

DID와 영지식 증명

Decentralized Identity & Zero Knowledge Proof

충남대학교 류재철

jcryou@home.cnu.ac.kr

2020-07-17

'모바일 신분증' 시대 온다

파이낸셜뉴스 입력 2020.01.30 12:00 수정 2020.02.02 23:52



오는 5월부터 차량공유 서비스 '쏘카(SoCar)'를 이용할 때나, 편의점에서 술·담배 등 연령제한이 있는 물품을 구매할 때 주민등록증이나 운전면허증 같은 플라스틱 신분증 대신 스마트폰 애플리케이션(앱) 기반 '모바일 운전면허증(사진)'으로 운전자격이나 연령을 증명할 수 있게 된다.

모바일 운전면허증을 이용하면 이용자의 성명이나 주소 등 다른 개인정보는 노출하지 않으면서, 운전자격 여부나 나이만 증명할 수 있어 개인정보 유출의 위험을 줄일 수 있고, 타인의 운전면허증 도용으로 인한 무면허 운전의 위험도 예방할 수 있을 것으로 기대된다.

분산아이디(DID), 사업은 경쟁하되 기술은 표준화 필요

코인데스크 코리아 2020-03-24

지금까지는 중앙 데이터베이스 기반의 아이디와 패스워드를 사용했다. 기존 법과 제도 안에서는 특정한 공인인증 기관과 신용평가사, 통신사 만이 인증 기능을 수행할 수 있다. 거기서 조금 더 나아간 게 소셜로그인이다. 그런데 소셜로그인은 말 그대로 로그인만 대체할 뿐, 본인 여부 인증까지 대체하기 어렵다. 게다가 네이버나 페이스북 등 소셜로그인 서비스 제공사가 운영을 중단하면, 인증의 연속성을 보장하기 어렵다.

반면, 블록체인 기반 분산인증 기술이 적용되면 로그인과 같은 간단한 인증뿐 아니라 본인인증, 자격증명까지 대체할 수 있게 된다. 또한 자기주권신원(SSI, self sovereign identity) 개념을 실현함과 동시에 데이터를 제공하는 소비자들에게 합당한 대가도 제공할 수 있다."



주민등록증

운전면허증

여권

대학졸업증
명

의료보험증
명

PART 1.

개요

PART 2.

DID(Decentralized IDentity)

PART 3.

영지식 증명

PART 4.

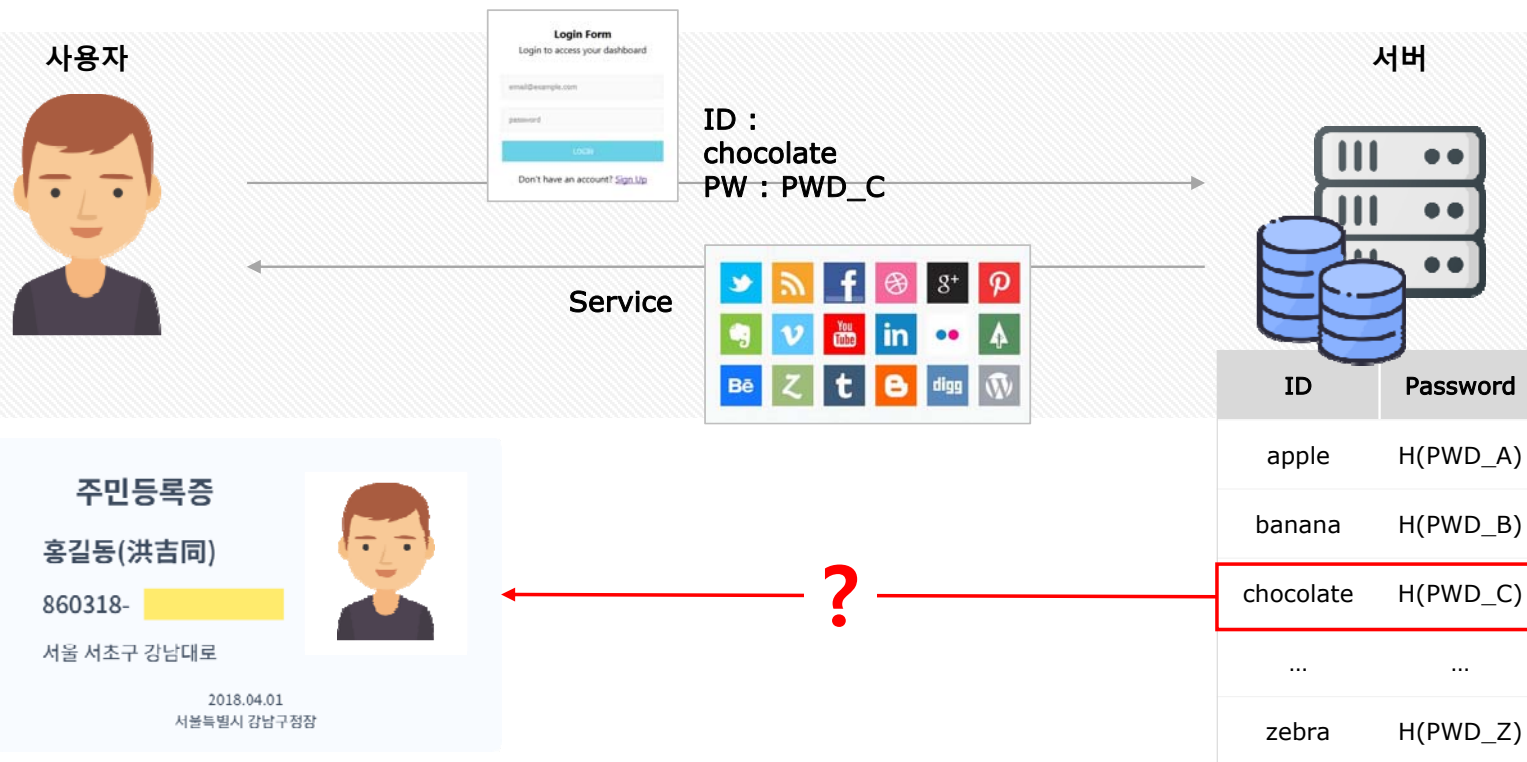
활용 사례

PART 1.

개요

PART 1. 개요

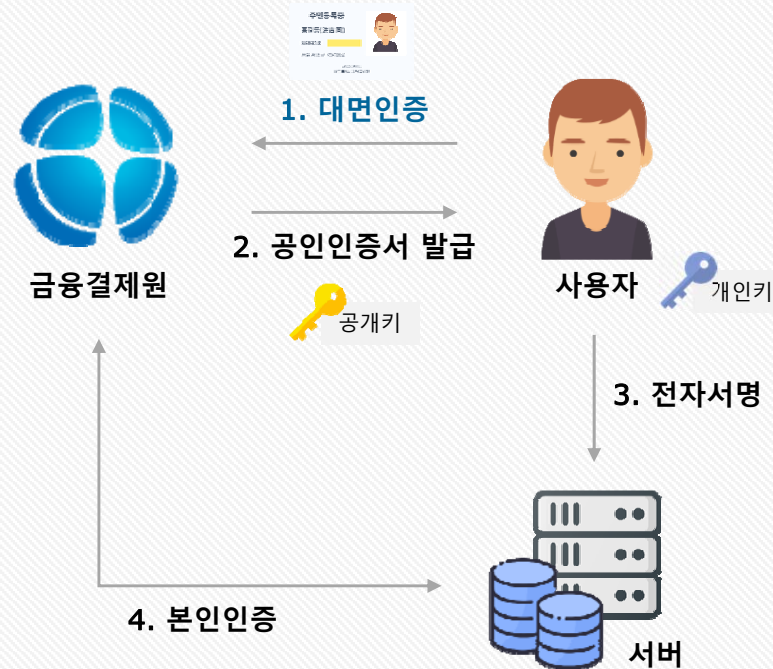
사용자 인증



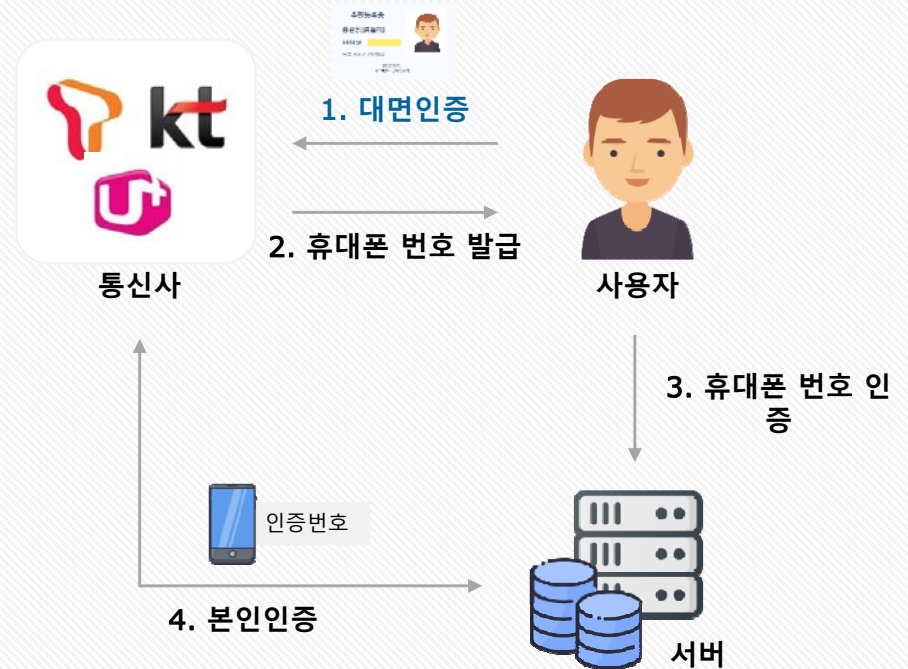
PART 1. 개요

본인 인증 (대면 인증)

