2018/05/08 @상명대학교

캡스톤디자인 최종 발표

- BIDaaS (Blockchain based ID as a Service) -

발표자: 임연주(chocobeeyj@gmail.com)

지도교수: 이종혁(jonghyouk@smu.ac.kr)

Underdog 임연주, 송영준, 곽수진

상명대학교 컴퓨터공학과

목 차

개 요

- BIDaaS(Blockchain based ID as a Service)
  - 블록체인 기반 신원 관리 서비스
  - 다양한 제휴 서비스를 하나의 ID로 통합 로그인
    - Microsoft, SK C&C 등에서 현재 도입 중
- 개발 배경
  - 기존 IDaaS 서비스의 한계점
    - IDaaS(Identity as a Service)
      - 클라우드 환경에서 신원(Identity)를 확인하기 위해 통합 접근 관리 인프라
    - 클라우드 환경에서 데이터 유출 사고 급증

개 요

- 개발 배경
  - 보다 안전한 로그인 서비스를 위해 블록체인을 활용한 **BIDaaS**를 개발
    - 제안된 논문을 기반으로 시스템 구현
      - “BIDaaS (Blockchain based ID as a Service)”, IEEE Access 2017.12

- 개발 목표

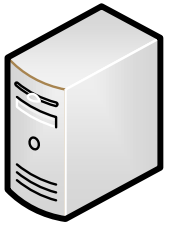
1. 블록체인 연구
2. 블록체인 기반 User 공개키 획득 알고리즘 연구 및 개발
3. BIDaaS 기반 Partner 어플리케이션 및 웹 서비스 프로토타입 개발

개 요

- 관련 연구
  - 블록체인 (Blockchain)
    - P2P 네트워크 환경에서 거래 데이터를 블록에 저장하고 이 블록들이 연결되어 체인을 이루는 기술
      - 2009 년 1 월, 사토시 나카모토 (Satoshi Nakamoto)에 “비트코인”에서 처음 제안됨
    - 특징
      - 중앙 관리자가 필요하지 않음
      - 분산형 원장 구조로 위/변조가 어려움
    - 현재 금융, 에너지, 물류 등 다양한 분야에서 활용

BIDaaS

- 시스템 엔티티



ethereum

1. Provider
   - Partner에게 BIDaaS 제공
   - 블록체인을 유지 및 관리
   - User의 가상 ID, 공개키 등을 서명과 함께 블록체인에 기록

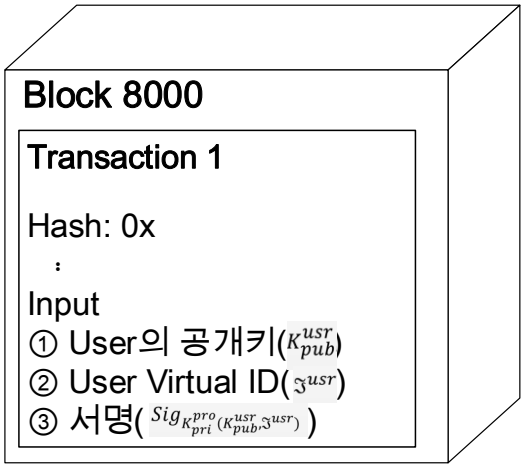
2. Partner
   - User에게 서비스 제공
   - 블록체인에 읽기 권한만 부여됨

