

최종발표

블록체인을 활용한 오픈마켓 플랫폼 설계 및 구현

학번 201502085

이름 유정현

지도교수 정성호 교수님

작성일 2019.06.25

목차

CONTENTS

| | | |
|----|-------|-----|
| 01 | 주제 제안 | 3p |
| 02 | 중간 연구 | 12p |
| 03 | 추가 연구 | 28p |
| 04 | 향후 연구 | 46p |
| 05 | 부록 | 47p |

01 주제 제안

01-1

주제 제안 4p

01-2

연구 배경 5p

01-3

연구 목적 11p

01-1 주제 제안

블록체인을 활용한 오픈마켓 플랫폼



G마켓

11번가

A.
AUCTION.

INTERPARK

coupang
Color Your Days

01-2 연구 배경 (1 / 3)



오픈마켓(Open Market)

온라인상에 개인이나 소규모 업체가 직접 상품을 등록하여 구매자에게 판매할 수 있도록 하는 전자상거래 사이트



중간유통 과정이 생략된 C2C(Customer to Customer) 거래 구조

01-2 연구 배경 (1 / 3)

편리한 이용 + 저렴한 가격

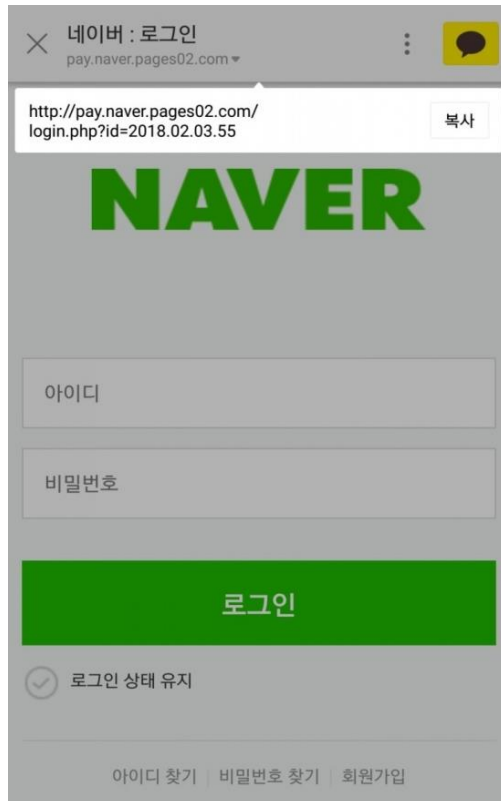


온라인 쇼핑 97% 중, 오픈마켓 이용비율 77%로 1위

01-2 연구 배경 (2 / 3)

돈은 물론 개인정보까지 털어가는 '가짜 네이버페이' 사기주의보

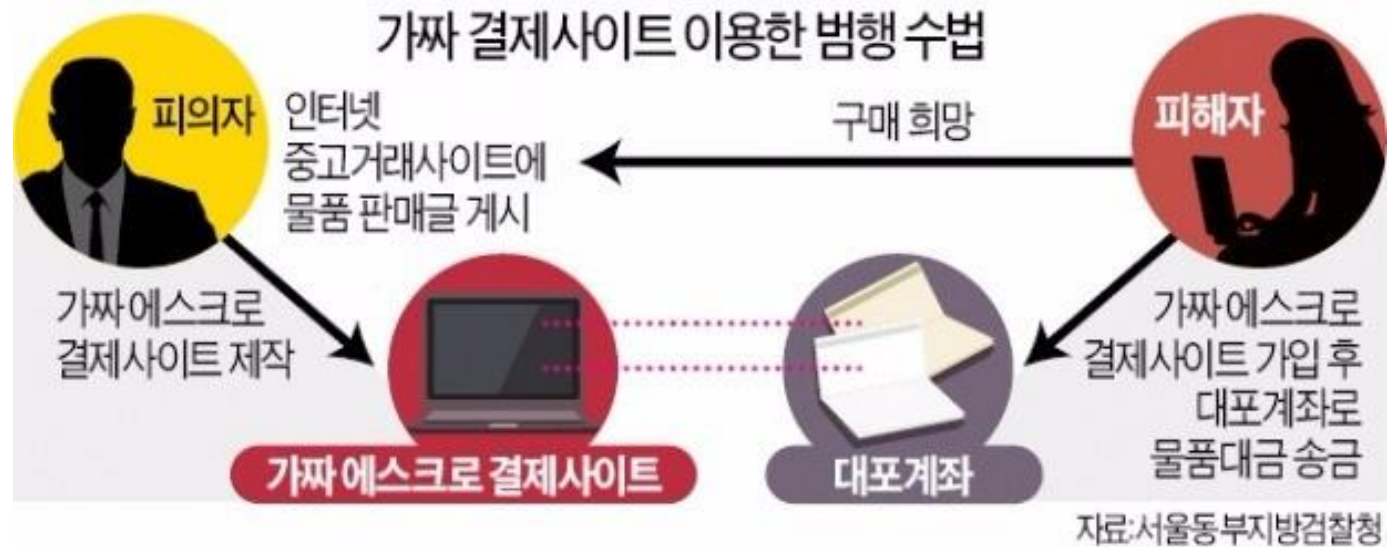
입력 2019.01.27 15:45 | 수정 2019.01.27 15:52



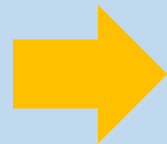
◆카톡 링크 주고 가짜 결제페이지 접속 유도

27일 업계에 따르면 가짜 네이버페이 링크(URL)을 만들어 안전거래를 유도한 뒤 돈은 물론 개인정보까지 털어가는 신종 사기가 활개를 치고 있다. 안전거래는 '에스크로(물건 배송 전까지 입금된 돈을 3자가 보관하는 기능)'의 일종으로, 구매자가 먼저 네이버페이에 송금한 뒤 물건을 배송받고 수취를 확인하면 비로소 판매자 계좌로 입금해주는 방식이다. 개인 간 거래에서 돈만 챙기고 물건은 보내지 않는 악덕 판매자를 막기 위해 2016년 도입됐다.

01-2 연구 배경 (2 / 3)



에스크로 (제 3자)의 개입으로 인한 거래 사기 사례



새로운 결제 시스템 필요

01-2 연구 배경 (3 / 3)

인터넷쇼핑 피해, 5년간 두 배 ↑ ... 소비자 신고 올 상반기만 4925건

입력 2018.09.09 19:14 | 수정 2018.09.10 03:25 | 지면 A14

11번가-G마켓-네이버 順...제품 하자에도 환불 안해줘

"오픈마켓, 중개 수수료 챙기고 품질·서비스는 책임 안져"

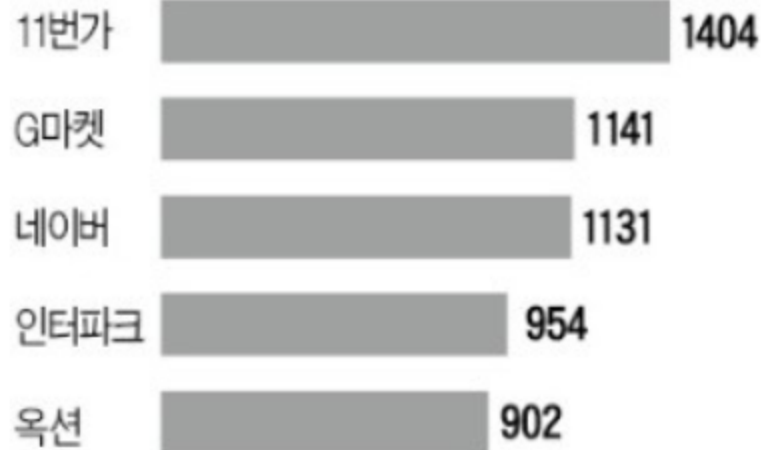
실시간 인기기

- 1 "물려받을 건 빗뚛"...상속포
- 2 "감싸주고 싶은 마음 없다"

01-2 연구 배경 (3 / 3)

인터넷쇼핑 업체별 소비자 피해 신고 건수

(단위:건)