공인인증서



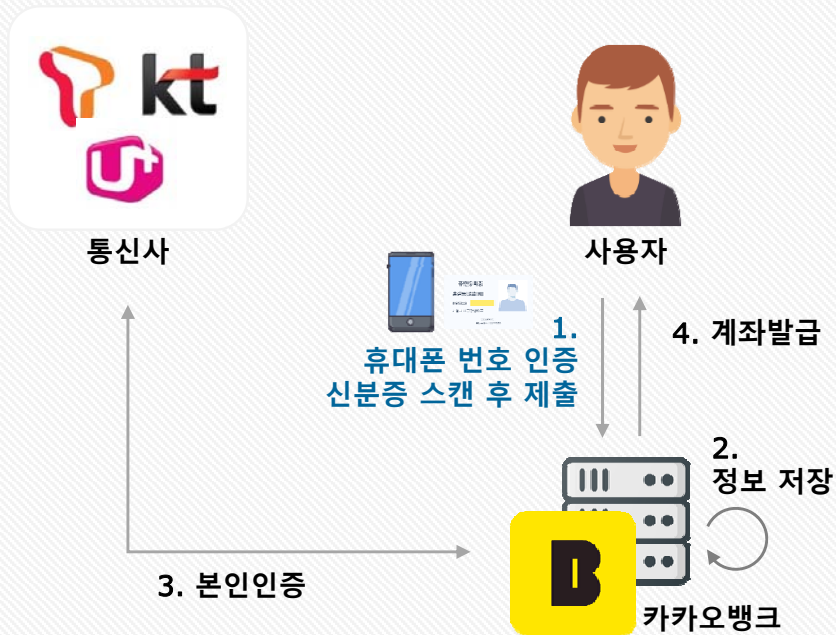
휴대폰인증



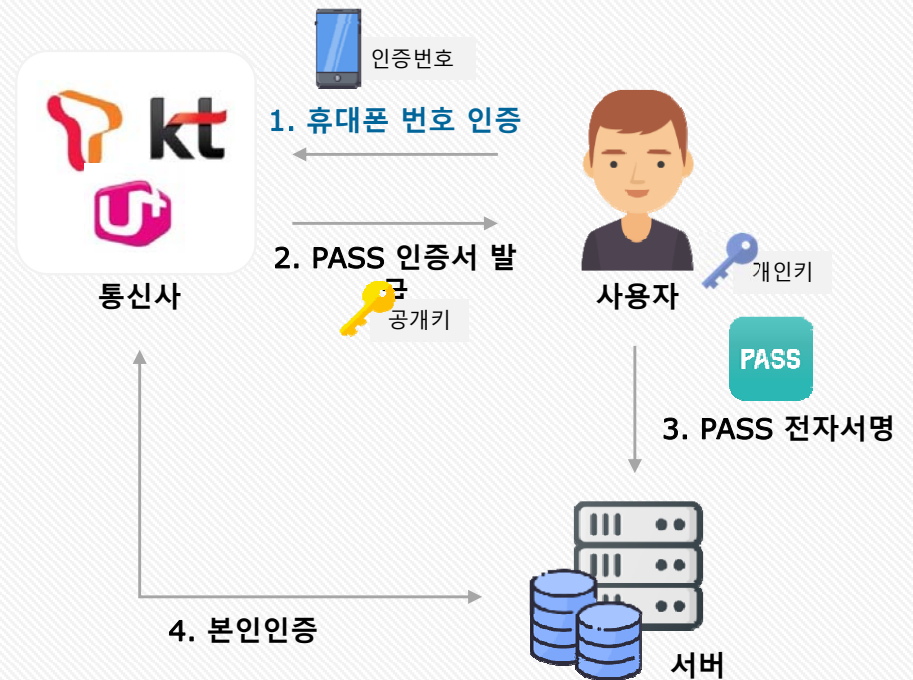
PART 1. 개요

본인 인증 (비대면 인증)

카카오뱅크



PASS 인증



PART 1. 개요

비대면 인증 수단

KakaoBank

이용자수 : 1,200만



- 2017년 7월 서비스 출시
- 비대면 인증을 통한 계좌 개설
- 100% 모바일 전용으로 운영

비대면 계좌 개설 (본인 인증)

PASS

이용자수 : 3,000만



- 2018년 8월 서비스 출시
- 통신 3사 통합 브랜드
- 통신사 가입자와 연계

민간 인증서 발급 (전자서명 + 본인인증)

TOSS

이용자수 : 1,100만



- 2018년 11월 서비스 출시
- 한국전자인증과 협력
- 수협 · 삼성화재 등 5곳 참여

PART 1. 개요

본인 인증과 신원 인증

본인
인증

전자서명

계좌 송금

증권 거래

연말 정산

전자 민원

...

인 증



공인인증서



PASS



TOSS



카카오횁크

...

신원
인증

자격증명

졸업증명서

장애증명서

국가기술자격증

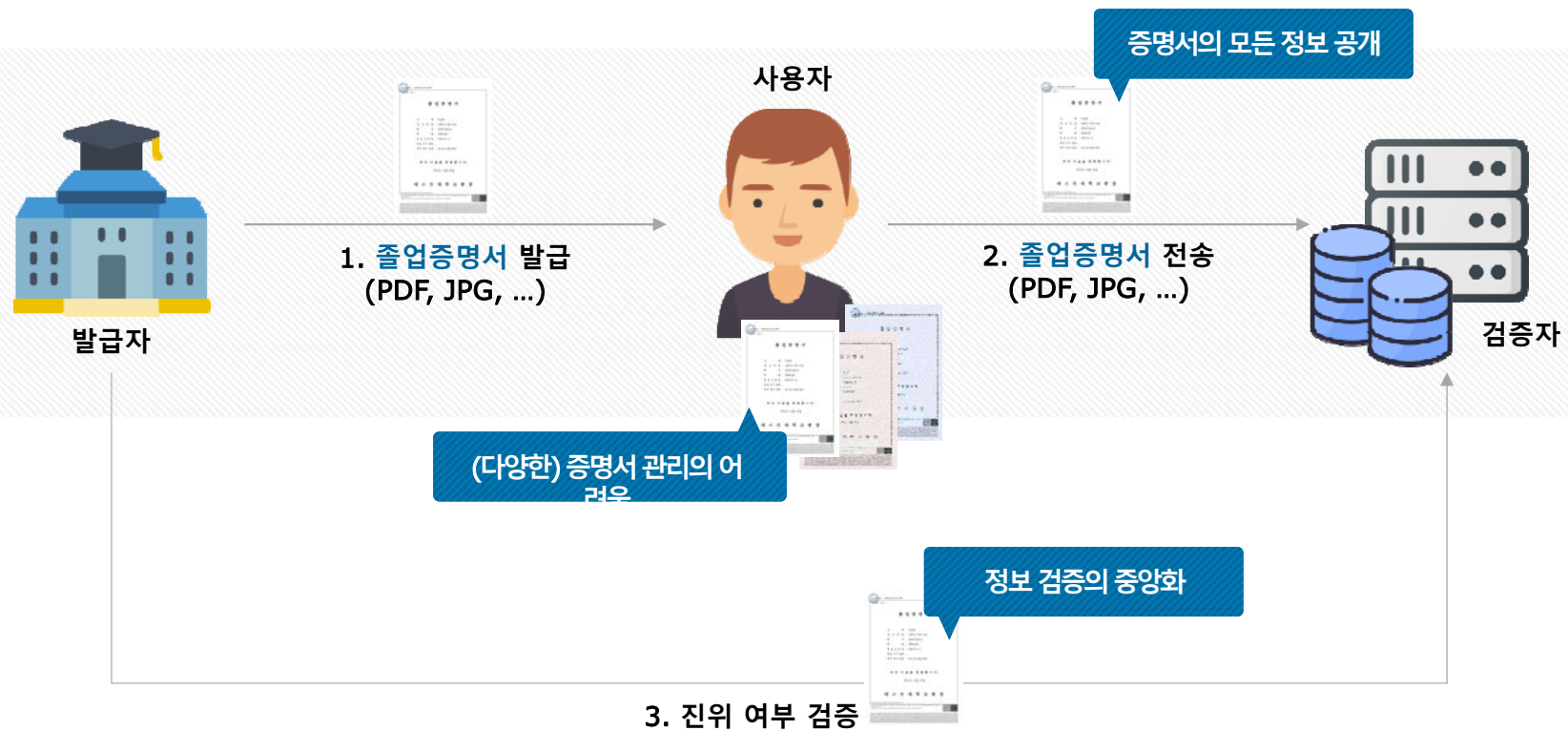
주민등록증

...