3. User
   - 어플리케이션, 웹 사이트 사용하는 일반 사용자

BIDaaS

- 블록체인 필드 구성

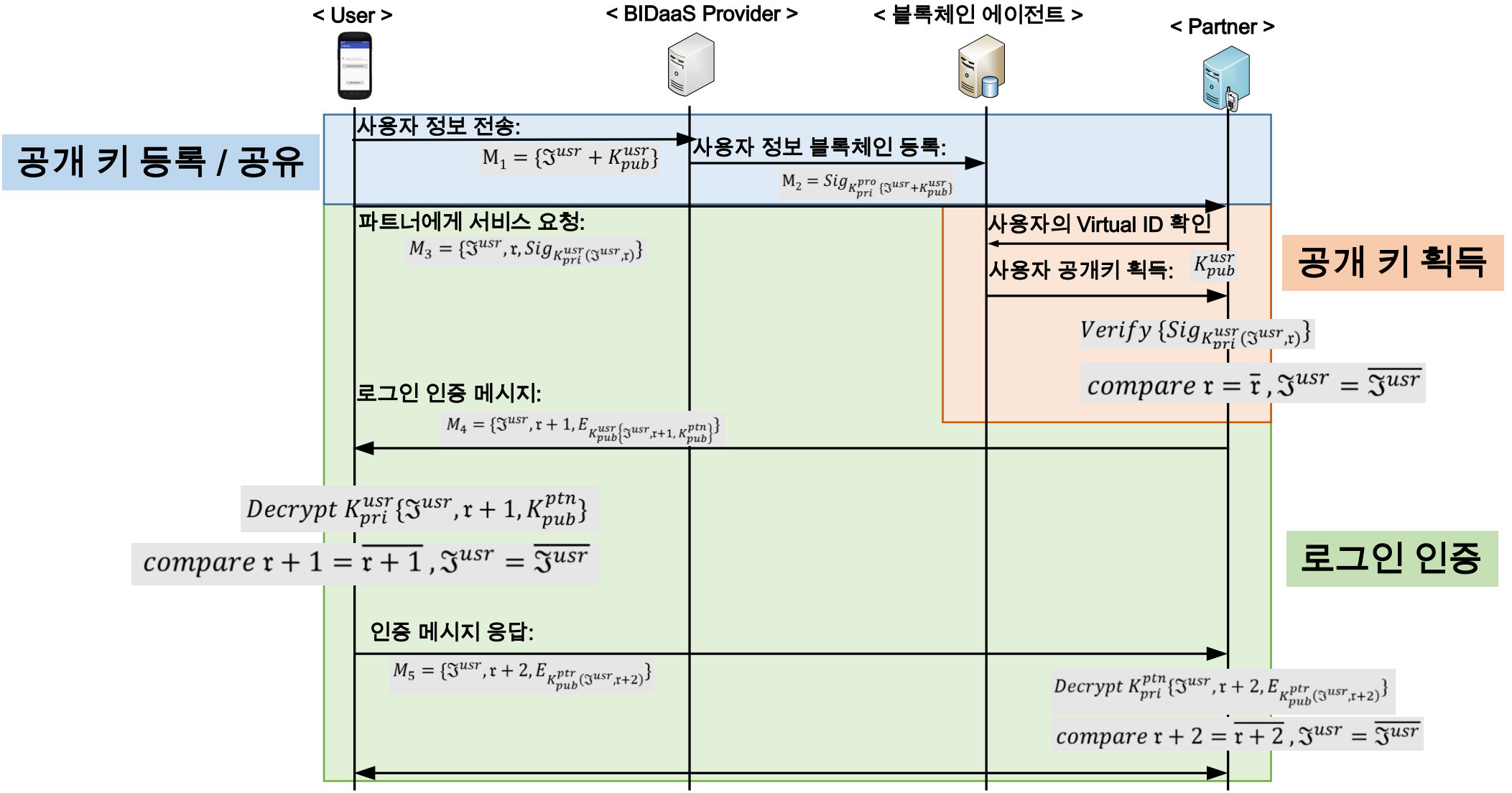
필드	내용
사용자 공개키(PUB)	RSA 쌍 중 공개키(1024bits)
사용자 가상 ID(VID)	공개키를 기반으로 해쉬 값(SHA256) 생성
서명 값(SIG)	공개키와 가상 ID을 Provider의 개인키로 서명한 값(RSA-SHA256)



- 블록체인에 저장되는 데이터
  - 필드 데이터를 hex로 변환 후, 트랜잭션 input 값에 삽입

BIDaaS

- 사용자 공개 키 등록/획득/공유



BIDaaS

- 프로토타입 개발 및 검증
  - 트랜잭션 발생
    - 데이터 서명 후, 블록체인에 삽입
      - hex 값으로 변환
    - 생성된 트랜잭션 정보 출력
      - 블록넘버
      - 트랜잭션 해시
      - input 정보
    - 블록체인에 저장된 사용자 정보 출력
      - error: 응답 에러 처리
      - PUBLIC: 사용자 공개키
      - VID: 가상 ID
      - SIG: 서명(sha256)
    - 서명 검증