※2013년 1월~2018년 6월 기준

자료:한국소비자원

계약 해지에 따른 **위약금 및 판매자의 계약 불이행 1위**



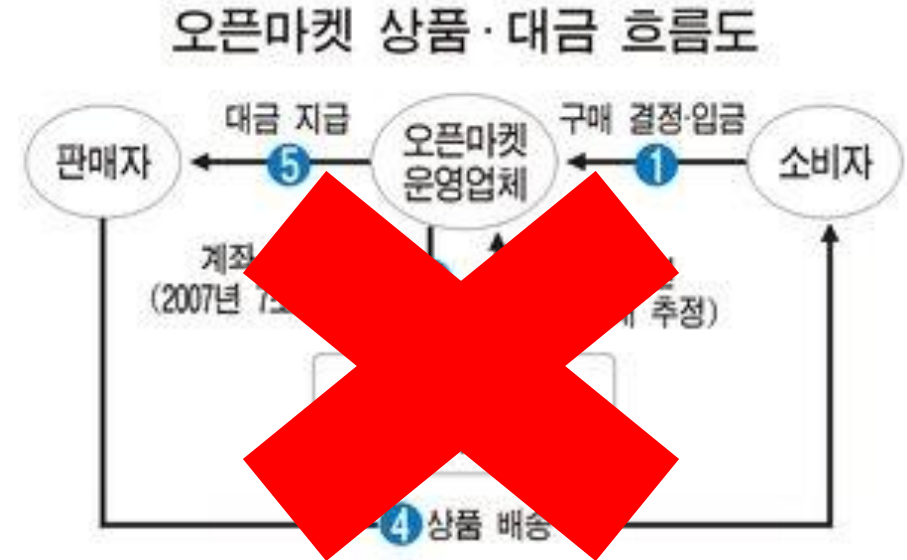
ex) 구입한 책상에 하자가 있어 반품을 요구 했지만,
판매자가 거부.., 구입한 에어컨에 바람이 나오지
않아 환불을 요구했지만, 이미 사용한 물품이라며 거부 등

01-3 연구 목적

블록체인을 이용한 오픈마켓 플랫폼



스마트 계약을 통한 자동 거래 이행
모든 거래는 블록체인에 영구히 저장



01 판매자와 구매자 간 제 3자 개입없이 신속하고 조작 없는 거래

02 블록에 저장된 기록 자체만으로 법적 효력 발생

02

중간 연구

구매자와 판매자 간 거래 구현
운영자 기능의 추가 구현

02-1

플랫폼 순서도 13p

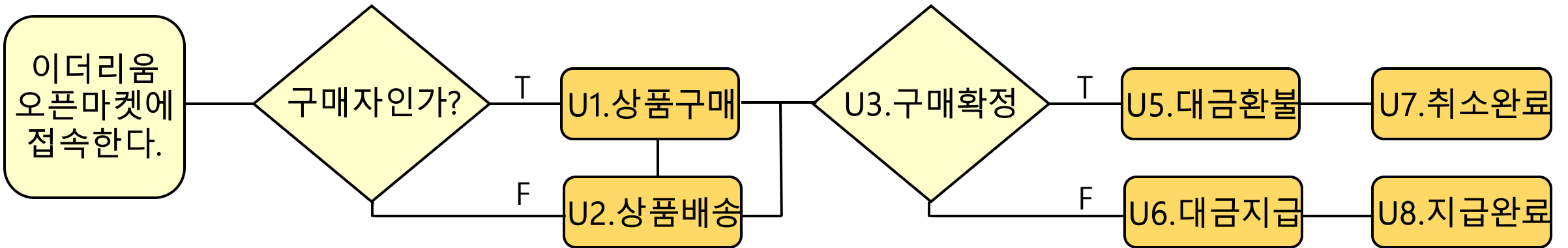
02-2

시스템 구성도 14p

02-3

구 현 내 용 16p

02-1 플랫폼 순서도

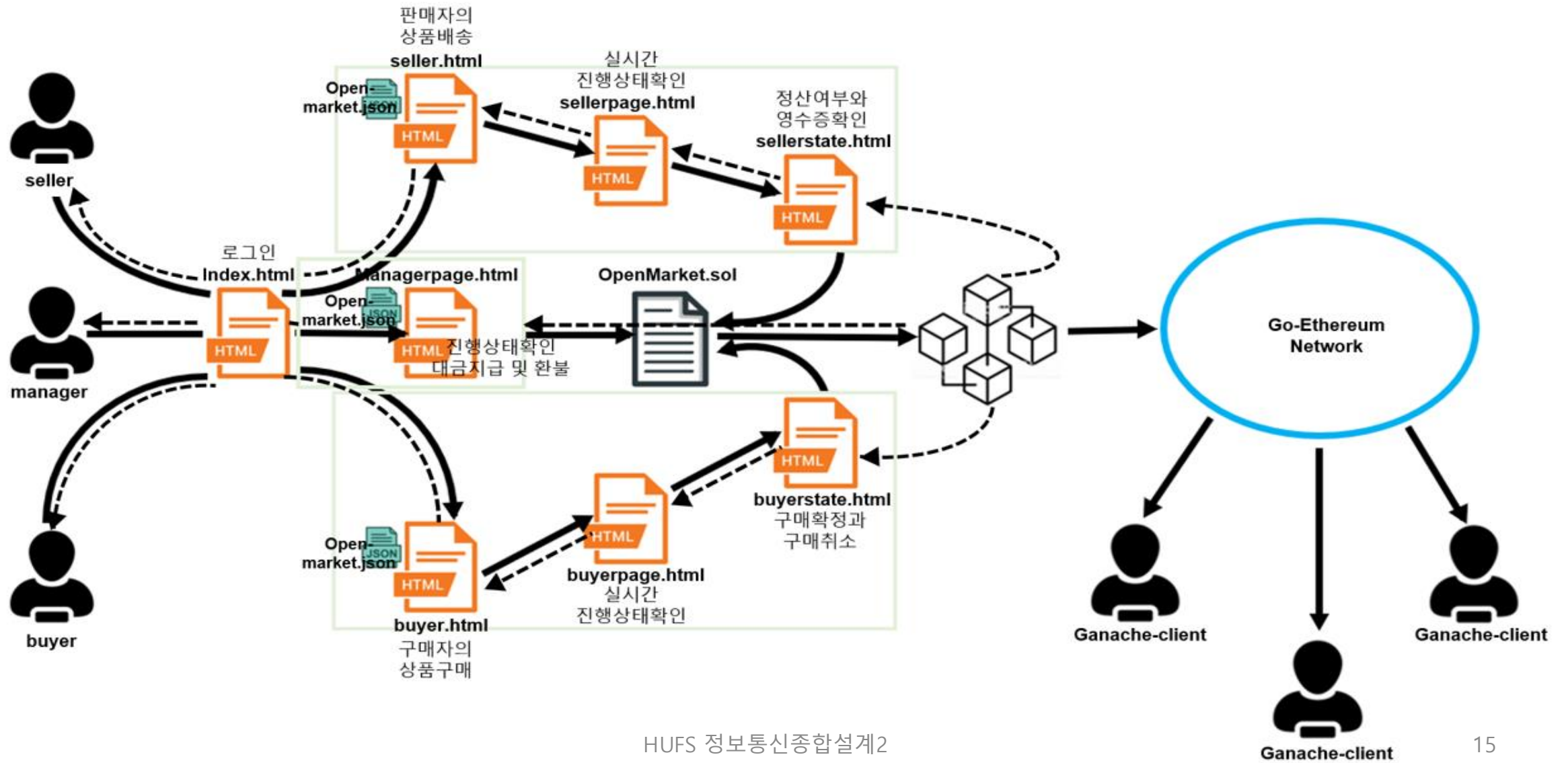


02-2 시스템 구성도 (1 / 2)