19 성인 여부

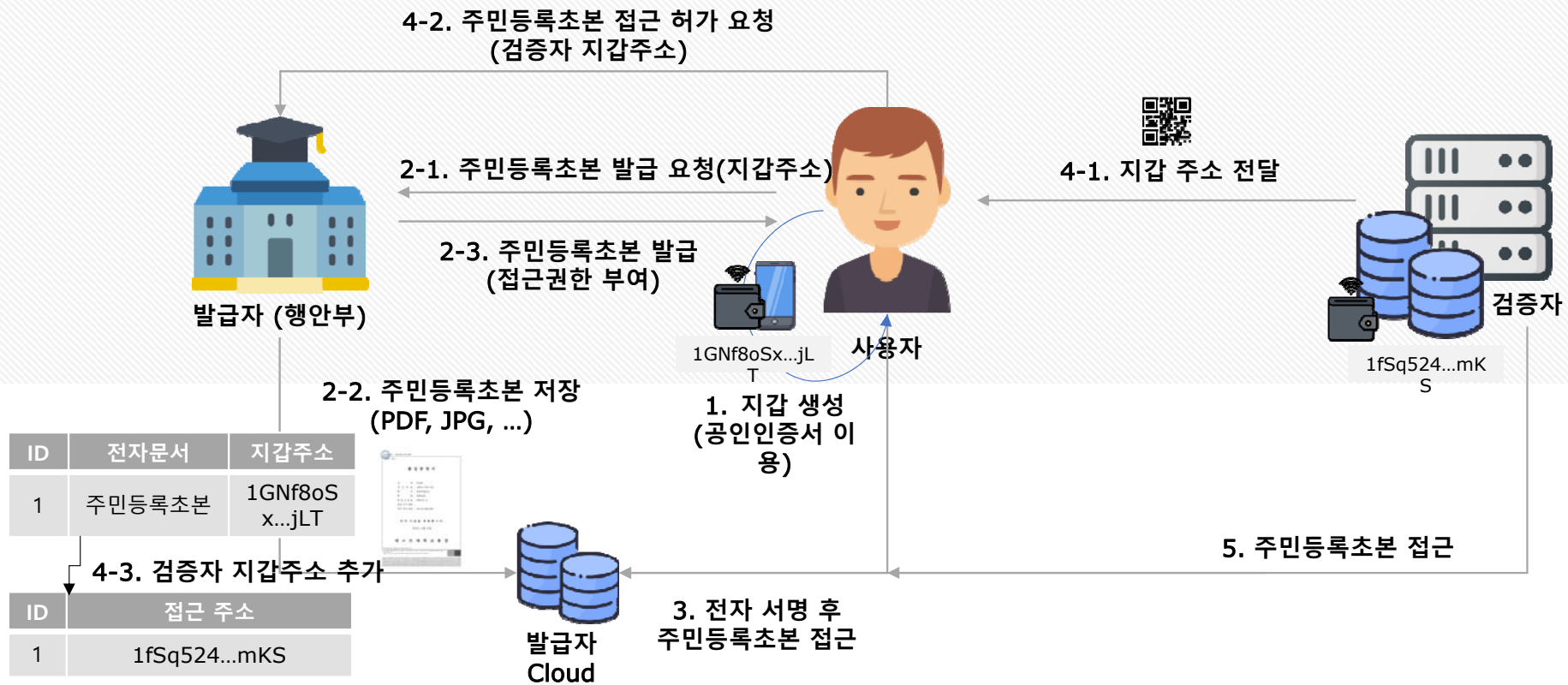
PART 1. 개요

신원 인증의 예



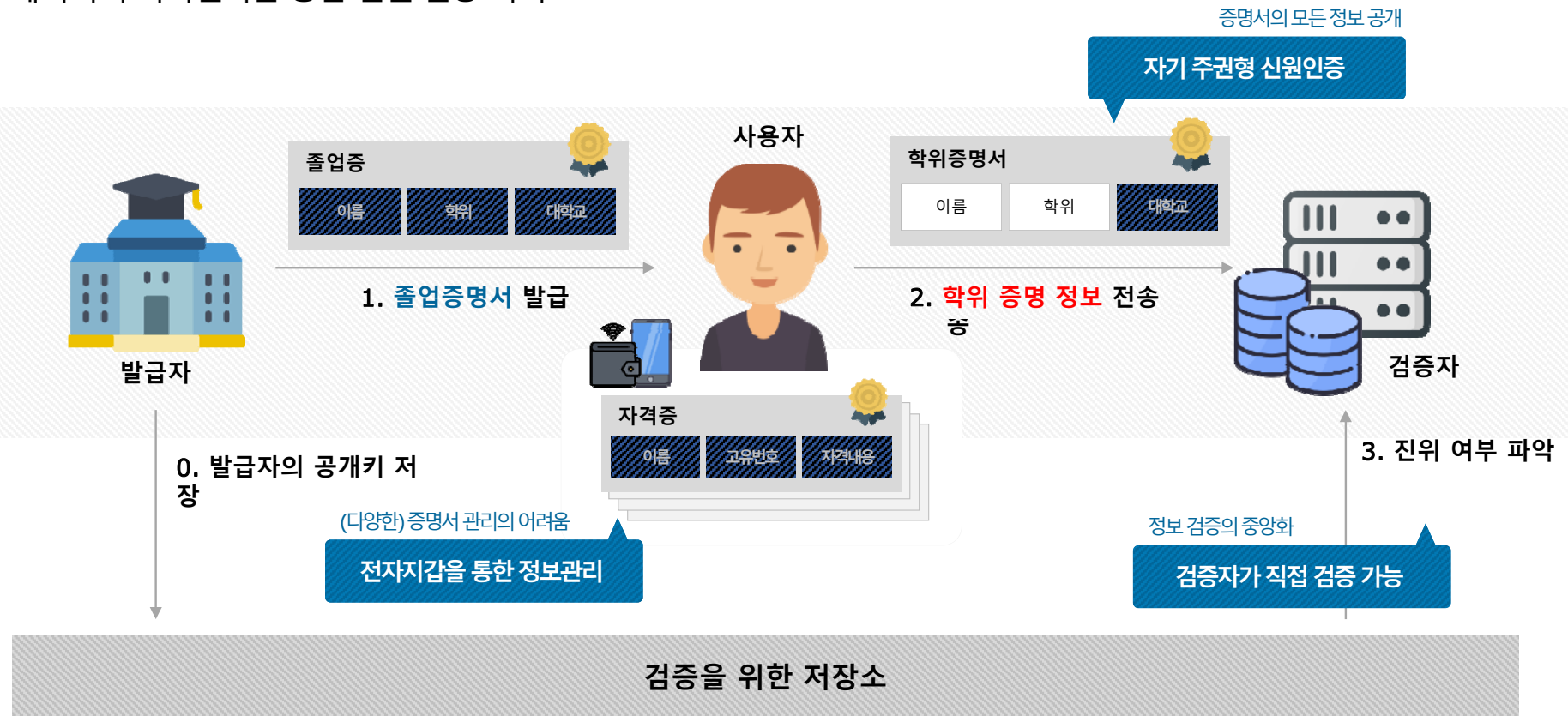
정부24의 전자문서지갑

- 모바일 기기를 이용하여 전자문서 발급 (현재 13종의 전자문서 지원) - 2020.05



DID (Decentralized Identity)

