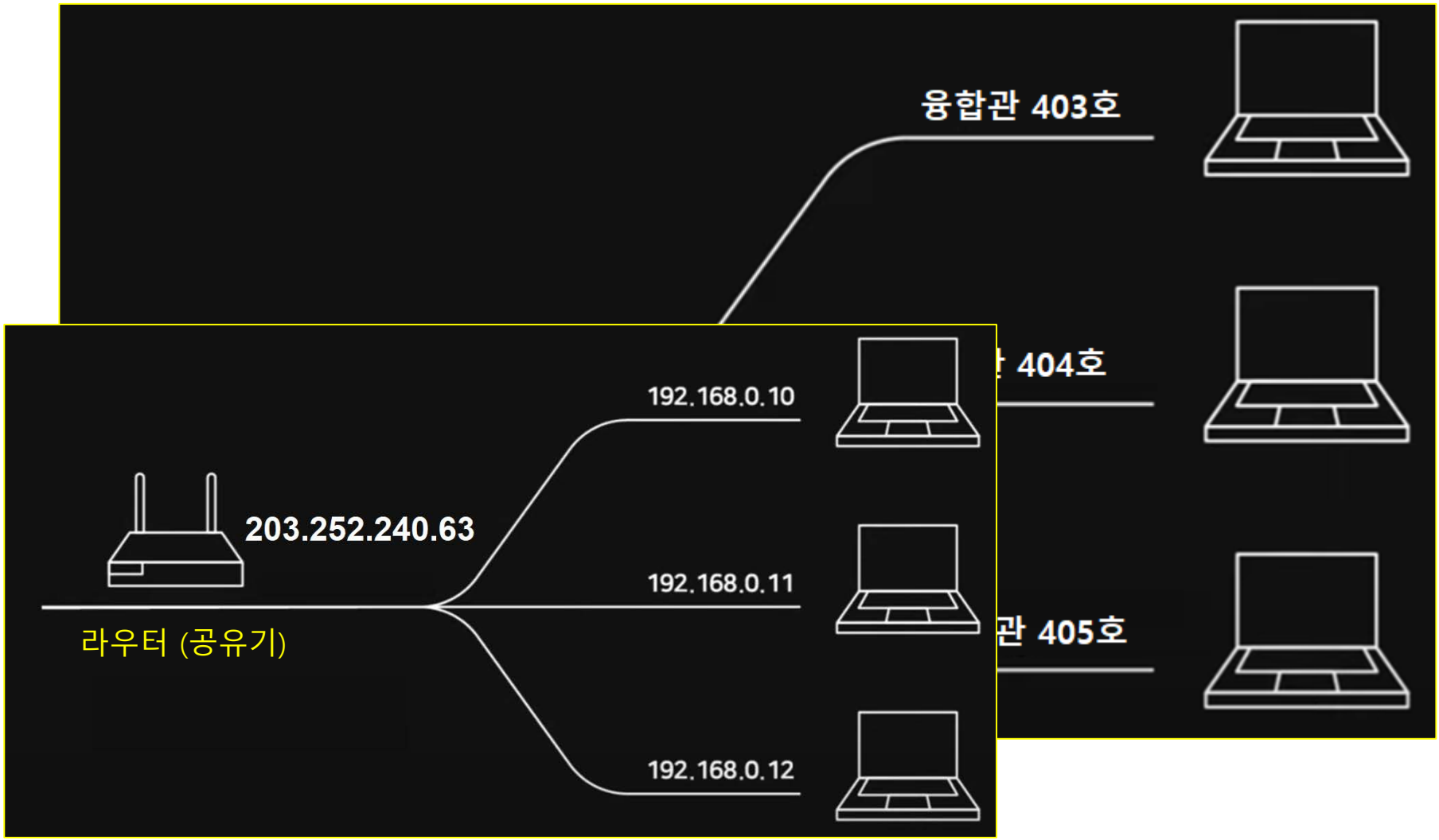
- 공인 IP (Public IP): 지구상에서 절대 유일한 주소
  - 예) 충청북도 청주시 청원구 대성로 298 ← 지구상에서 오직 하나인 주소
  - 예) 203.252.240.63 ← 지구상에서 오직 하나밖에 없는 컴퓨터 논리 주소
- 사설 IP (Private IP): Public IP 내부에서만 유일한 주소
  - 예) 융합관 405호
  - 예) 192.168.0.12