개발환경

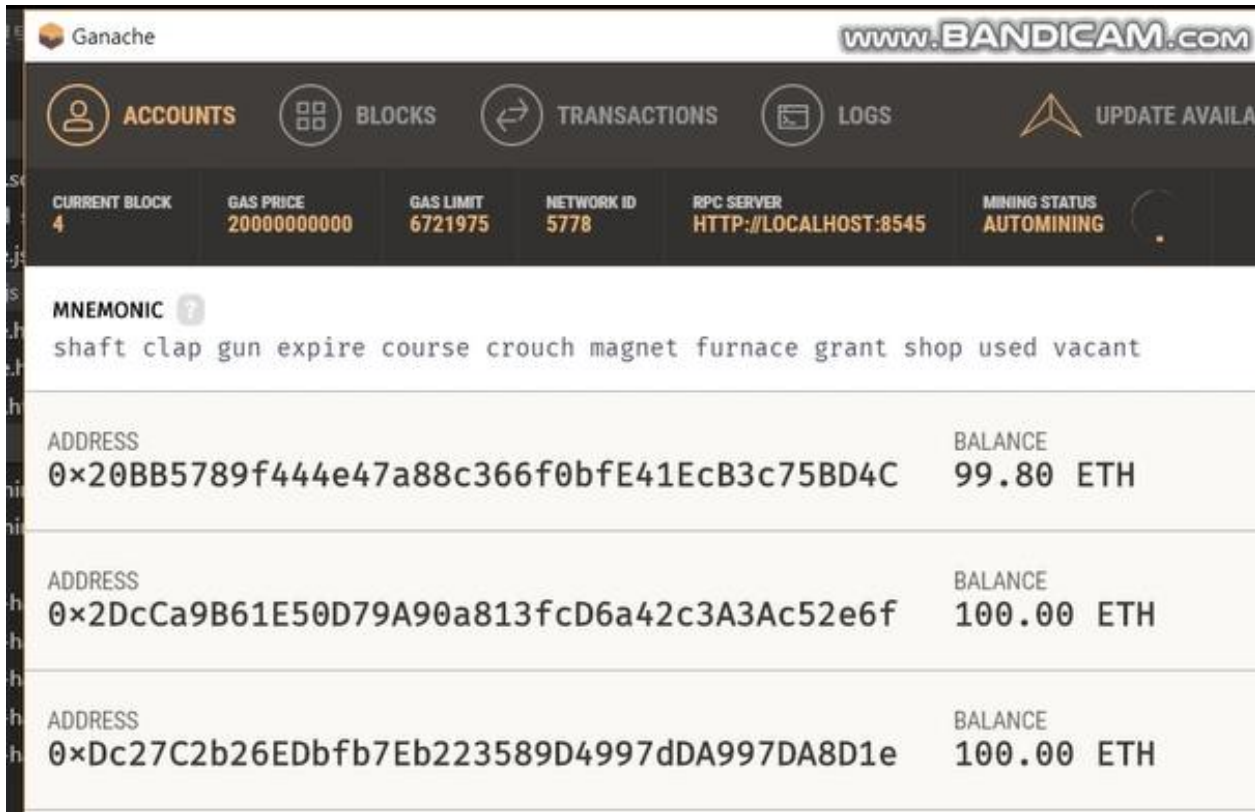
| | |
|---------|--|
| 하드웨어 | Intel® Core™ i7-8550U / 24GB RAM / 480GB HDD |
| 소프트웨어 | Window10 / Geth v1.8.12-stable / Ganache v1.10 / Meta mask v6.0 |
| 프로그래밍언어 | HTML / Truffle / Node.js / Solidity / Go |
| 개발도구 | Visual Studio Code / Chrome |

02-2 시스템 구성도 (2 / 2)



02-3 구현내용 (1 / 10)

01 truffle를 통해 smart contract 배포 후, 클라이언트 확인



*Ganache Client

각 계정에 기본 100ETH씩 제공됨

*거래 수수료

처음 컨트랙트 배포로,
컨트랙트 소유자 계정의 ETH값이 감소

*운영자 계정주소(컨트랙트 소유자 계정)

Meta mask상에서 Ganache1

*구매자 계정주소

Meta mask 상에서 Account1

*판매자 계정주소

Meta mask 상에서 Account2

02-3 구현내용 (2 / 10)

02 오픈마켓(html) 접속



이더리움 오픈마켓에 오신 것을 환영합니다!

03

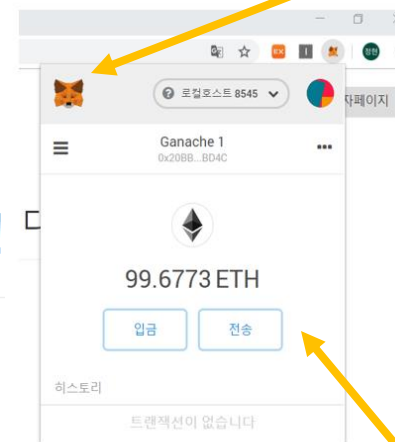
*구매자 접속

*판매자 접속

04

구매자입니다.

아니오, 판매자입니다.



운영자페이지

*운영자 접속

06

*Meta Mask

Client들이 소유한 금액(ETH) 값을 표시, 지갑역할

02-3 구현내용 (3 / 10)

03 구매자(Account2) 상품 구매 (1 / 2)

이더리움 오픈마켓

구매자로 로그인하셨습니다.





HOME 마이페이지

*첫 페이지 이동

*마이페이지
구매내역 확인
거래 진행상태 확인
구매확정페이지 이동

*상품정보
상품번호 및 종류
배송유형
가격(ETH단위)표시

*구입버튼

| 상품 | 상품 | 상품 | 상품 |
|---|---|--|--|
|  상품번호: 0 종류: 공기청정기 배송: 무료배송 가격(ETH): 5.23 |  상품번호: 1 종류: 참외 배송: 무료배송 가격(ETH): 1.56 |  상품번호: 2 종류: 닭가슴살 배송: 유료배송 가격(ETH): 1.74 |  상품번호: 3 종류: 건미리 팩트 배송: 무료배송 가격(ETH): 2.87 |
| 구입 | 구입 | 구입 | 구입 |

02-3 구현내용 (3 / 10)

03 구매자(Account2) 상품 구매 (2 / 2)

구입자 정보

이유진

23

서울시 중랑구 상봉동

04231

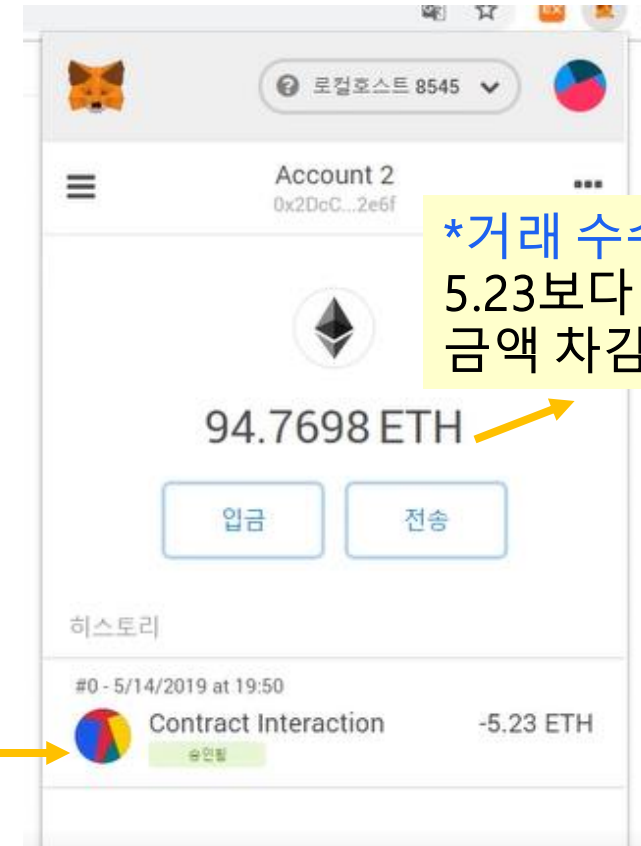
01032324242

닫기

제출

구매자정보,
판매자정보,
거래정보가
블록에 기록

*거래 실행



02-3 구현내용 (4 / 10)

04 판매자(Account3) 상품 배송 (1 / 2)

이더리움 오픈마켓

판매자로 로그인하셨습니다.