------블록체인에 데이터 저장을 위해 hex 변환 중------

data:
0x7b224552524f52223a7b22636f6465223a3230302c226d657373616765223a22737563636573732273d2c225055424c4943223a222d2d2d2d2d424547494e20505
5424c4943204b45592d2d2d2d2d5c6e4d4947664d4130474353714753496233445145424151554141344746e414443426951 4b4267514438434274 4d6b644551
7a78452b6f38335461334d74447844575c6e416d546c4c676b5578687434567535334549486c2b356a7633615335704f43437657645678 6d3031754f39302f564
6755a6953307241495438436831595369655c6e31684b4a5168446967557347787468355956356641 2f455361 4f34765431386c53696d3354376b4d4e7076534b
2b7a44794e392f43647276766c3838757678675c6e692b4b6163725469 4a346f547871796c4951494441514142 5c6e2d2d2d2d2d454e44205055424c4943204b4
5592d2d2d2d2d5c6e222c22564944223a22534841323536287573657 25f7075626c69632e70656d293d2066346561323434306530376137653261343035626561
6361343731333664636662373436326634323364373831666639663264626138343062643562663534325c6e222c22534947223a22376139366364663161656 26
2626539666665353039336430326163343336333235376532326532343061376664396632316132386139663263633632656534393365633763353461643662
3764623137373534336436316166363131353263316231646233396635323139343031613935623139376335373465643666262646133396366653465663363 6
3333236353261656436616333373337356630653932306430623235613334363130396466393430396637356664313161633939663565323362613431 64646238
3737393735636666630333436623630653832633065656433636231373661396162646366326664373039303837373939323063227d

------블록체인 트랜잭션 해시-------

txhash: 0xe85c32a78aaf80f700c4cdc182d5760c0a7db41da72f262bc298e89b8cdb557b

------저장된 User 관련 블록체인 정보------

Block number: 8709
Transaction hash: 0xe85c32a78aaf80f700c4cdc182d5760c0a7db41da72f262bc298e89b8cdb557b
Input:
0x7b224552524f52223a7b22636f6465223a3230302c226d657373616765223a22737563636573732273d2c225055424c4943223a222d2d2d2d2d424547494e205
5424c4943204b45592d2d2d2d2d5c6e4d4947664d4130474353714753496233445145424151554141344746e414443426951 4b4267514438434274 4d6b644551
7a78452b6f38335461334d74447844575c6e416d546c4c676b5578687434567535334549486c2b356a7633615335704f43437657645678 6d3031754f39302f564
6755a6953307241495438436831595369655c6e31684b4a5168446967557347787468355956356641 2f455361 4f34765431386c53696d3354376b4d4e7076534b
2b7a44794e392f43647276766c3838757678675c6e692b4b6163725469 4a346f547871796c4951494441514142 5c6e2d2d2d2d2d454e44205055424c4943204b4
5592d2d2d2d2d5c6e222c22564944223a22534841323536287573657 25f7075626c69632e70656d293d2066346561323434306530376137653261343035626561
6361343731333664636662373436326634323364373831666639663264626138343062643562663534325c6e222c22534947223a22376139366364663161656 26
2626539666665353039336430326163343336333235376532326532343061376664396632316132386139663263633632656534393365633763353461643662
3764623137373534336436316166363131353263316231646233396635323139343031613935623139376335373465643666262646133396366653465663363 6
3333236353261656436616333373337356630653932306430623235613334363130396466393430396637356664313161633939663565323362613431 64646238
3737393735636666630333436623630653832633065656433636231373661396162646366326664373039303837373939323063227d

------블록체인에 저장된 값을 문자열로 변환 중------

Encodeing Input: {"ERROR":{"code":200,"message":"success"},"PUBLIC":"-----BEGIN PUBLIC KEY-----
₩nMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQD8CBtMkdEQzxE+o83Ta3MtDxDW₩nAmTILgkUxht4Vu53EIHI+5jv3aS5p0CCvWdVxmOIu09O/VFuZiSOrAIT8ChlY
Sie₩n1hKJQhDigUsGxth5YV5fA/ESa04vTI8ISim3T7kMNpvSK+zDyN9/Cdrvvl88uvxg₩ni+KacrTiJ4oTxqyIlQIDAQAB₩n-----END PUBLIC KEY-----
₩n","VID":"SHA256(user_public.pem)=
f4ea2440e07a7e2a405beaca47136dcfb7462f423d781ff9f2dba840bd5bf542₩n","SIG":"7a96cdflaebbbe9ffe5093d02ac4363257e22ce240a7fd9f21a28a
9f2cc62ee493ec7c54ad6b7db177543d61af61152c1b1db39f5219401a95b197c574ed3fbbda39cfe4ef3cc32652aed6ac37375f0e920d0b25a346109df9409f7
5fd1lac99f5e23ba41ddb877975cff0346b60e82c0eed3cb176a9abdcf2fd70908779920c"}