- 데이터의 디지털화를 통한 신원 인증 서비스

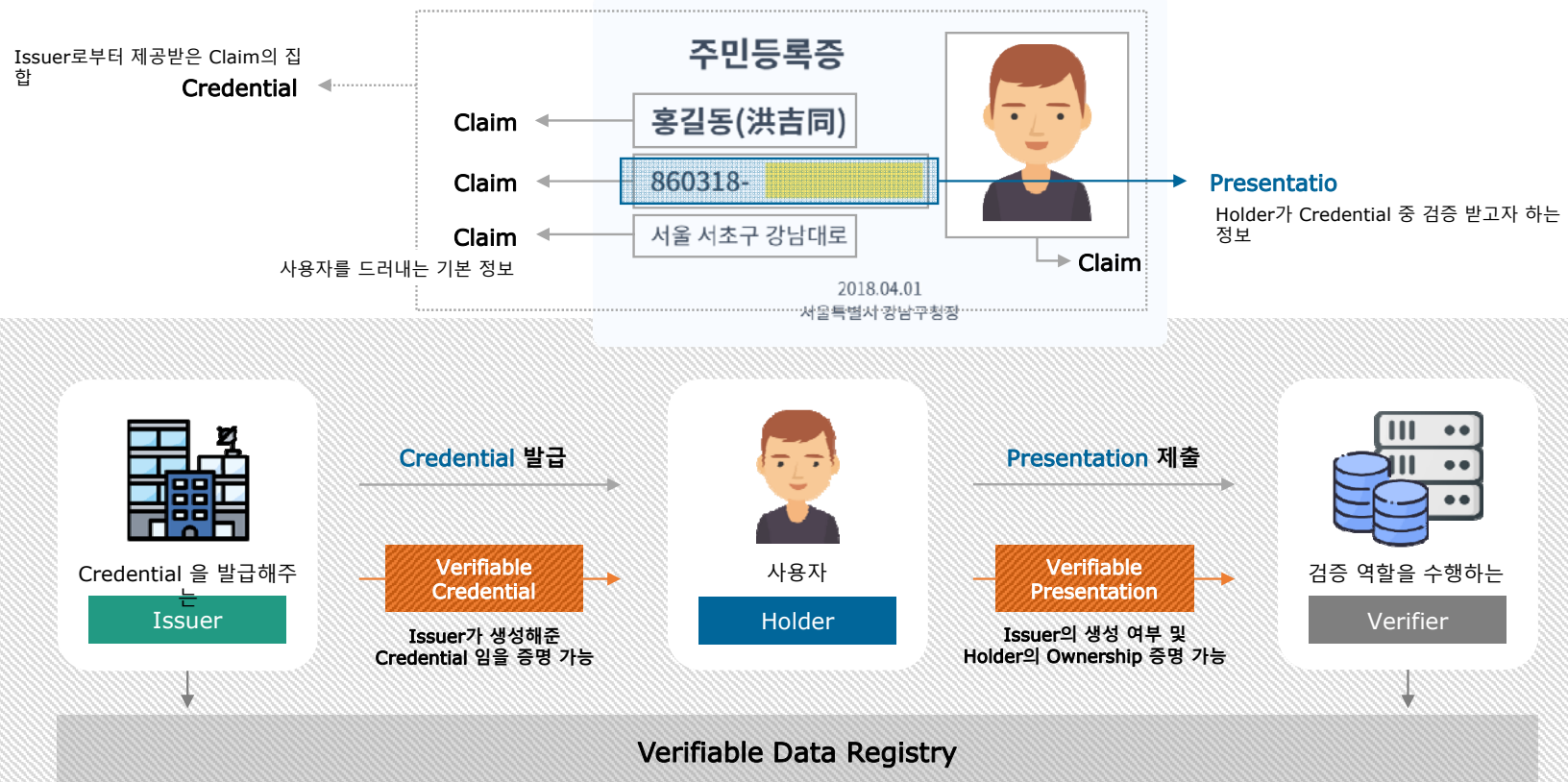


PART 2.

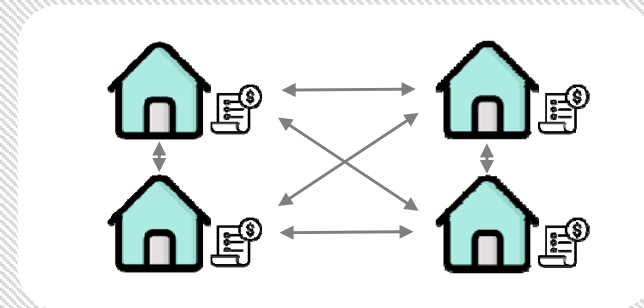
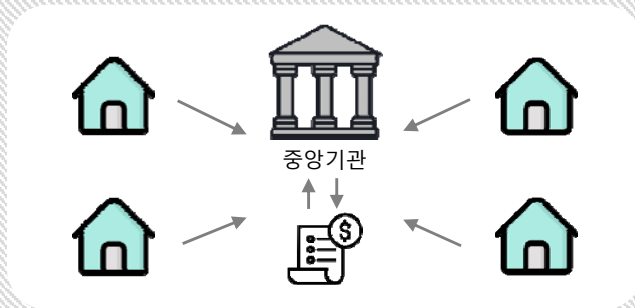
DID(Decentralized IDentity)

PART 2. DID(Decentralized IDentity)

DID 구성요소



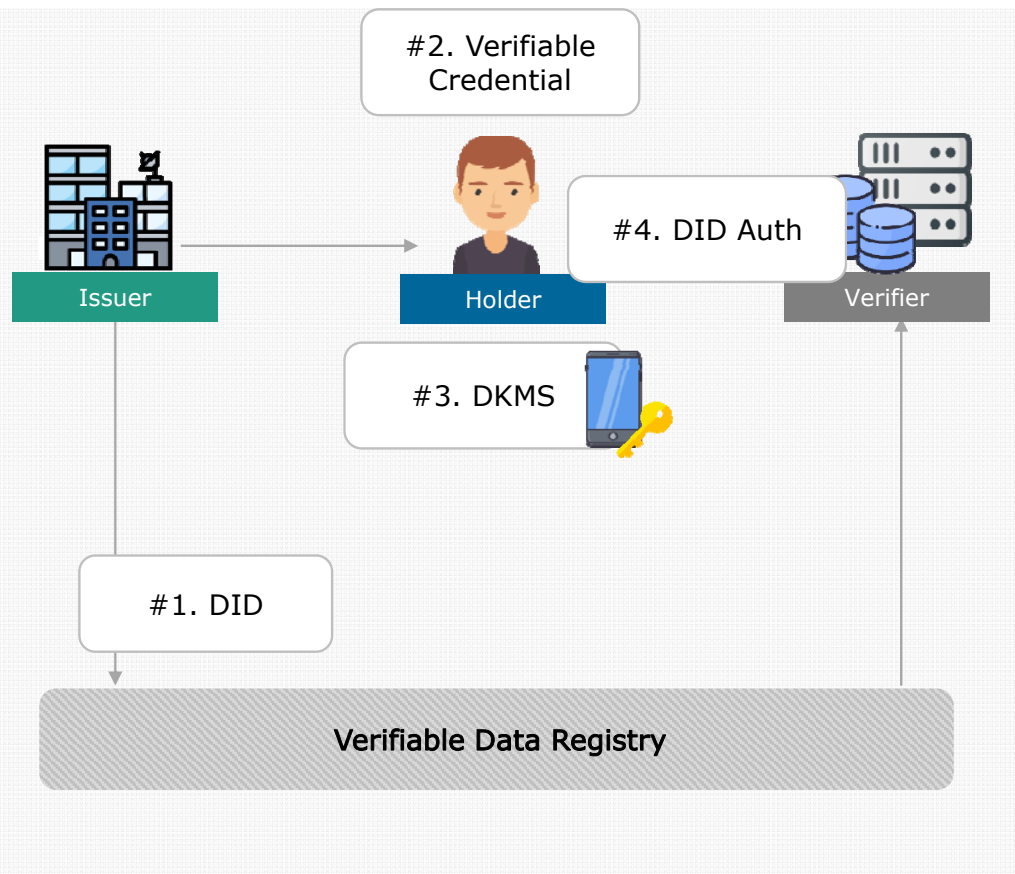
Verifiable Data Registry



(중앙집중형) 서버-클라이언트	Architecture	(Blockchain) 분산 원장
X	무결성	O
X	투명성	O
O	속도 (Read-Write)	Δ (Write는 느림)
X	보안	O

PART 2. DID(Decentralized IDentity)

DID 표준화 현황



#1. DID



World Wide Web Consortium

: DID 포맷 등 전반적인 아이디 관리 방안

#2. Verifiable Credential



: Verifiable Credential 포맷 및 전자서명의 타입 정의

#3. DKMS



: 탈중앙화 키 관리 방안 (키 복구 및 키 유도방법 등)
(Decentralized Key Management System)

#4. DID Auth



Decentralized Identify Foundation
Internet Engineering Task Force

: 웹 브라우저, 모바일기기 등 구성요소간의 인터페이스

#1. DID(Decentralized Identifier)(1)