HOME 관리페이지

***관리페이지**
구매된 상품내역 확인
거래 진행상태 확인
정산 페이지 이동

***첫 페이지 이동**

***구매자정보 버튼**
구매된 상품에 대하여
구매자의 정보 확인 가능

상품

상품번호: 0
종류: 공기청정기
배송: 무료배송
가격(ETH): 5.23

구입자정보

상품

상품번호: 1
종류: 참외
배송: 무료배송
가격(ETH): 1.56

상품

상품번호: 2
종류: 닭가슴살
배송: 유료배송
가격(ETH): 1.74

상품

상품번호: 3
종류: 견미리 팩트
배송: 무료배송
가격(ETH): 2.87

2 in 1 고속충전

PLEOMAX

02-3 구현내용 (4 / 10)

04 판매자(Account3) 상품 배송 (2 / 2)

구입자 정보

계정주소: 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
이름: 이유진
나이: 23
집주소: 서울시 중랑구 상봉동
우편번호: 4231
핸드폰번호:1032324242

닫기

판매자 정보 입력

아이온

서울시 노원구 중계동

04241

01042421212

배송

판매자정보,
배송정보가
블록에 기록

*배송 실행

배송정보

판매자
회사계정: 0xdc27c2b26edbf7eb223589d4997dda997da8d1e
회사이름: 아이온
회사주소: 서울시 노원구 중계동
우편번호: 4241
핸드폰번호:1042421212

에서,
구매자
계정주소: 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
이름: 이유진
나이: 23
집주소: 서울시 중랑구 상봉동
우편번호: 4231
핸드폰번호:1032324242

로 배송되었습니다.

닫기

02-3 구현내용 (5 / 10)

05 구매자(Account2) 구매 확정



구매확정/구매취소

구매를 확정하시면, 판매자에게 리뷰를 작성할 수 있습니다.
리뷰는 구매자 접속페이지에서 확인가능

구매확정

구매를 취소하시면, 구매자분께 돈이 환불됩니다.

구매취소

닫기

02-3 구현내용 (6 / 10)

06 운영자(Ganache1), 판매자(Account2)에게 대금 지급

MetaMask Notification

로컬호스트 8545

Ganache 1 → 0x0b4B...48...

CONTRACT INTERACTION

5.23

DETAILS DATA

GAS FEE 0.013806
변환 비율을 찾을 수 없습니다

TOTAL 5.243806
변환 비율을 찾을 수 없습니다

EDIT

거부 승인

이더리움 오픈마켓 운영자 페이지

지급 정보

구매자: 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

구매확정일자: 20190530

구매후기: 진짜 좋네요

20190603

취소 지급

*지급 정보

구매자의 구매확정일자, 구매후기 확인
지급날짜 입력 후 지급

*지급영수증 발행

0 번 상품이 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f 구매자로부터 구입되었습니다.


0 번 상품이 0xdc27c2b26edbf7eb223589d4997dda997da8d1e 판매자로부터 배송되었습니다.

0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f 구매자가 0 번 상품의 구매를 확정하였습니다.

0x20bb5789f444e47a88c366f0bfe41ecb3c75bd4c 운영자가 0 번 상품을 정산하였습니다.

거래목록

로켓아이마트 WINIX



상품번호: 0
종류: 공기청정기
배송: 무료배송
가격(ETH): 5.23

정산이 완료되었습니다.

지급 영수증 정산완료

환불불가


02-3 구현내용 (7 / 10)

07 판매자(Account3) 정산 완료

이더리움 오픈마켓

localhost:3000/sellerstate.html

구매된 상품



롯데아이마트
x
WINIX

상품번호: 0
종류: 공기청정기
배송: 무료배송
가격(ETH): 5.23

지금 이 완료되었습니다.
지금 영수증

지금 영수증

해당 상품은 다음과 같이 정산되었습니다.
구매자: 0x2dcca9b61e50d79a90a813cd6a42c3a3ac52e6f
구매확정일자: 20190530
구매후기:진짜 좋네요
운영자:0x20bb5789f444e47a88c366f0bfe41ecb3c75bd4c
정산일자:20190603

닫기

*운영자계정에서 판매자 계정으로 5.23ETH가 전달 된 것을 확인

BLOCKS

TRANSACTIONS

LOGS

UPDATE AVAILABLE

GAS PRICE
20000000000

GAS LIMIT
6721975

NETWORK ID
5778

RPC SERVER
HTTP://LOCALHOST:8545

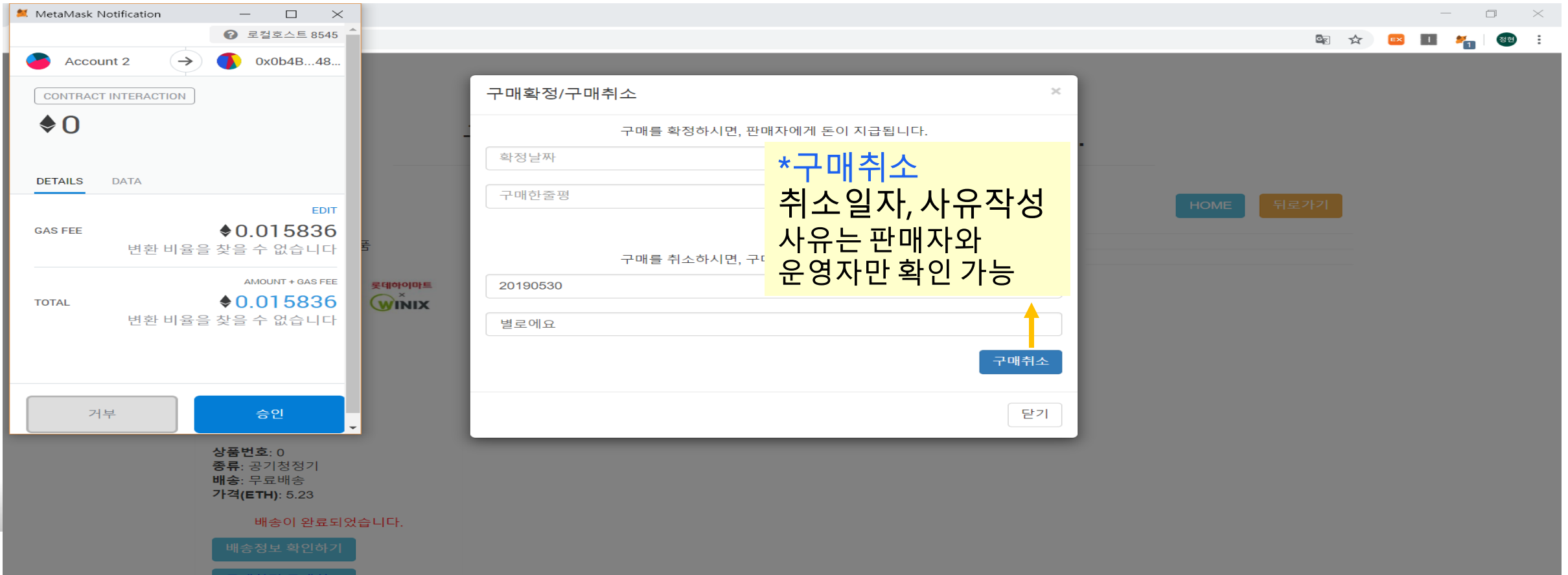
MINING STATUS
AUTOMINING

gun expire course crouch magnet furnace grant shop used vacant

| | |
|------------------------------------|-----------------------|
| 39f444e47a88c366f0bfe41ecb3c75BD4C | BALANCE 99.67 ETH |
| 861E50D79A90a813cd6a42c3A3Ac52e6f | BALANCE 94.74 ETH |
| 26EDbf7Eb223589D4997dDA997DA8D1e | BALANCE 105.22 ETH |

02-3 구현내용 (8 / 10)

05 구매자(Account2) 구매 취소



02-3 구현내용 (9 / 10)

06 운영자(Ganache1), 구매자(Account1)에게 대금 환불

이더리움 오픈마켓 × +
localhost:3000/managerpage.html

0 번 상품이 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
0 번 상품이 0xdc27c2b26edbf7eb223589d
0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

거래목록

상품번호: 0
종류: 공기청정기
배송: 무료배송
가격(ETH): 5.23

정산불가
환불하기

환불 정보

구매자: 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
구매취소일자: 20190530
취소사유: 별로예요

환불날짜

취소 환불

***환불 정보**
구매자의 구매취소일자, 취소사유 확인
환불날짜 입력 후 환불

***환불영수증 발행**

0 번 상품이 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
0 번 상품이 0xdc27c2b26edbf7eb223589d
0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f
0x20bb5789f444e47a88c366f0bfe41ecb3

거래목록

상품번호: 0
종류: 공기청정기
배송: 무료배송
가격(ETH): 5.23

정산불가
환불하기

환불이 완료되었습니다.