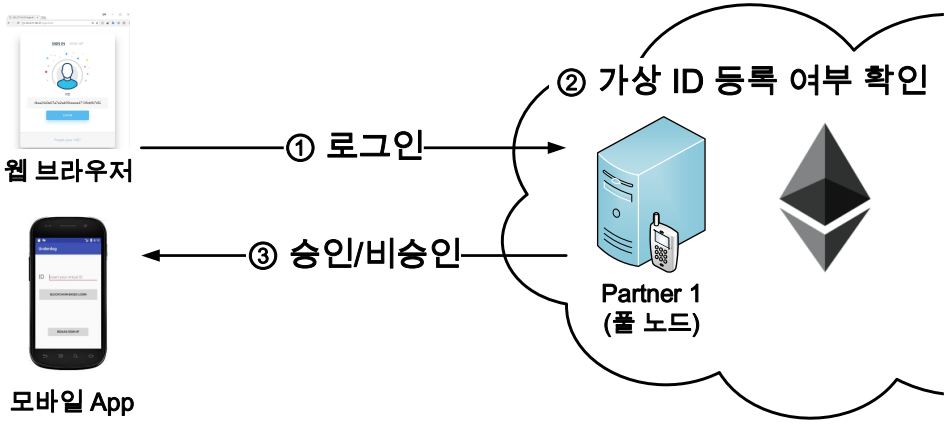
------블록체인에 저장된 값 서명 검증-----

valid

결 론

- 기대 효과
  1. 안전하고 간편한 로그인 서비스
     - 하나의 ID로 관련 어플리케이션, 웹 등에서 간편 로그인
  2. P2P 기반의 블록체인 적용방안 연구
     - 서버의 과부화 감소
  3. 인증, 권한, 데이터베이스 관리 등의 반복성을 피함

- 시연 영상
  1. 사용자 정보 등록 및 로그인
     - 모바일 App
  2. 사용자 로그인 과정
     - 웹 브라우저



감사합니다!

백업1: 테스트 환경

- 개발환경 구성
  - Provider
    - 블록체인 플랫폼:이더리움(Ethereum) 블록체인
      - Go-ethereum: 1.8.2-stable
  - Partner 모바일/웹 서버
    - Apache: 2.4 (Ubuntu)
    - PHP: 7.0
  - User(모바일 유저)
    - 안드로이드 Nogart(7.0), html(웹 브라우저)

백업2: 프로젝트 진행과정

- 프로젝트 진행 계획
  - 간드 차트

3월 2일 ~ 5월 8일	Week								
	1	2	3	4	5	6	7	8	9
BIDaaS 환경 정의	■	■							
엔티티 및 기능 명세	■	■	■						
인증 프로세스 정의			■	■	■				
블록체인 네트워크 구현 및 연동				■	■				
웹 인터페이스 구현					■	■	■	■	
테스트							■	■	■
보완 및 검증							■	■	■