공인 IP (Public IP) vs. 사설 IP (Private IP) 결합

- 충청북도 청주시 청원구 대성로 298 융합관 405호

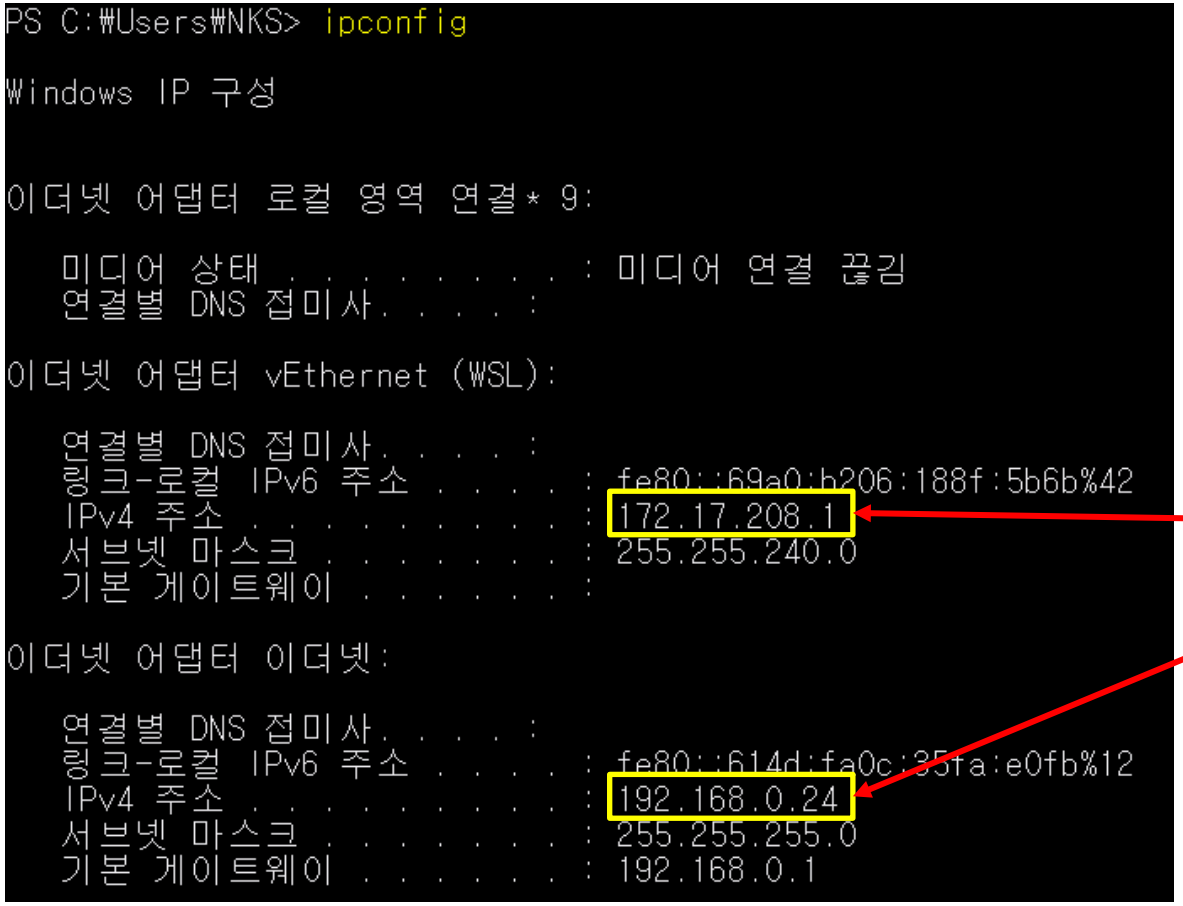
  공인(Public) IP 사설(Private) IP