환불 영수증
정산불가
환불완료

02-3 구현내용 (10 / 10)

07 구매자(Account2) 환불 완료

이더리움 오픈마켓

localhost:3000/managerpage.html

0 번 상품이 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

0 번 상품이 0xdc27c2b26edbf7eb223589d0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

0x20bb5789f444e47a88c366f0bfe41ecb3c75bd4c

거래목록

거울 이미지

롯데아이마트 x WINIX

상품번호: 0

종류: 공기청정기

배송: 무료배송

가격(ETH): 5.23

환불이 완료되었습니다.

환불 영수증

정산불가

환불완료

환불 영수증

해당 상품은 다음과 같이 환불되었습니다.

구매자: 0x2dcca9b61e50d79a90a813fcd6a42c3a3ac52e6f

구매취소일자: 20190530

취소사유: 별로에요

운영자: 0x20bb5789f444e47a88c366f0bfe41ecb3c75bd4c

환불일자: 20190603

닫기

*운영자계정에서 구매자 계정으로 5.23ETH가 전달 된 것을 확인

뒤로가기

BLOCKS

TRANSACTIONS

LOGS

UPDATE AVAILABLE

SEARCH FOR

GAS PRICE 20000000000

GAS LIMIT 6721975

NETWORK ID 5778

RPC SERVER HTTP://LOCALHOST:8545

MINING STATUS AUTOMINING

un expire course crouch magnet furnace grant shop used vacant

9f444e47a88c366f0bfe41ecb3c75bd4c BALANCE 99.67 ETH

61E50D79A90a813fcd6a42c3A3Ac52e6f BALANCE 99.98 ETH

03

추가 연구

블록체인 기술 검증을
통한 연구 타당성 입증

03-1

연구 내용 29p

03-2

실험 구성 32p

03-3

실험 내용 36p

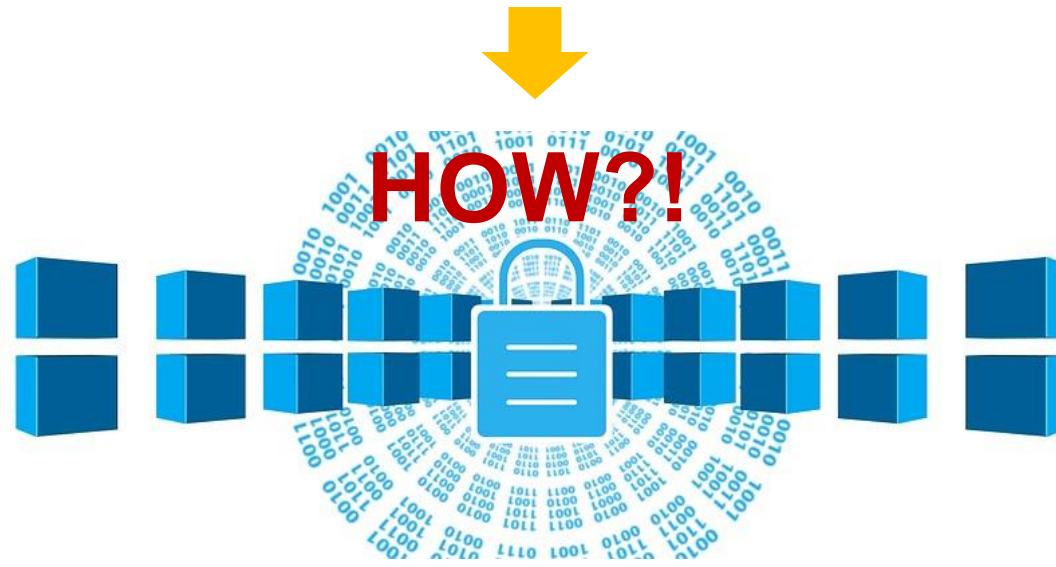
03-4

결론 42p

03-1 연구 내용 (1 / 3)

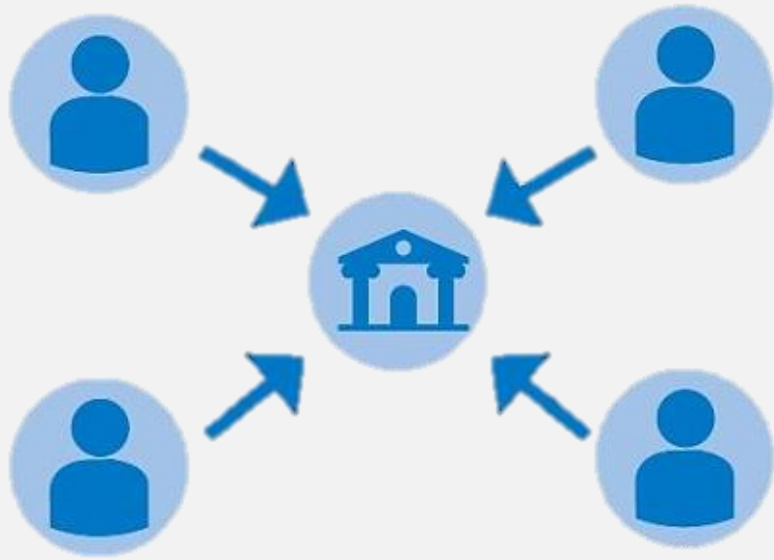
01 제 3자 개입없이 신속하고 조작 없는 거래

02 블록에 영구히 저장된 기록으로 법적 효력 발휘



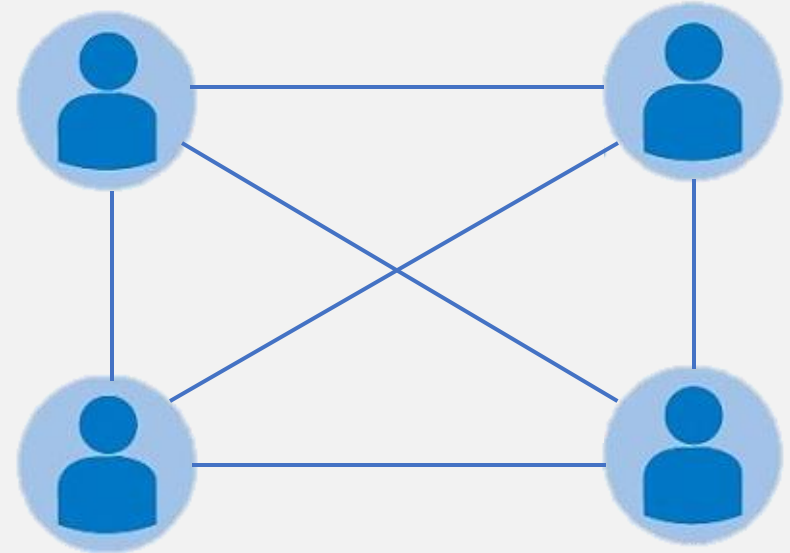
03-1 연구 내용 (2 / 3)