백업2: 프로젝트 진행과정

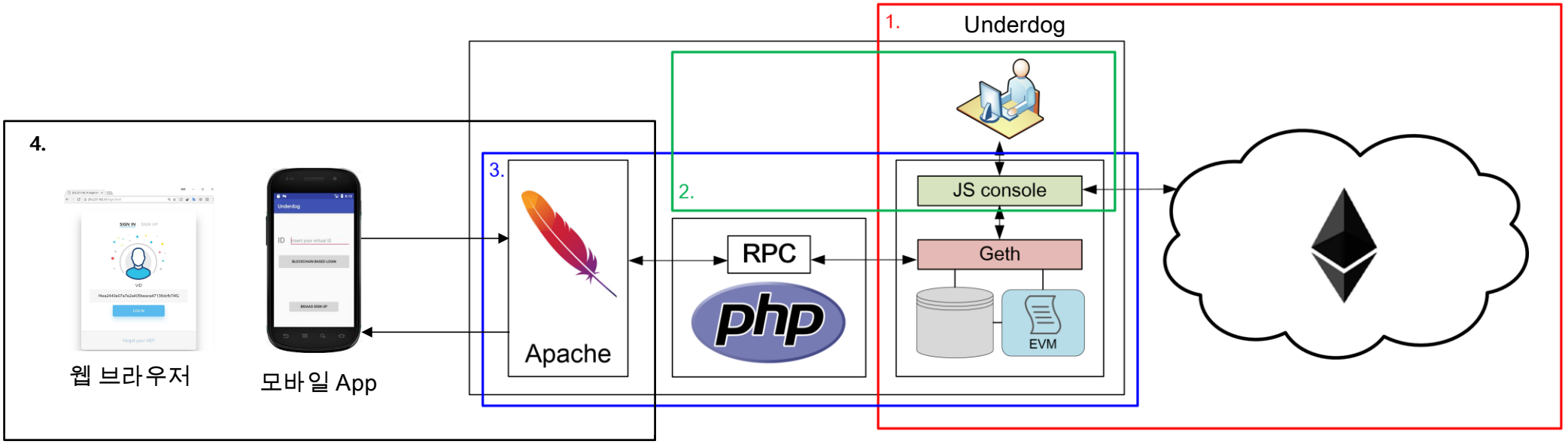
- 프로젝트 참여도

	임연주	송영준	곽수진
블록체인 분석	2017년 후반기 진행		
IDaaS 환경 정의		O	O
엔티티 및 기능 명세	O	O	O
인증 프로세스 정의 및 명세	O	O	O
블록체인 네트워크 구현	O		
블록체인-웹 연동	O		
모바일 어플리케이션 구현		O	
웹 인터페이스 구현			O
테스트	O	O	O
보완 및 검증	O	O	O