공인 IP (Public IP) vs. 사설 IP (Private IP) 구성



컴퓨터의 IP 주소 확인하는 방법

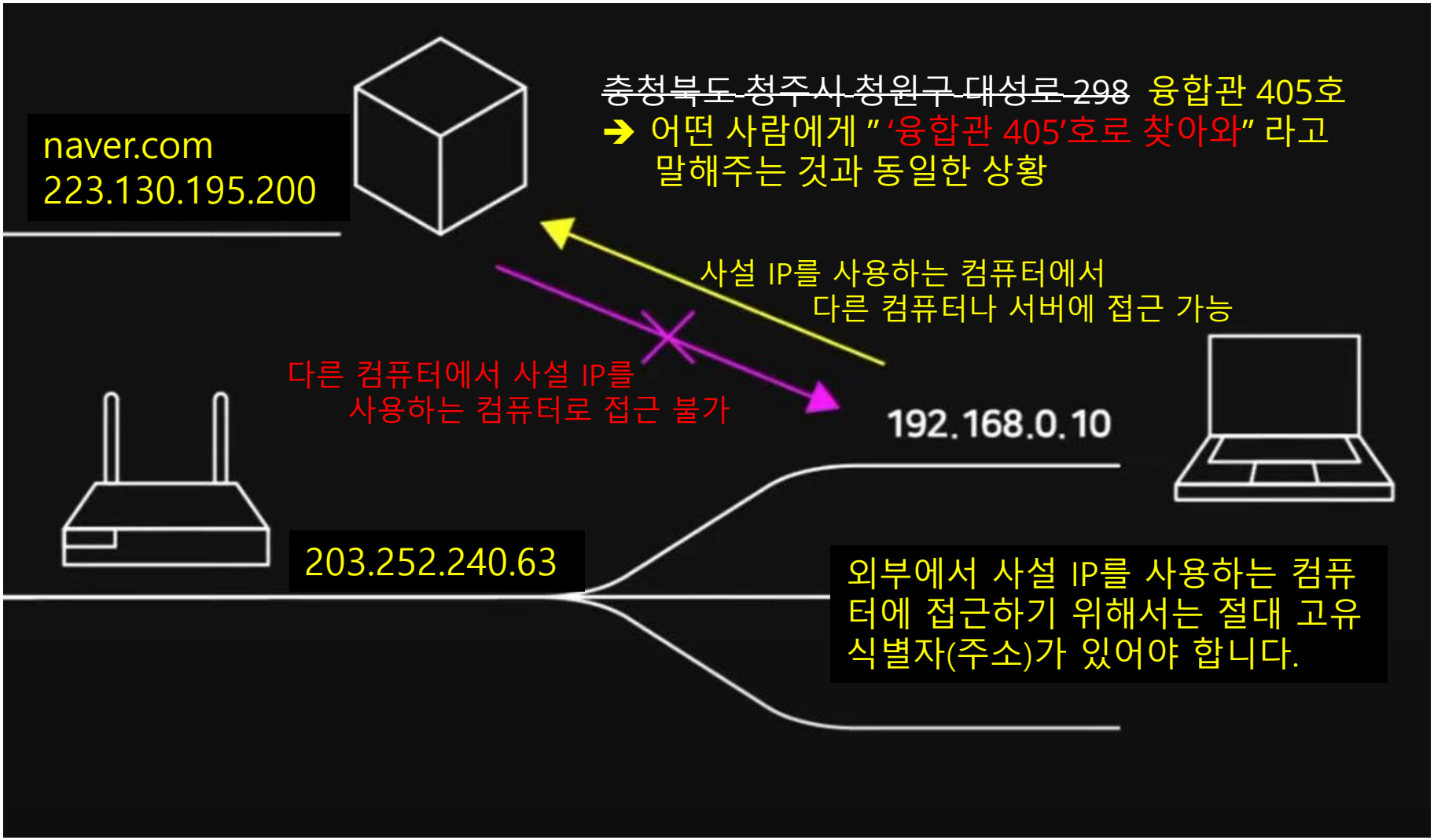
- **윈도우 OS는 ipconfig, 리눅스 운영체계는 ifconfig 명령어 활용**