기존 거래방식



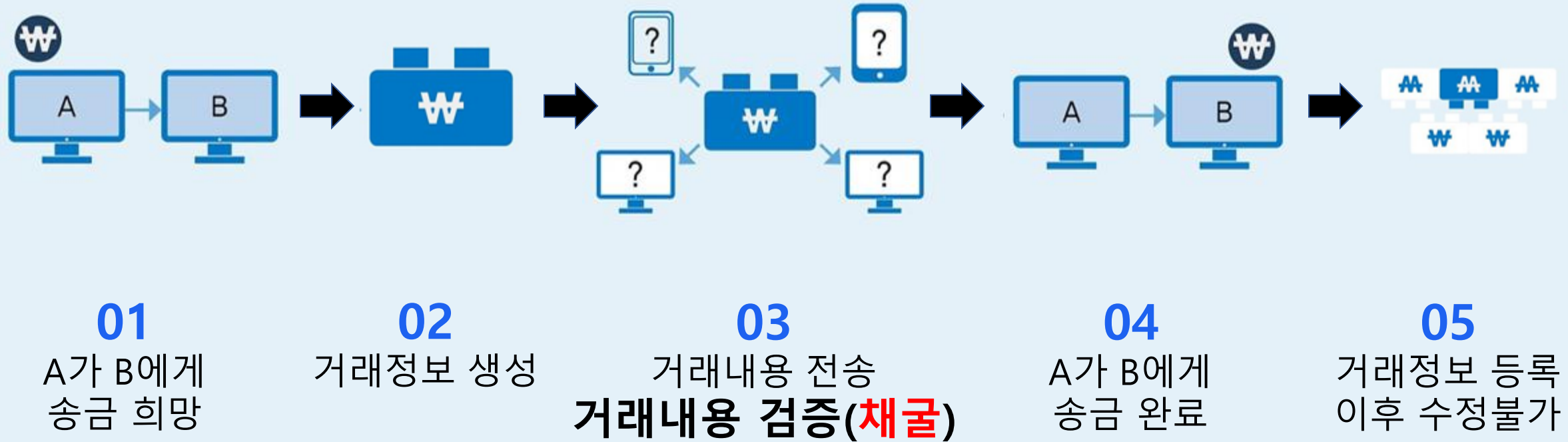
은행이 중앙에서 거래장부 관리
통일된 거래내역 유지

블록체인 거래방식



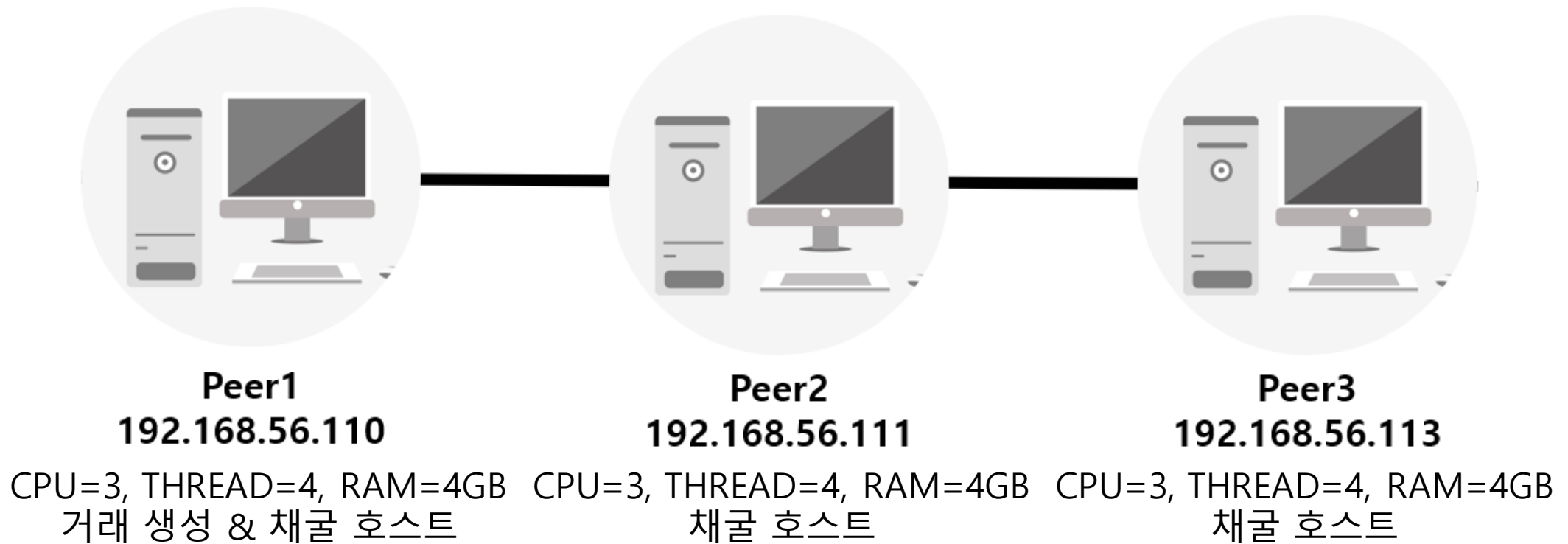
중앙에서 거래에 관여하는 노드가 없음

03-1 연구 내용 (3 / 3)



03-2 실험 구성 (1 / 4)

* Linux 환경



03-2 실험 구성 (2 / 4)

* Multi-node Blockchain Network 구축

1.*Geth --networked 1005 --port 30331/30332/30334 console 2>> 1005번 id의 이더리움 테스트넷으로 포트번호 30331~4을 이용하여 접속한 네트워크 Console창

Peer1

Peer2

Peer3

2. *Net.peerCount(): 현재 1005번 네트워크에 접속한 노드는 3개
*Eth.blockNumber(): 세 노드가 보유하는 현재 블록의 개수 7923개

03-2 실험 구성 (3 / 4)

* 제네시스 블록(Genesis.json) 생성

```
{
  "config": {
    "chainId": 1005,
    "homesteadBlock": 0,
    "eip155Block": 0,
    "eip158Block": 0
  },
  "alloc"           : {},
  "coinbase"        :
"0x000000000000000000000000000000000000000000000000",
  "difficulty"      : "0x40000",
  "extraData"        : "",
  "gasLimit"         : "0xffffffff",
  "nonce"            : "0x0000000000000000",
  "mixhash"          :
"0x000000000000000000000000000000000000000000000000",
  "parentHash"       :
"0x000000000000000000000000000000000000000000000000",
  "timestamp"        : "0x00"
}
```

1.Config{}: 이더리움 네트워크의 설정

2.chainId: 현재 체인을 식별하는 값, 1005로 네트워크ID와 동일해야 함

3.Difficulty, nonce, mixHash:

채굴의 난이도를 결정하는 부분

03-2 실험 구성 (4 / 4)

* 거래 계정과 채굴 계정 생성

The image shows three virtual machines (Peer1, Peer2, Peer3) running Geth. The terminal windows display the following commands and outputs:

Peer1:

```
yjh@yjh-VirtualBox: ~/test
geth history static-nodes.json
yjh@yjh-VirtualBox:~/test$ rm -rf keystore
yjh@yjh-VirtualBox:~/test$ geth --networkid 1005 --datadir /home/yjh/test --port 30331 console 2>> /home/yjh/test/geth.log
```

Peer2:

```
yjh2@yjh2-VirtualBox: ~/test
> net.peerCount
2
> eth.blockNumber
7272
> exit
yjh2@yjh2-VirtualBox:~/test$ ls
genesis.json geth.log keystore
geth history static-nodes.json
yjh2@yjh2-VirtualBox:~/test$ rm -rf keystore
yjh2@yjh2-VirtualBox:~/test$ geth --networkid 1005 --datadir /home/yjh2/test --port 30332 console 2>> /home/yjh2/test/geth.log
Welcome to the Geth JavaScript console!

instance: Geth/v1.8.27-stable-4bcc0a37/linux-amd64/go1.10.4
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

> net.peerCount
2
> eth.accounts
[]
> personal.newAccount("miner")
"0x4ce7d2694357bf6e1b5efd1531e4189b97d4465e"
```

Peer3:

```
yjh4@yjh4-virtualbox: ~/test
File Edit View Search Terminal Help
> net.peerCount
2
> eth.blockNumber
7272
yjh4-virtualbox:~/test$ rm -rf keystore
yjh4-virtualbox:~/test$ geth --networkid 1005 --datadir /home/yjh4/test --port 30334 console 2>> /home/yjh4/test/geth.log
Welcome to the Geth JavaScript console!

instance: Geth/v1.8.27-stable-4bcc0a37/linux-amd64/go1.10.4
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0

> net.peerCount
2
> eth.accounts
[]
> personal.newAccount("miner")
"0x525c20a56d3ded166c4b580b487804791a5af55e"
```

Annotations:



- ***sender:** 송금 하는 계정 (accounts[1])
- ***receiver:** 송금 받는 계정 (accounts[2])
- ***miner:** 거래 검증(채굴) 계정 (각 노드의 accounts[0])

03-3 실험내용 (1 / 5)

01 Sender(Accounts[1])가 Receiver(Accounts[2])에게 10ETH 송금 희망

* 현재 Sender가 소유한 ETH는 19.9,
Receiver가 소유한 ETH는 15

```
> web3.fromWei(eth.getBalance(eth.coinbase),"ether")  
118.750021  
> web3.fromWei(eth.getBalance(eth.accounts[1]),"ether")  
19.999958  
> web3.fromWei(eth.getBalance(eth.accounts[2]),"ether")  
15
```