- 분산 ID 서비스 주체에 대한 신원 식별 코드

Decentralized Identifiers (DIDs) v1.0

Core Data Model and Syntaxes



W3C Working Draft 09 December 2019

This version:

<https://www.w3.org/TR/2019/WD-did-core-20191209/>

Decentralized Identifier Resolution (DID Resolution) v0.2

Resolution of Decentralized Identifiers (DIDs)



Draft Community Group Report 09 January 2020

Latest editor's draft:

<https://w3c-ccg.github.io/did-resolution/>

DID Method Registry

A registry for Decentralized Identifier Methods



Draft Community Group Report 17 January 2020

Latest editor's draft:

<https://w3c-ccg.github.io/did-method-registry/>

Editors:

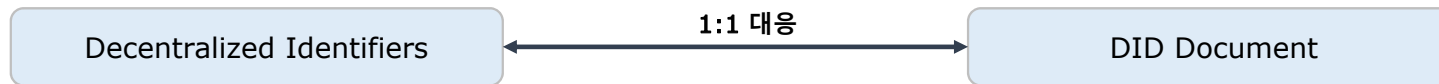
Manu Sporny (Digital Bazaar)
Drummond Reed (Evernym)

DID의 핵심 규격으로 아이디의 형식, DID의 데이터 모델 정의

탈중앙화 아이디의 메타 정보가 담겨 있는 DID 문서를 처리하기 위한 규격 정의

DID 메소드 이름의 중복을 방지하기 위해 등록된 DID method를 정리한 문서

#1. DID(Decentralized Identifier)(2)



`did:example:123456789abcdefghi`



Scheme DID Method DID method Specific String

Method Name	Status	DLT or Network
did:abt:	PROVISIONAL	ABT Network
did:btc:	PROVISIONAL	Bitcoin
did:stack:	PROVISIONAL	Bitcoin
did:erc725:	PROVISIONAL	Ethereum
did:example:	PROVISIONAL	DID Specification
did:ipid:	PROVISIONAL	IPFS
did:life:	PROVISIONAL	RChain

W3C

```
{
  "@context": "https://www.w3.org/ns/did/v1",
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    // used to authenticate as did:...fghi
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "RsaVerificationKey2018",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----"
  }],
  "service": [{
    // used to retrieve Verifiable Credentials associated with the DID
    "id": "did:example:123456789abcdefghi#vcs",
    "type": "UniversityDegreeCredential"
  }],
  "proof": {
    "type": "RsaSignature2018",
    "created": "2018-06-17T10:03:48Z",
    "jws": "eyJ.....BBPM" ...
  }
}
```

문서 요소

DID

Public Key(인증 및 전자
서명)

연계 서비스 정보(URL 등)

문서 무결성 검증 정보

#2. Verifiable Credentials(1)

Verifiable Credentials Data Model 1.0

Expressing verifiable information on the Web



W3C Recommendation 19 November 2019

This version:

<https://www.w3.org/TR/2019/REC-vc-data-model-20191119/>

Verifiable Credentials Implementation Guidelines 1.0

Implementation guidance for Verifiable Credentials



W3C Editor's Draft 23 September 2019

This version:

<https://w3c.github.io/vc-imp-guide/>

Verifiable Credentials Use Cases

W3C Working Group Note 24 September 2019



This version:

<https://www.w3.org/TR/2019/NOTE-vc-use-cases-20190924/>

Latest published version:

<https://www.w3.org/TR/vc-use-cases/>

Latest editor's draft:

<https://w3c.github.io/vc-use-cases/>

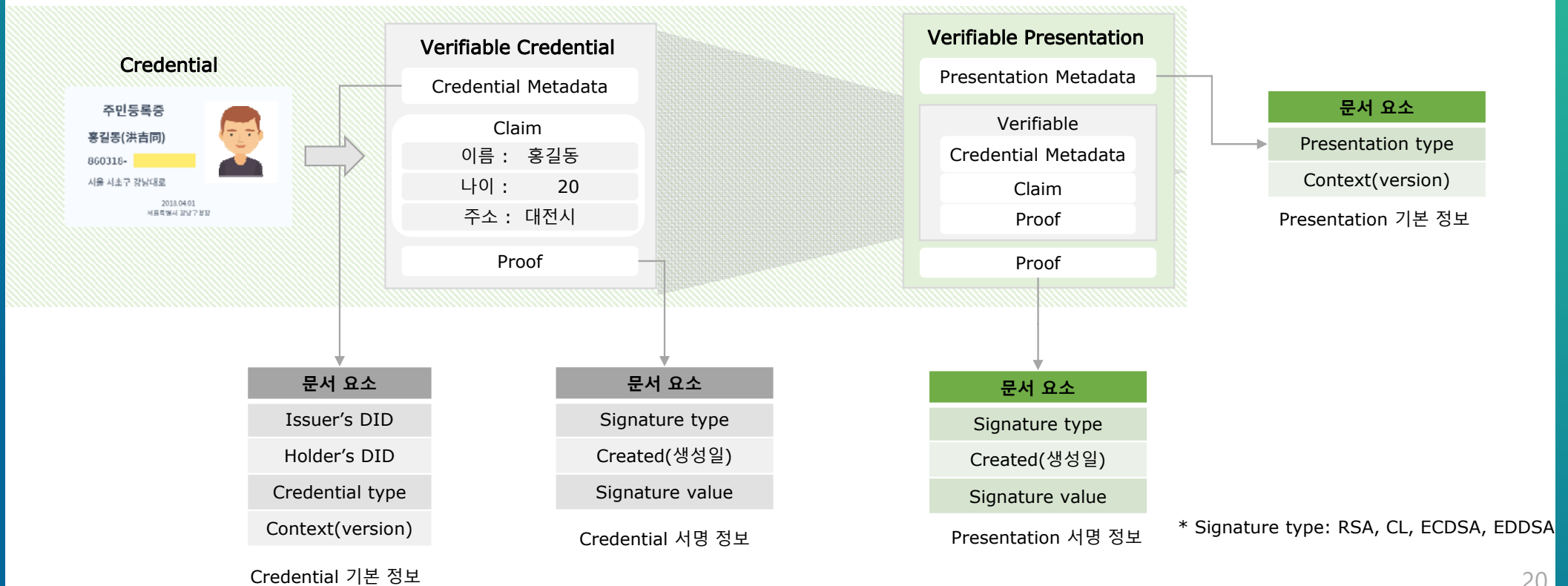
Verifiable Credential의 기본 개념, 구성 요소 및 요구사항 정의

Verifiable Credential의 구현 가이드 제시

Verifiable Credential을 활용하여 나타낼 수 있는 시나리오 제시

#2. Verifiable Credentials(2)

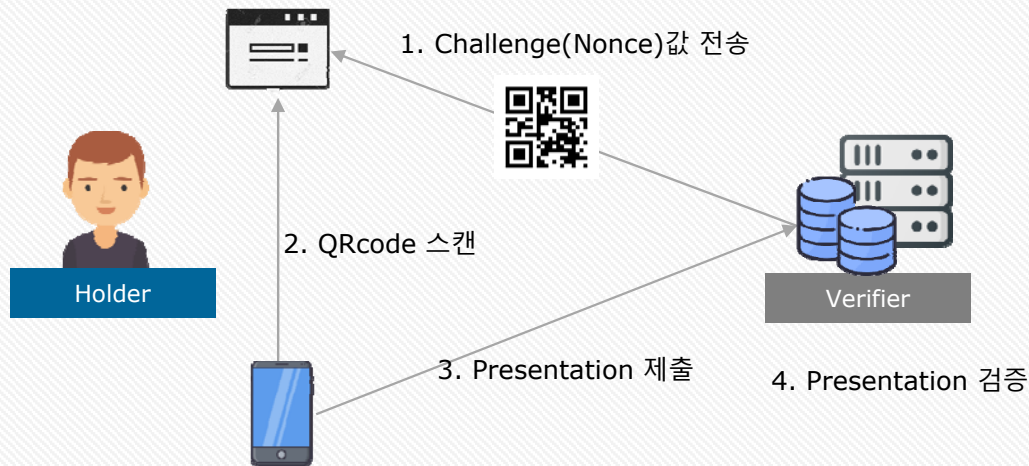
- Verifiable Credential의 형태
 - Holder는 발급받은 Verifiable Credential로 부터 Verifiable Presentation을 생성하여 전달