백업 3: 동작과정

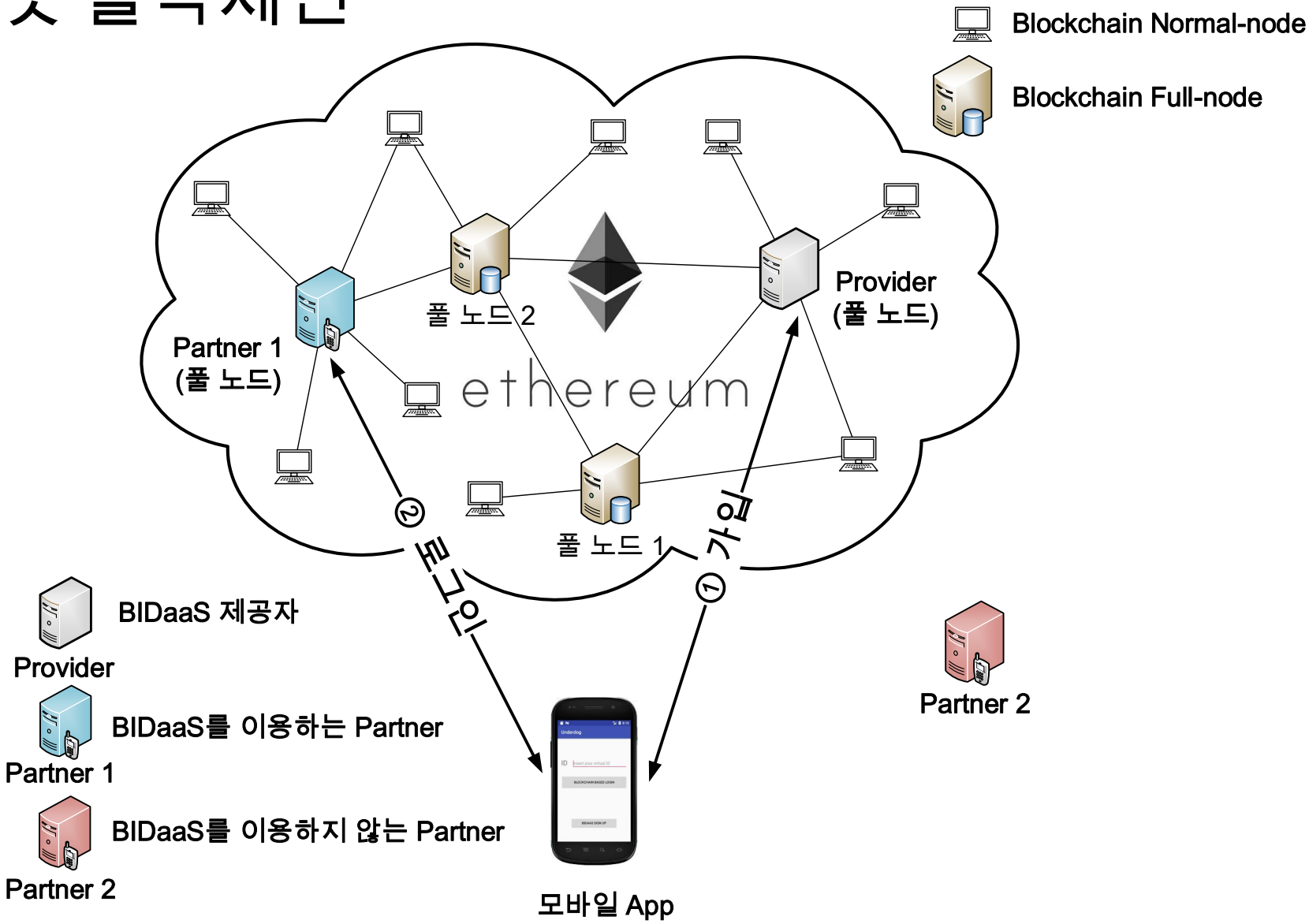
- BIDaaS 동작 과정

1. 블록체인 네트워크
2. Provider
3. Partner
4. User(웹, 모바일 App)



백업4: 네트워크 구성도

- 프라이빗 블록체인



백업5: 주요 함수

- provider.js: 블록체인에 사용자 등록

타입	함수명	파라미터	설명
function	str2hex	Char string	사용자 데이터 hex값으로 변환
function	hex2str	Char string	Tx input의 저장된 hex 값 문자열로 변환
function	hextobin	Char string	서명 값(이진 파일)을 hex로 변환
function	json_make	Char string1, Char string2, Char string3	string들을 json 형식으로 변환 및 병합
function	open_ether	Object ethereum	블록체인 객체 생성
function	sendtx	Object tx	생성한 트랜잭션 객체 생성
function	print_tx	Object tx, Number i	입력된 트랜잭션 객체 tx 정보 출력

백업5: 주요 함수

- partner.php: 블록체인에 저장된 사용자 확인

타입	함수명	파라미터	설명
function	connect	-	데이터 베이스 연결
function	getid	-	Post 데이터 추출 후, db 삽입
function	result	id, VID	해당 id에 대한 레코드 출력
function	resultval	-	id가 있는 경우와 없는 경우를 반환
function	str2hex	Char string	사용자 데이터 hex값으로 변환
function	hex2str	Char string	Tx input의 저장된 hex 값 문자열로 변환
function	hextobin	Char string	서명 값(이진 파일)을 hex로 변환
function	encryption	Char string	데이터 암호화 후, 송신
function	decryption	Char string	수신 후, 데이터 복호화
function	gettx	Object tx, String VID	가상 ID를 통해 해당 트랜잭션 객체 추출
function	print_tx	Object tx, Number i	입력된 트랜잭션 객체 tx 정보 출력