1. `web3.fromWei(eth.getBalance(),"ether")`:
이더리움 가상화폐의 또 다른 단위인 Wei로
표시되는 금액을 ether로 변환

2. `eth.sendTransaction(from, to, value)`
Sender의 계정을 unlock 시킨 후 거래 생성
(from에서 to로 value만큼의 ETH 전송)

```
> personal.unlockAccount(eth.accounts[1])  
Unlock account 0x554d0735fe802fb1ee7baf951a13263ca201d08e  
Passphrase:  
true  
> eth.sendTransaction({from:eth.accounts[1], to:eth.accounts[2],  
value:web3.toWei(10, "ether")})  
"0x500b6a0970044dbe877d42e9c530f915380717d25be265742398ed094fd157df"
```

03-3 실험내용 (2 / 5)

02 거래 정보가 블록(Block) 형태로 생성

```
> eth.pendingTransactions
[
  {
    blockHash: null,
    blockNumber: null,
    Rhythmbox: "0x554d0735fe802fb1ee7baf951a13263ca201d08e",
    gas: 90000,
    gasPrice: 1000000000,
    hash: "0x500b6a0970044dbe877d42e9c530f915380717d25be265742398ed094fd157df",
    input: "0x",
    nonce: 2,
    r: "0xcc65e2378cf26c770d99acdb216cfb8e7cb0a58ffc786e9bdfa0f632e7dba326",
    s: "0x46cd0e5ee314b7e9c168afd8478698291788ab1453f630d38c39078421273a1c",
    to: "0x47a96df570f03e57d36607484237343c6475b91b",
    transactionIndex: 0,
    v: "0x71c",
    value: 1000000000000000000
  }
]
```

* **eth.pendingTransactions**: 생성된 거래 정보를 보여주는 명령어

* **blockNumber**: 아직은 블록번호가 지정되지 않은 것을 확인

Transaction 정보

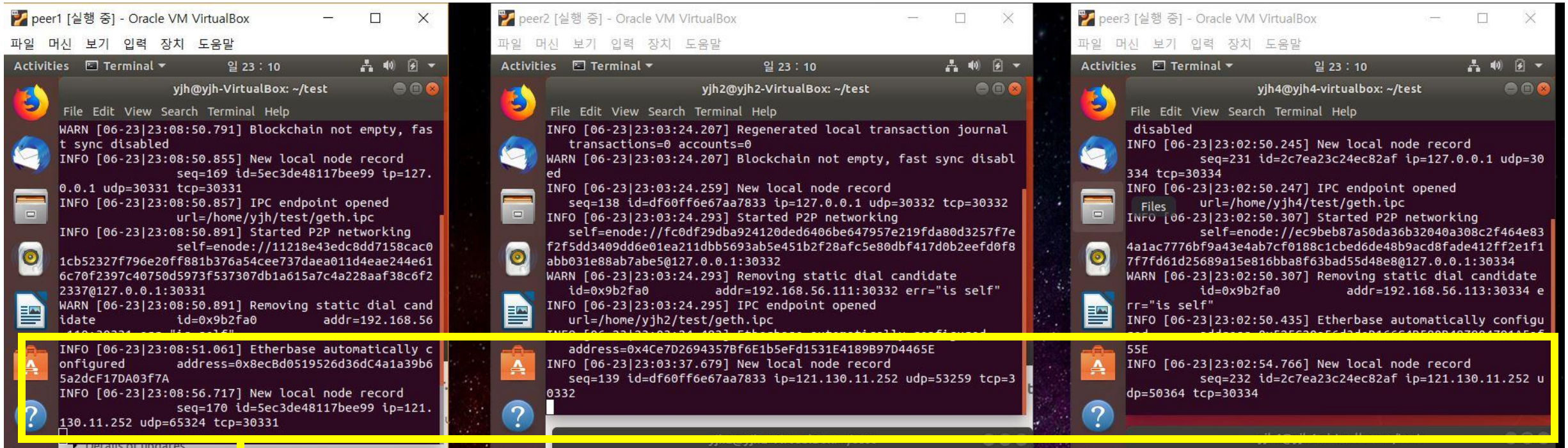
***from**: sender의 계정주소

***to**: 송금받는 계정주소(Transaction1: 2번 계정, Transaction: 0번 계정)

***value**: from에서 to로 보내는 ETH 값

03-3 실험내용 (3 / 5)

03-1 거래 정보가 블록 형태로 모든 참여자들에게 전송



*New Local new record: 새로운 거래 정보가 생성되었음을 알림
블록체인 네트워크에서 일어나는 모든 상황을 네트워크의 전 노드가 투명하게 공유

03-3 실험내용 (3 / 5)

03-2 참여자들은 거래 정보가 올바른 지 확인 (채굴)

```
peer2 [실행 중] - Oracle VM VirtualBox
파일 머신 보기 입력 장치 도움말
Activities Terminal 월 16 : 14
yjh2@yjh2-VirtualBox: ~/test
File Edit View Search Terminal Help
number=7927 sealhash=bc64e8...bdd393 uncles=0 txs=0 gas=0 f
ees=0 elapsed=408.18µs
INFO [06-24|16:13:36.195] Commit new mining work
number=7927 sealhash=a4bc8d...0216b1 uncles=0 txs=1 gas=210
00 fees=2.1e-05 elapsed=1.149ms
INFO [06-24|16:13:45.012] Successfully sealed new block
number=7927 sealhash=a4bc8d...0216b1 hash=39b893...f5d62a ela
psed=8.817s
INFO [06-24|16:13:45.012] mined potential block
number=7927 hash=39b893...f5d62a
INFO [06-24|16:13:45.013] Commit new mining work
number=7928 sealhash=be316d...38319d uncles=0 txs=0 gas=0
fees=0 elapsed=139.238µs

yjh2@yjh2-VirtualBox: ~/test
File Edit View Search Terminal Help
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net
:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
> net.peerCount
2
> eth.blockNumber
7923
> miner.start(4)
Show Applications
>
```

```
peer3 [실행 중] - Oracle VM VirtualBox
파일 머신 보기 입력 장치 도움말
Activities Terminal 월 16 : 14
yjh4@yjh4-virtualbox: ~/test
File Edit View Search Terminal Help
=0 gas=0 fees=0 elapsed=138.16µs
INFO [06-24|16:13:40.832] Commit new mining work
number=7927 sealhash=29500d...a13ba9 uncles=0 txs=0 gas=0
fees=0 elapsed=409.086µs
INFO [06-24|16:13:44.726] Imported new chain segment
blocks=1 txs=1 mgas=0.021 elapsed=7.381ms m
sps=2.845 number=7927 hash=39b893...f5d62a cache=4.54kB
INFO [06-24|16:13:44.726] Commit new mining work
number=7928 sealhash=2d5461...45ffc9 uncles=0 txs=0 gas=0
fees=0 elapsed=319.728µs
INFO [06-24|16:13:44.727] Commit new mining work
number=7928 sealhash=2d5461...45ffc9 uncles=0 txs=0 gas=0
fees=0 elapsed=742.689µs

yjh4@yjh4-virtualbox: ~/test
File Edit View Search Terminal Help
modules: admin:1.0 debug:1.0 eth:1.0 ethash:1.0 miner:1.0 net
:1.0 personal:1.0 rpc:1.0 txpool:1.0 web3:1.0
> net.peerCount
2
> eth.blockNumber
7923
> miner.start(4)
null
>
```

* **geth.log**: 블록체인 네트워크 상에서 일어나는 모든 상황을 공유하고 있음

* **Commit new mining work**:
아까 Pending 되었던 Transaction들을 mining 해야 함을 알림

* **mined potential block**:
채굴하고 있음을 나타냄

* **miner.start(4)**: 스레드 4개로 채굴
* **eth.mining=True**: 채굴 중임을 확인

03-3 실험내용 (4 / 5)

04 Sender(Accounts[1])가 Receiver(Accounts[2])에게 송금 완료

```
> web3.fromWei(eth.getBalance(eth.accounts[1]), "ether")
9.999937
> web3.fromWei(eth.getBalance(eth.accounts[2]), "ether")
25ether
> 
```