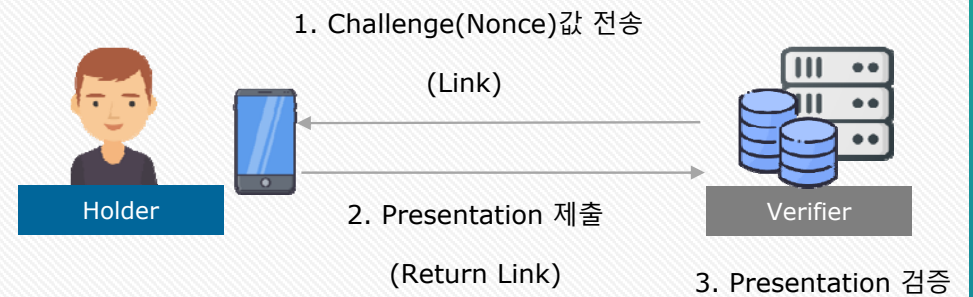
#3. DID Auth

- Holder와 Verifier간의 인터페이스 제시
 - End Point 종류별 시나리오 제시 (Mobile, Web browser) 등

(모바일 & Web browser) || (Web sever)



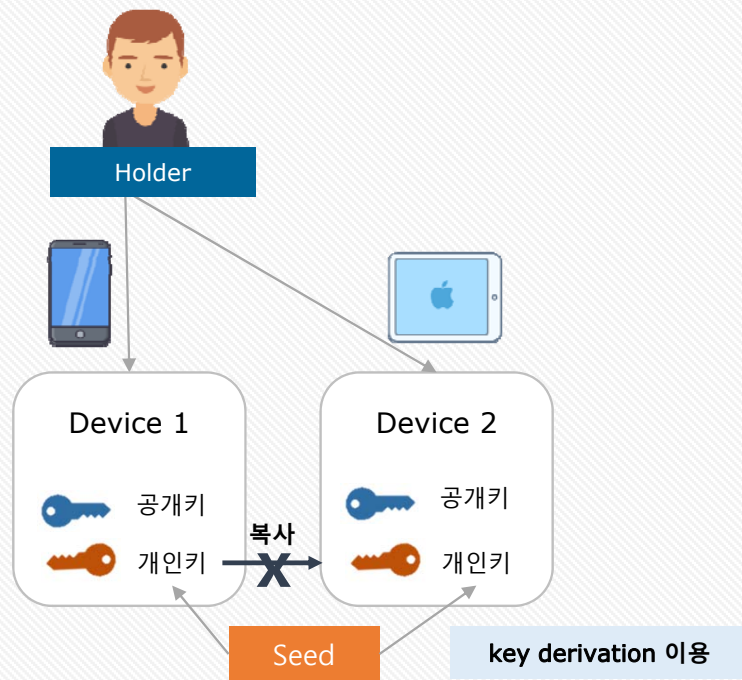
(모바일) || (Web sever)



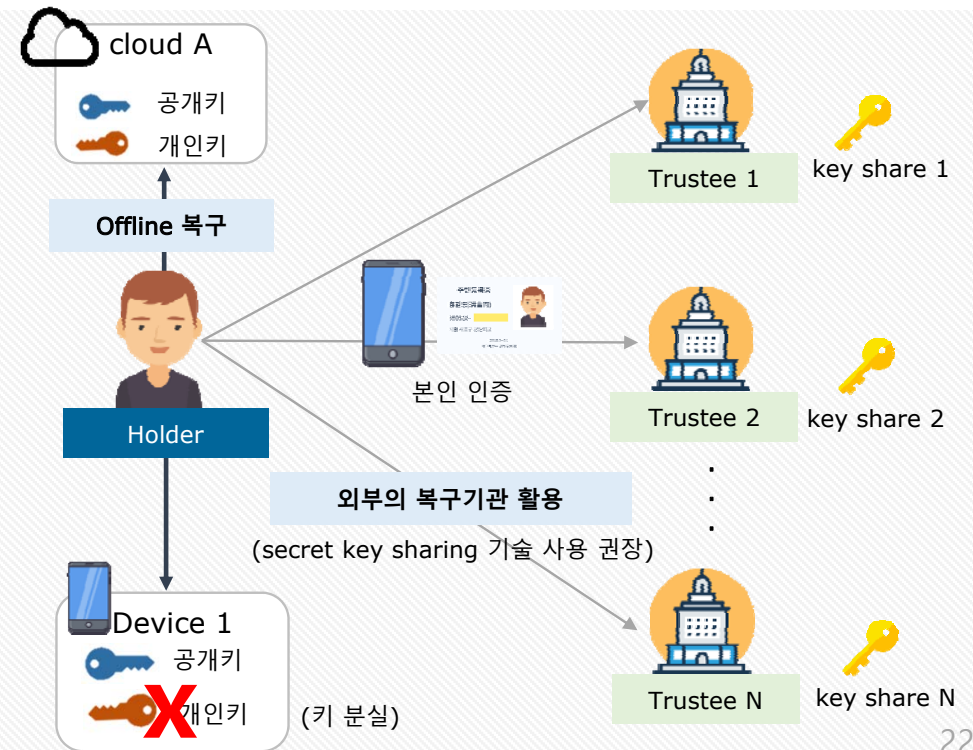
#4. DKMS

- 키 관리 방안 제시

Device 추가시



Key 복구시



Credential 유형 분류

		TYPE 1	TYPE 2	TYPE 3
형태		Credential	Credential ... Credential	Credential 
발급	속성 (Claim)	N	N	N
	인증서 (Credential)	1	N	1
Presentation 생성 방법		Credential 변경없이 Presentation에 사용 (자기통제 불가능)	Credential 변경없이 Presentation에 사용 (자기통제 불가능)	영지식증명을 사용하여 Presentation에 사용 (자기통제 가능)
영지식증명 제공 여부		X	X	O
선택적 노출		불가능	Credential Level로 가능 (발급기관 의존형)	Claim Level로 가능 (발급기관 독립적)
사용자 연결성		사용자의 서명이 포함된 Presentation	사용자의 서명이 포함된 Presentation	사용자의 서명이 아닌 일회용 Presentation (Anonymous Credential)
암호 알고리즘		RSA Signature	RSA Signature	CL Signature (Camenisch-Lysyanskaya)

Verifiable Credential

Credential Metadata

Claim

이름 : 홍길동

나이 : 20

주소 : 대전시

Proof

TYPE 1,2 Credential

Verifiable Credential

Credential Metadata

Claim

이름 : 

나이 : 

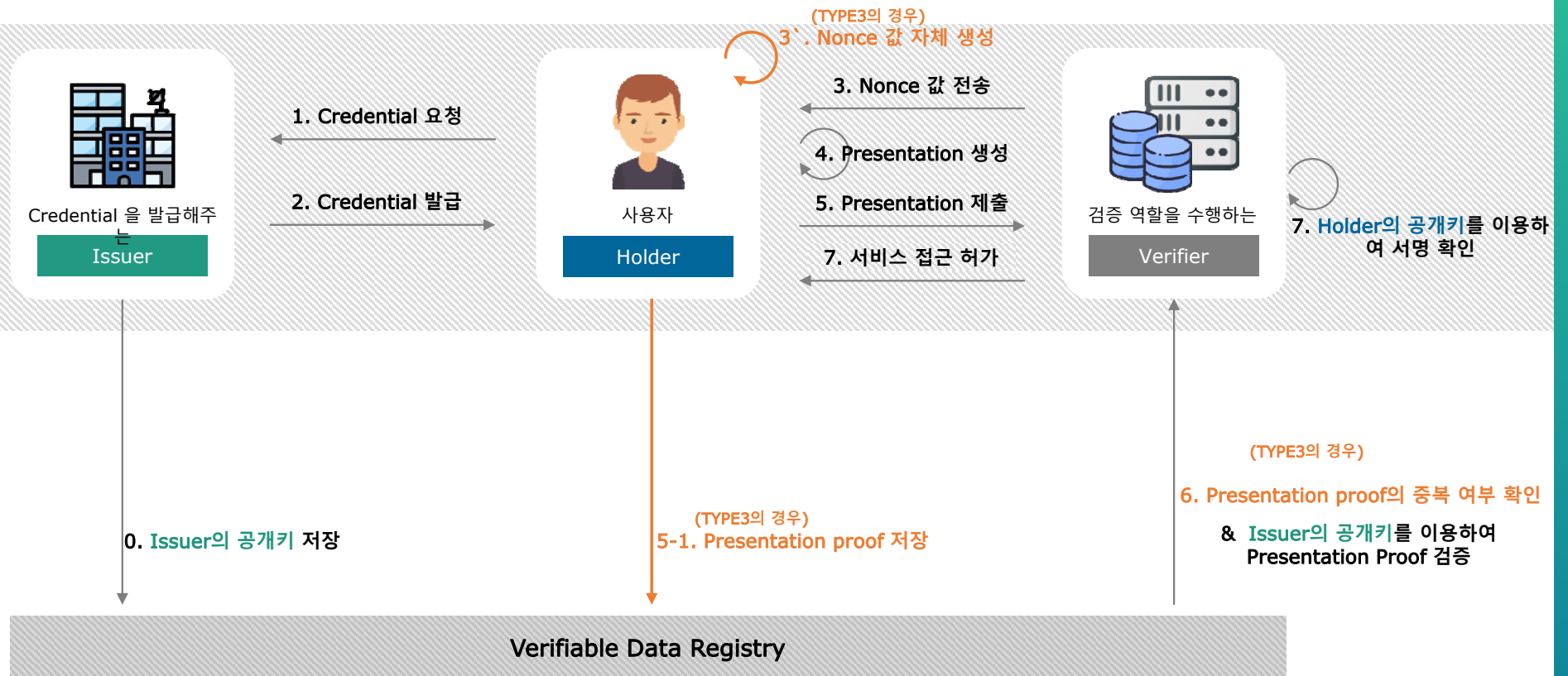
주소 : 