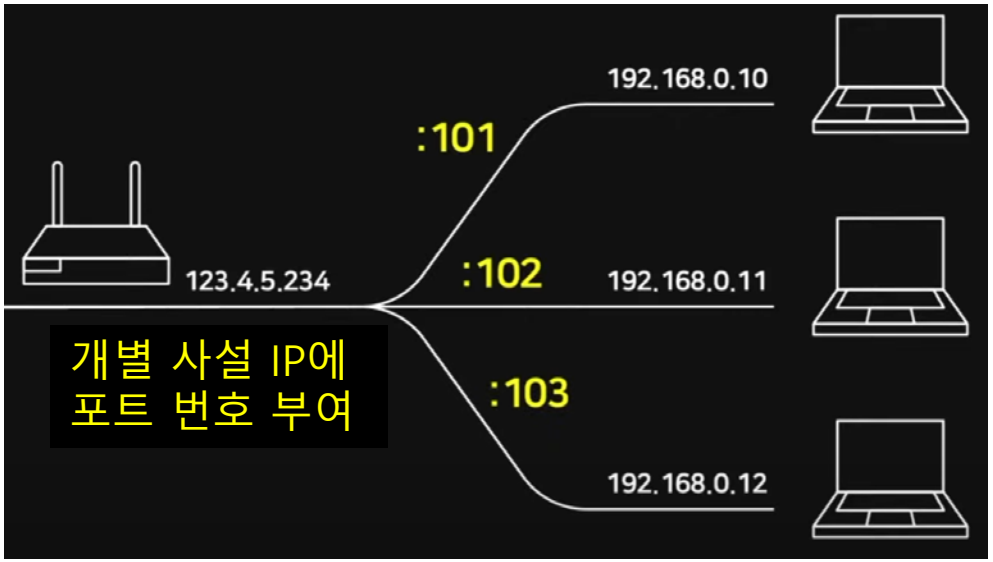
라우터(공유기)에 할당된 공인 IP

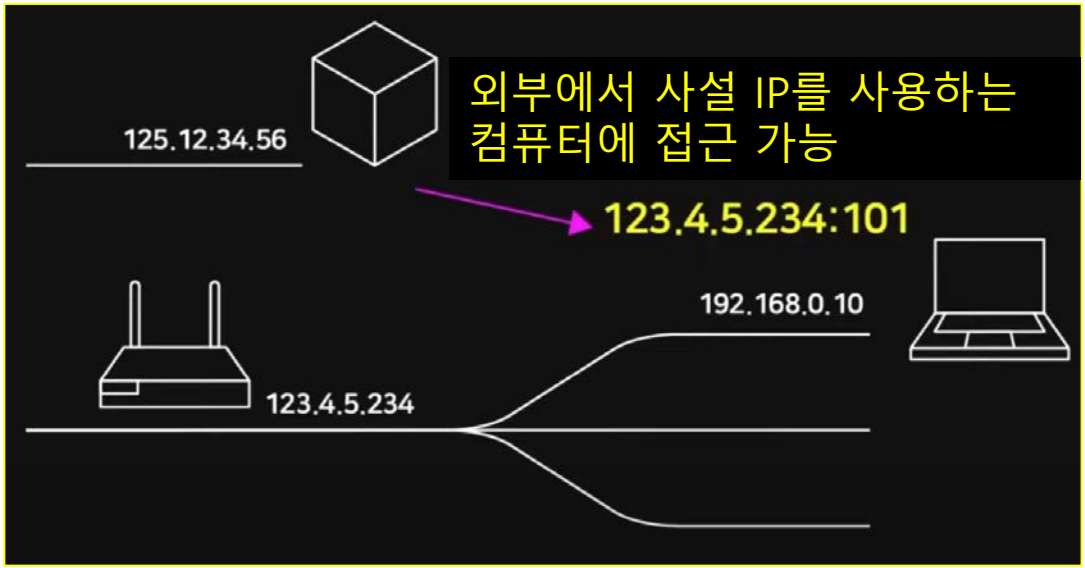
라우터(공유기)가 할당한 사설 IP

공인 IP (Public IP) vs. 사설 IP (Private IP) 통신



포트 포워딩 (Port Forwarding)

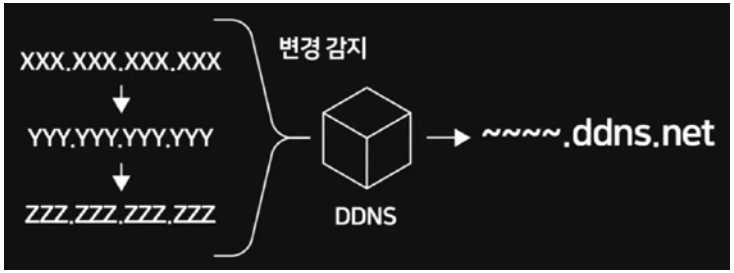


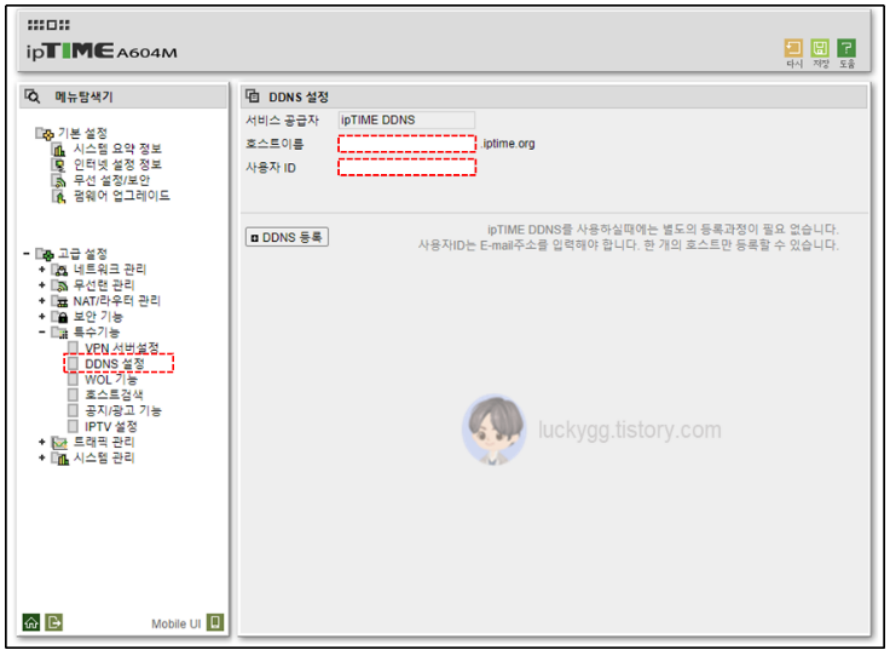


Dynamic DNS

동적 DNS (DDNS)

- 유동(Dynamic) IP의 경우 주기적으로 IP 주소가 변경
- 가정이나 개인이 서버를 운영할 경우 문제 발생
- IP 변경이 있을 경우 자동 감지하여 DNS에 연결해 주는 서비스





고정 IP (Static IP) vs. 유동 IP (Dynamic IP)

고정/유동 IP

- 한정된 IP 자원을 효율적으로 사용
- 전세계 43억 개 IP 중 한국에 배정된 개수는 대략 1억 개
  - 1억 개 IP를 국내 필요한 사람들이 나눠 쓰는 구조
  - 이렇게 IP를 분배하고 관리하는 기관을 ISP 라고 (대표적으로 KT, SKT, LGU+ 등) 부릅니다.