* 현재 Sender가 소유한 ETH는 19.9에서 9.9로,
Receiver가 소유한 ETH는 15에서 25로 변화

*eth.GetBalance:

Accounts1에서 Accounts2로 송금이 완료되었음

*거래 수수료

블록체인 네트워크의 운영을 위해 사용됨

03-3 실험내용 (5 / 5)

05 거래 정보는 블록체인에 저장되어 수정 불가

```
eth.getTransaction("0x500b6a0970044dbe877d42e9c530f915380717d25be265742398ed094fd157df")
```

```
{  
  blockHash: "0x39b893452dc10f4ada4ed5595d52ba...",  
  blockNumber: 7927,  
  from: "0x554d0735fe802fb1ee7baf951a13263ca201d08e",  
  gas: 900000,  
  gasPrice: 10000000000,  
  hash: "0x717d25be265742398ed094fd157df",  
  input: "0x...",  
  nonce: 2,  
  r: "0xcc65e2378cf26c770d99acdb216cfb8e7cb0a58ffc786e9bdfa0f632e7dba326",  
  s: "0x46cd0e5ee314b7e9c168afd8478698291788ab1453f630d38c39078421273a1c",  
  to: "0x47a96df570f03e57d36607484237343c6475b91b",  
  transactionIndex: 0,  
  v: "0x7fc",  
  value: 1000000000000000000  
}
```

*Transaction 정보
거래 정보가
블록체인 7927번 블록에 저장

[illegible]

03-4 결론 (1 / 4)



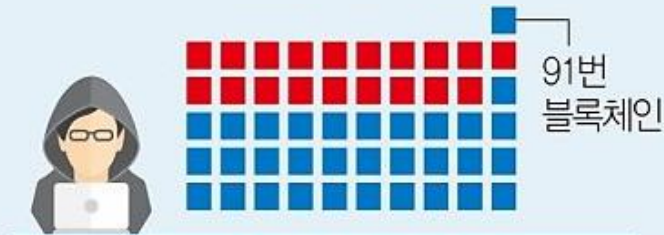
각각의 블록마다 바로 앞 블록의 거래내역이 저장돼 있어 사슬(Chain)처럼 연결된 구조
특정 블록이 해킹되어도 **이전** 블록에 의해 복구 가능

03-4 결론 (2 / 4)



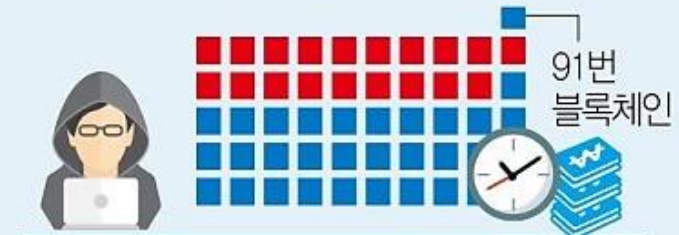
01

해커가 74번의 거래 블록을
위조하려고 시도



02

91번 블록이 생성되기 전
74~90번 블록의 기록을
모두 수정해야 함



03

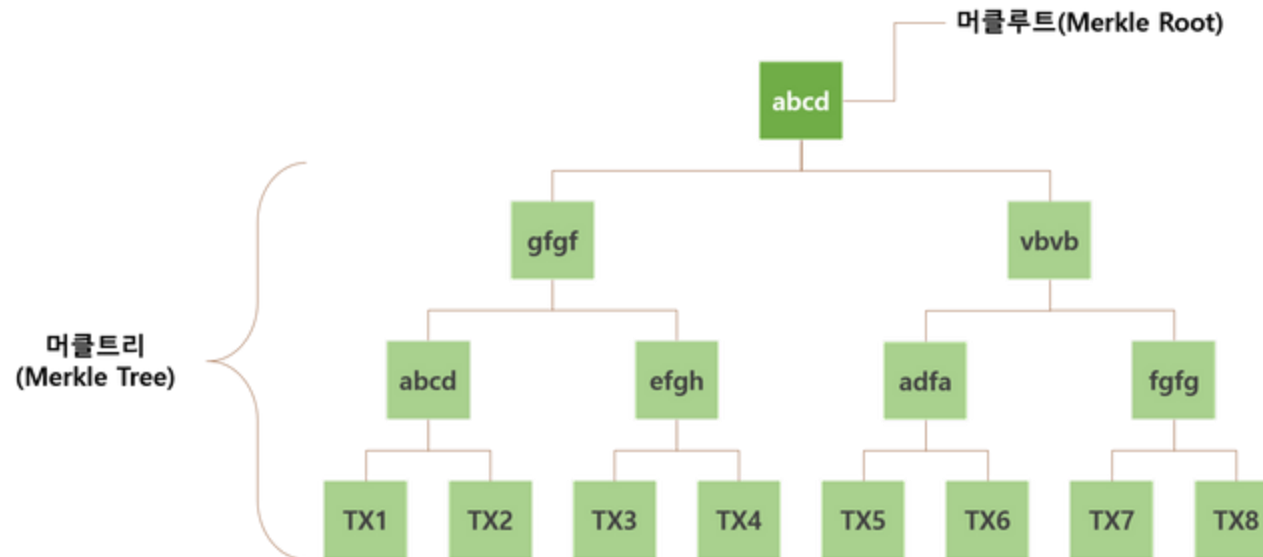
이 과정에서 막대한 비용과
시간 소요

03-4 결론 (3 / 4)

각각의 거래가 **SHA-256**으로 암호화를 거치며 **꼭대기 거래 루트 생성(머클트리)**



거래의 경로를 따라가며 변조된 거래 추적



03-4 결론 (4 / 4)

01 탈중앙성



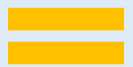
02 신속성



03 투명성



04 보안성



공인된 제 3자 없이 **개인 간 거래** 가능

다수가 자동으로 **거래를 승인하고 기록**

모든 거래 기록에 공개적으로 접근 가능

거래 기록을 다수가 소유해 **해킹 불가능**

블록체인을 활용한 오픈마켓 플랫폼 !!

수수료 절감



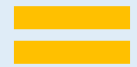
신속성 극대화



거래 투명화



보안비용 절감



04 향후 연구

04-1

로그인

블록체인의 계정주소를 인증하는 로그인 구현

04-2

암호기술

블록체인에서 사용하는 암호화 알고리즘에 대한 연구

05 부

록

05-1

참고 문헌 48p

05-2

연구 일정 49p

05-1 참고 문헌

[1] 블록체인 방식의 전자투표 시스템 구현 및 성능 개선 방안 연구
A Study on Performance Improvement and Implementation of
Electronic Voting System using Blockchain
아주대학교 정보통신대학원 정보보호공학 유헌우, 2016년 2월
<http://www.riss.kr/link?id=T14010220>

[2] 블록체인 기반 스마트 계약을 활용한 전자상거래 에스크로 대체 플랫폼 구축
A study on establishment of an alternative e-business escrow platform using
block chain based smart contract
동국대학교 국제정보보호대학원 정보보호학과 장승일, 2018년 5월
<http://www.riss.kr/link?id=T14877174>

[3] 블록체인 기반 신분증명 시스템
한국외국어대학교 정보통신공학과 김영상, 이재혁 2018년 7월

05-2 연구일정

| 개발내용 | | 3월 | | | | 4월 | | | | 5월 | | | | 6월 | | | |
|---------|----------|----|--|--|--|----|--|--|--|----|--|--|--|----|--|--|--|
| 관련 연구 | 관련 지식 학습 | | | | | | | | | | | | | | | | |
| | 관련 기술 학습 | | | | | | | | | | | | | | | | |
| 설계 | 요구사항 정의 | | | | | | | | | | | | | | | | |
| | 설계 | | | | | | | | | | | | | | | | |
| 구현 및 검증 | 네트워크 구축 | | | | | | | | | | | | | | | | |
| | 플랫폼 구축 | | | | | | | | | | | | | | | | |
| | 모의 공격실험 | | | | | | | | | | | | | | | | |
| | 모의 테스트 | | | | | | | | | | | | | | | | |
| | 보수 및 점검 | | | | | | | | | | | | | | | | |

감사합니다.