Proof

TYPE 3 Credential

PART 2. DID(Decentralized IDentity)

DID 동작 과정


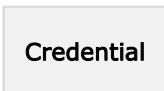

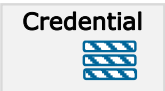


PART 2. DID(Decentralized IDentity)

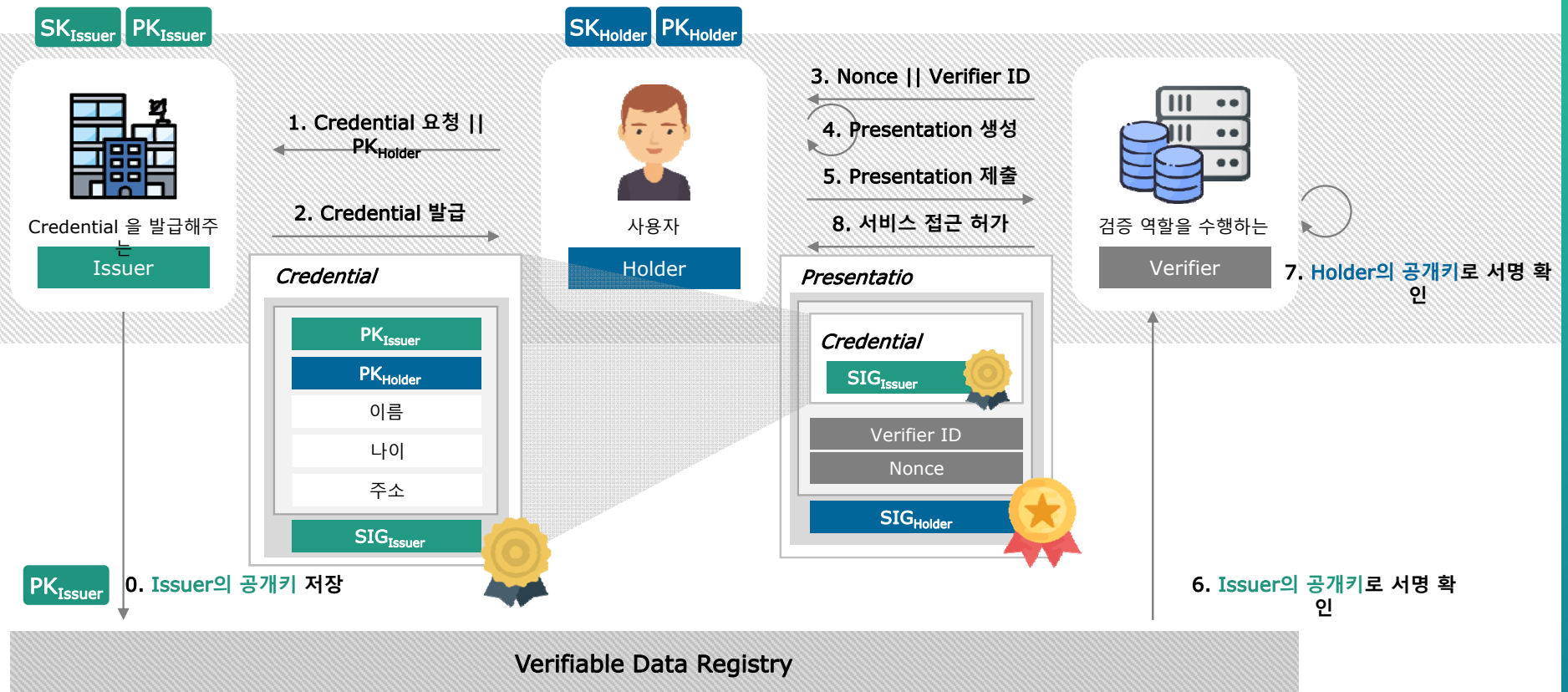
DID 동작 과정



TYPE별 DID 의 문제 해결

	TYPE 1	TYPE 2	TYPE 3
형태		 ... 	
1. Issuer과 생성해준 인증서 여부	O Issuer의 RSA 서명	O Issuer의 RSA 서명	O Issuer의 CL 서명
2. 인증서가 Holder의 소 유 여부	O Holder의 RSA 서명 개인키 관리 필요	O Holder의 RSA 서명 개인키 관리 필요	O Holder의 CL 서명 Credential 관리 필요
3. Verifier의 인증서 남용 불가능 여부	O 남용 불가능 Verifier 및 Nonce 명시	O 남용 불가능 Verifier 및 Nonce 명시	O 남용 불가능 Presentation proof를 Registry에 공개
4. Claim의 최소화 여부	X Issuer가 생성해준 원본 그대로 제출	X Issuer가 생성해준 원본 그대로 제출	O Holder가 Claim별 컨트롤 가능

Credential 사용 예 : TYPE 1, TYPE 2 (RSA)



PART 3.

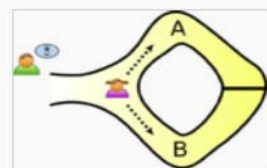
영지식 증명

PART 3. 영지식 증명

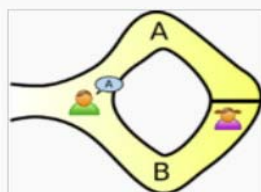
영지식 증명

- 상대방에게 어떠한 정보도 제공하지 않은 채, 해당 정보를 가지고 있음을 증명

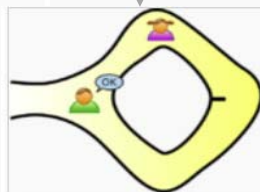
동굴 예제



Peggy randomly takes either path A or B, while Victor waits outside



Victor chooses an exit path



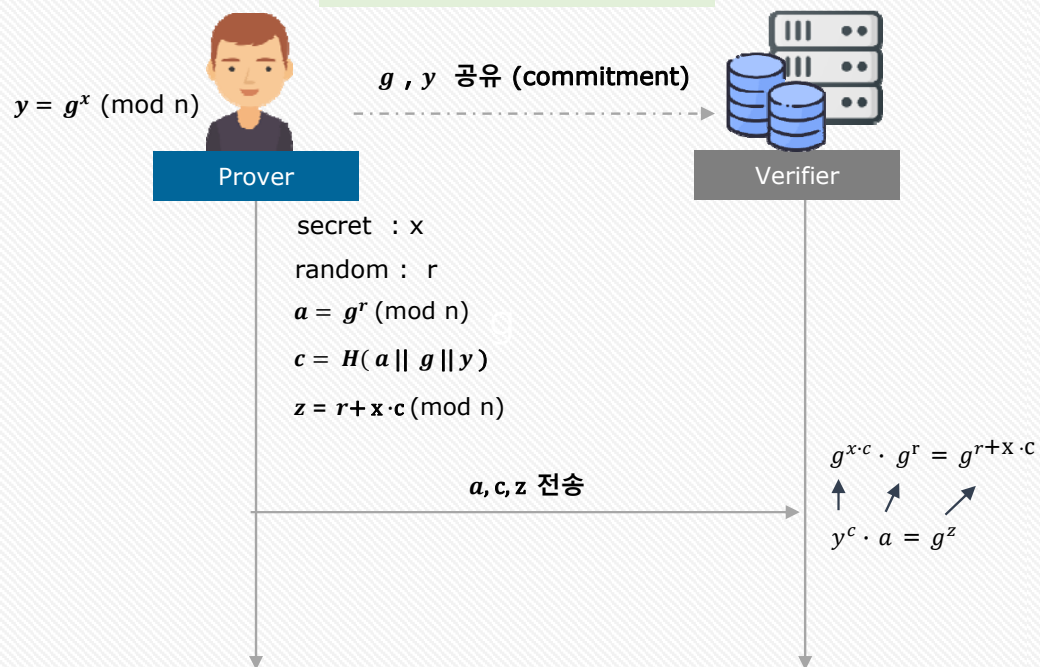
Peggy reliably appears at the exit Victor names