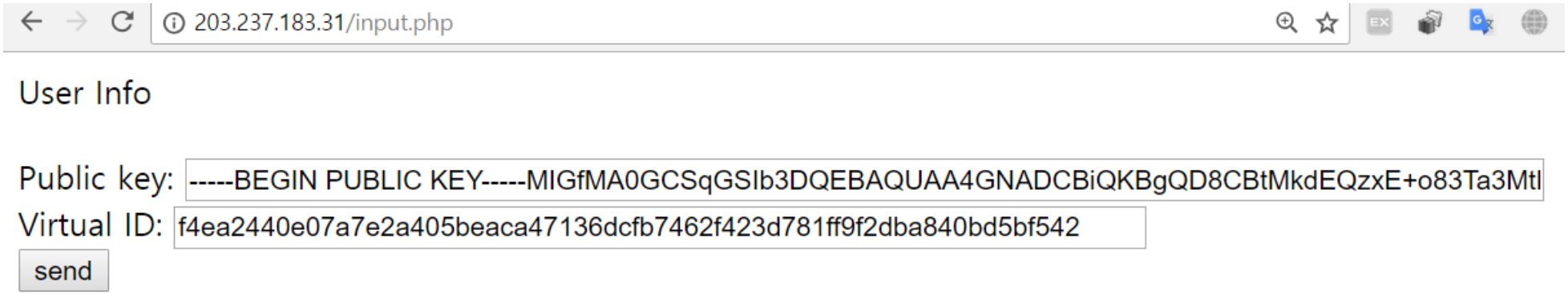
백업5: 주요 함수

- user.java: 가상 아이디 기반 로그인 어플리케이션

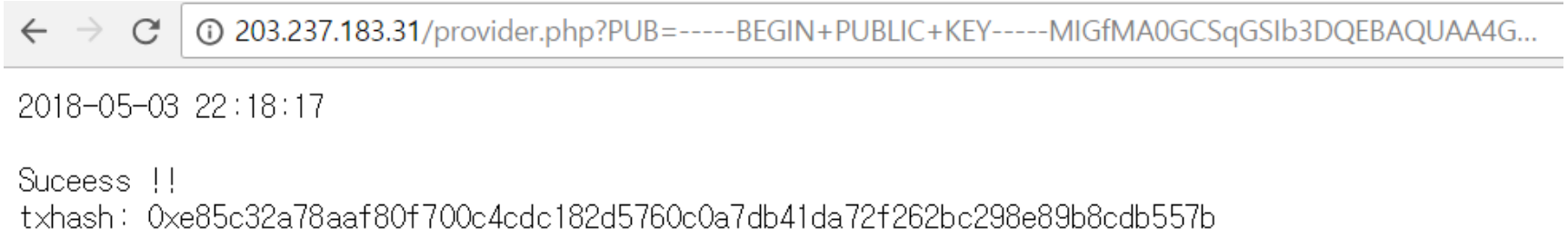
타입	함수명	파라미터	설명
function	login	String id	Post 방식으로 http 전송
function	Clisignup	-	Sign up 페이지로 액티비티 이동
function	getdata	-	getStringExtra로 액티비티 id 값 읽어오는 함수
function	savedata	String id	Id 값을 데이터베이스에 저장하는 함수
function	insertdata	String url, String vid	url 설정 후 서버에 데이터 전송
function	create_thread	-	실시간 통신을 위한 스레드 생성
function	postexcute	String s	S 문자열 통신
function	encryption	Char string	데이터 암호화 후, 송신
function	decryption	Char string	수신 후, 데이터 복호화