고정(Static) IP

- ISP가 서버와 같이 IP 번호가 바뀌면 곤란한 컴퓨터에게 할당
- 매우 비쌉니다.

유동(Dynamic) IP

- 일반 가정에 있는 컴퓨터, 휴대용 디바이스, IoT 기기 ➔ IP 주소가 바뀌어도 문제되지 않는 컴퓨터
- 주기적으로 사용하지 않는 IP를 수거하고, 필요한 곳에 재할당
- 놀고 있는 컴퓨터의 소중한 IP를 회수, 필요한 컴퓨터에게 나눠주는 개념

사설망을 사용하는 경우

공인 IP 주소와 내부 IP 주소가 다를 경우

- 대부분 공유기를 통해 사설망이 운영되는 경우임
- 내부 IP 확인 방법

```
import socket
host_name = socket.gethostname()
ip = socket.gethostbyname(host_name)
print(ip)
```

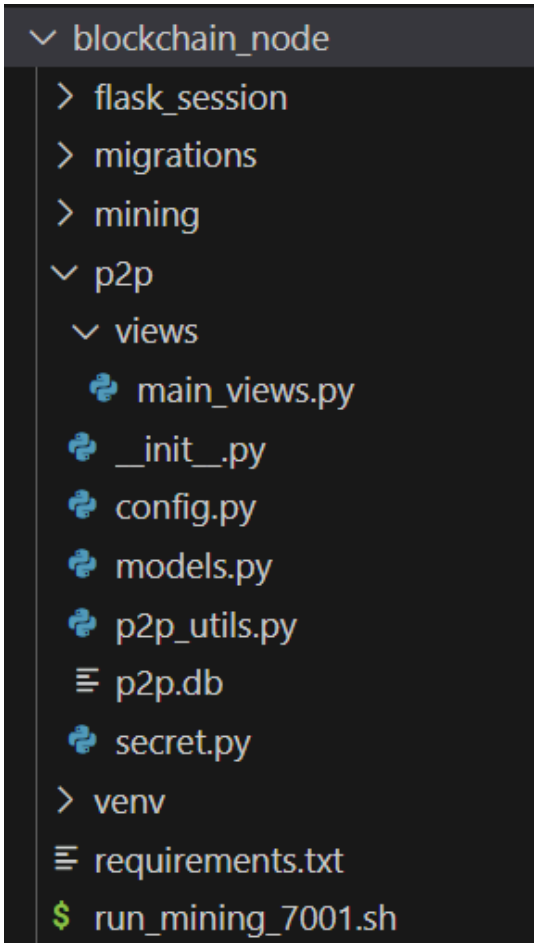
- 공인 IP 확인 방법

```
import requests
ip = requests.get('https://checkip.amazonaws.com').text.strip()
print(ip)
```

P2P 서버 구축 Overview

P2P 서버 구축

P2P 서버 구조



P2P Database 구조

blockchain_node > p2p > models.py > ...

```python
from p2p import db


class MiningNode(db.Model):
    '''블록체인 마이닝(채굴) 노드'''
    id = db.Column(db.Integer, primary_key=True)
    ip = db.Column(db.String(50), nullable=False)
    port = db.Column(db.String(50), nullable=False)
    timestamp = db.Column(db.Float)
```

p2p.db

blockchain_node > p2p > p2p.db

SELECT * FROM mining_node | Schema | Query Editor | Auto Reload | sqlite 3.31.1

Find | Other Tools...

	id INTEGER NOT NULL PRIMARY KEY	ip VARCHAR(50) NOT NULL	port VARCHAR(50) NOT NULL	timestamp FLOAT
1	2	203.252.240.43	22901	1689049199.602071
2	3	182.209.156.73	22901	1689049178.1504903
3	4	203.252.240.46	22901	1689049178.1323495

다음 강의

실습 해야죠 ^^

수고하셨습니다 ..^^..