백업6: 결과화면

- Provider 웹 페이지
  - 사용자 정보 전송 화면

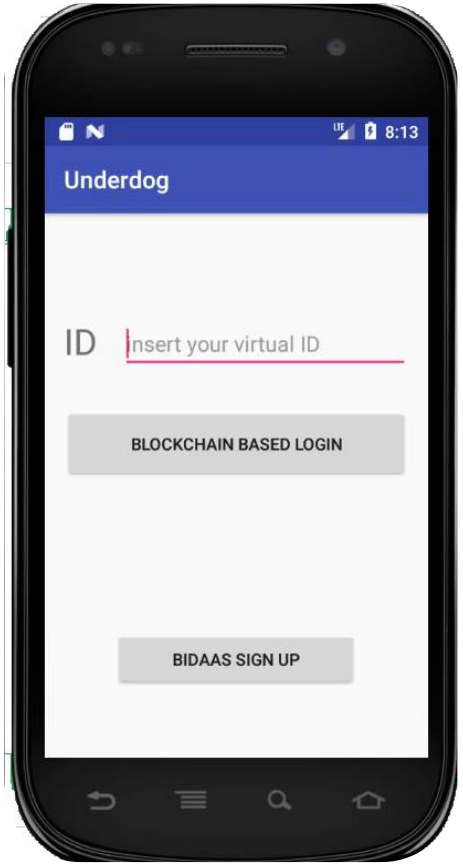


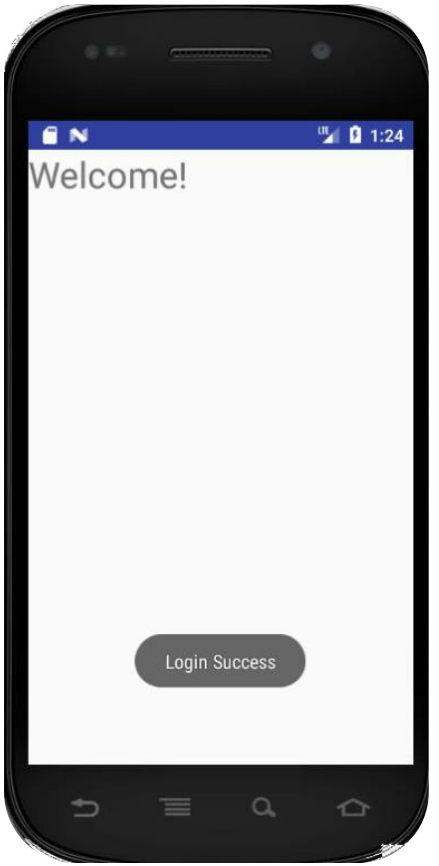
  - 사용자 정보 등록 완료 화면



백업6: 결과화면

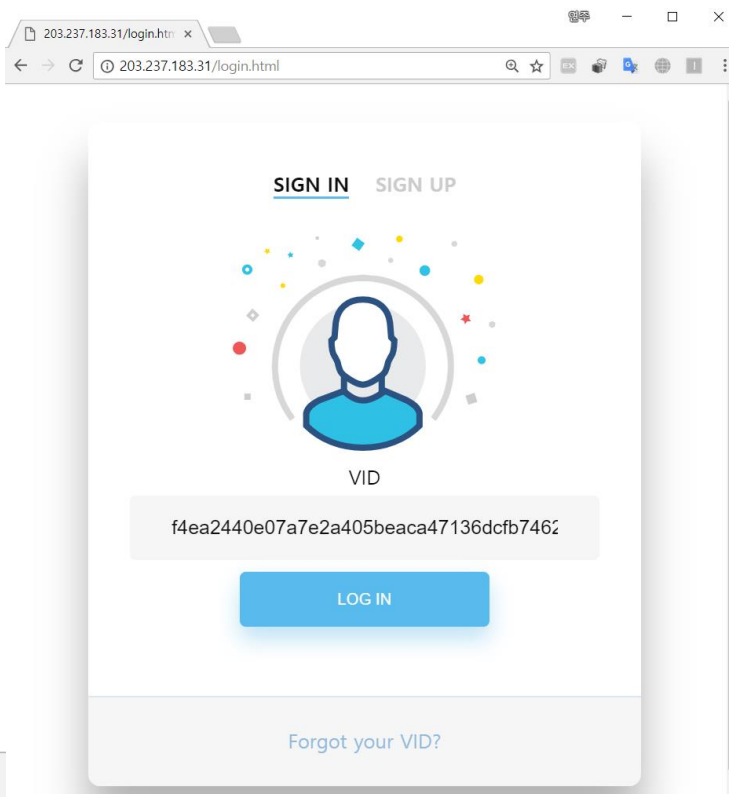
- Partner 어플리케이션
  - 로그인 성공 화면





백업6: 결과화면

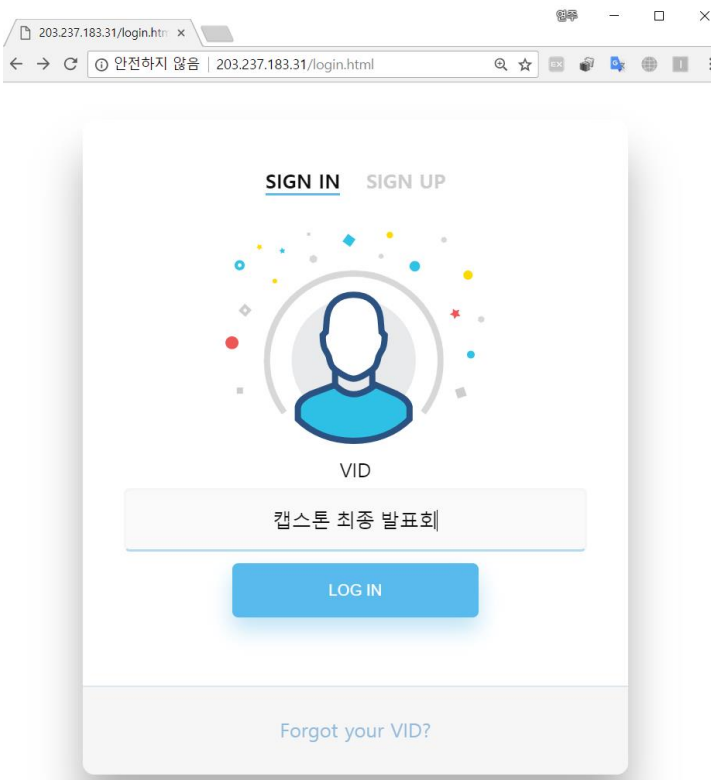
- Partner 웹 페이지(1/2)
  - Virtual ID 기반 로그인 화면



  - 상호 인증 완료 후 로그인 성공 화면

백업6: 결과화면

- Partner 웹 페이지(2/2)
  - Virtual ID 기반 로그인 화면



  - 로그인 실패 화면