비밀에 대한 증명 요청 N 번 반복 수행 →
 $1 - (\frac{1}{2})^N$ 의 확률로 비밀을 알고 있음을 확신

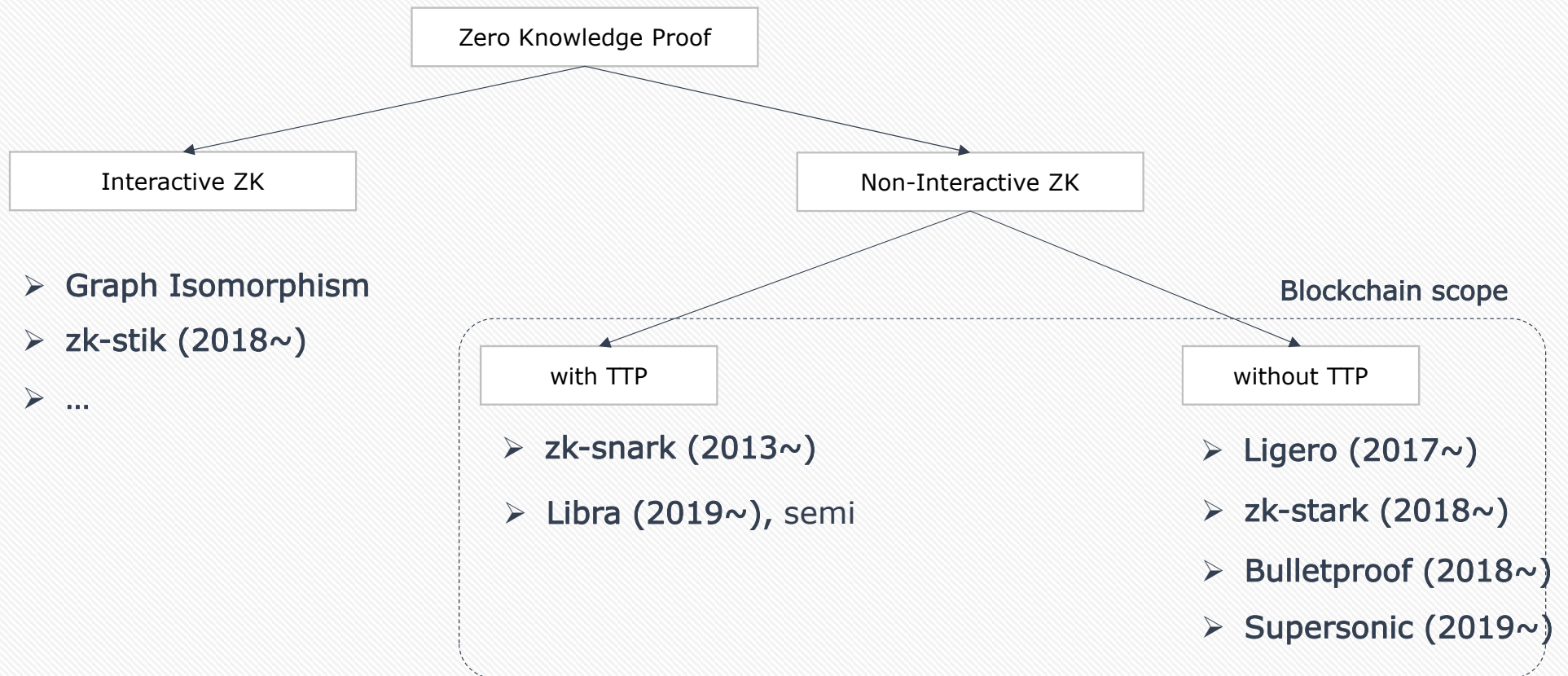


Sigma protocol – 이산로그 문제

Non – Interactive



영지식 증명 분류



PART 3. 영지식 증명

ZKP Standard

snark(Succinct Non interactive ARgument of Knowledge)
stark(Scalable Transparent ARgument of Knowledge)

- ZKProof.org NIST 표준과 영지식 증명의 발전

1st ZKProof Standards Workshop 2018, 5/10-11



2nd ZKProof Standards Workshop 2019, 4/10-12



3rd ZKProof Standards Workshop 2020, 4/20-5/21(월,목)



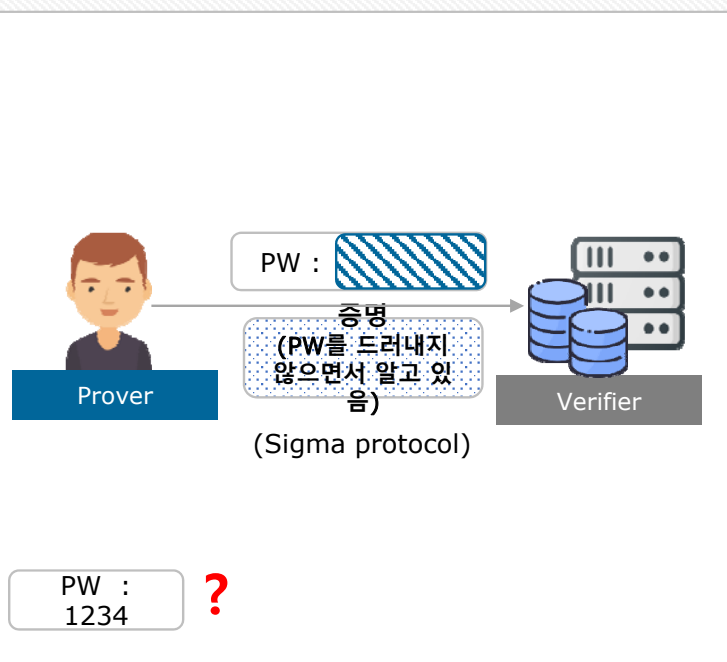
기업 : MS, IBM, ING, QED-it, Deloitte,
Zcash, Hyperledger, R3, Consensus 등
학교 : MIT, Technion, UIUC, Yale, Stanford Boston, 등

Challenges

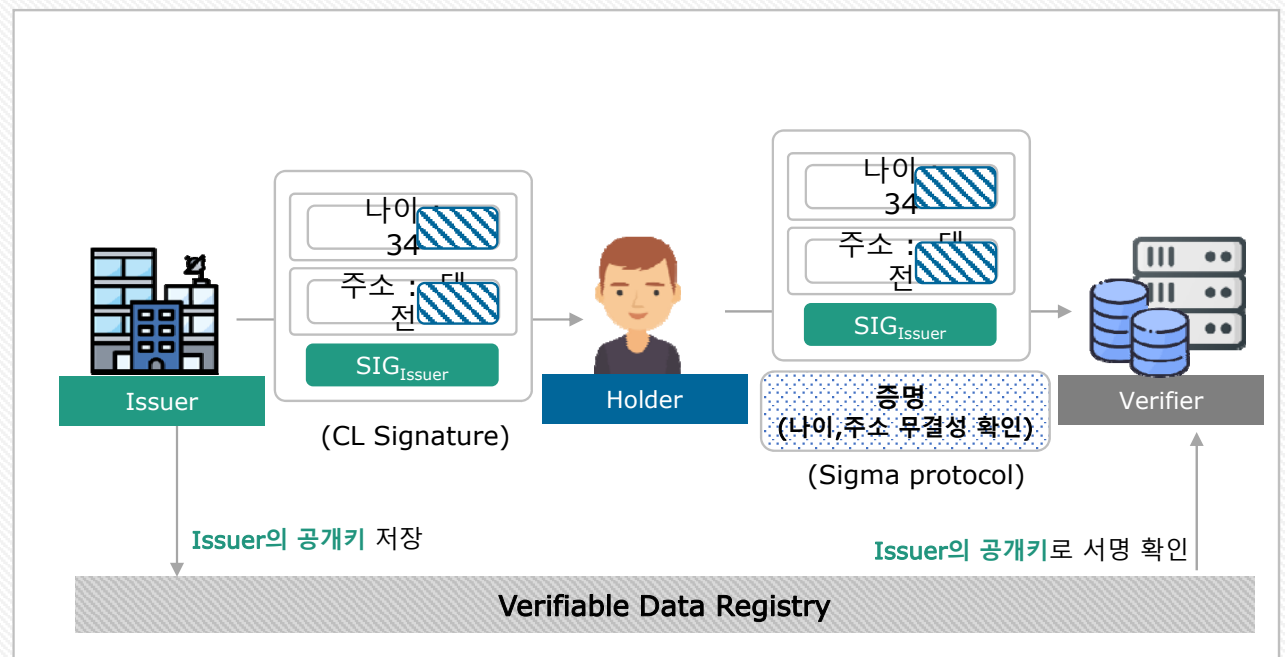
- Parameter setup
- Performance
- Verifying correctness of algorithms and implementations
- New (and correct!) application

DID에서 사용하는 영지식 증명

기존 영지식 증명

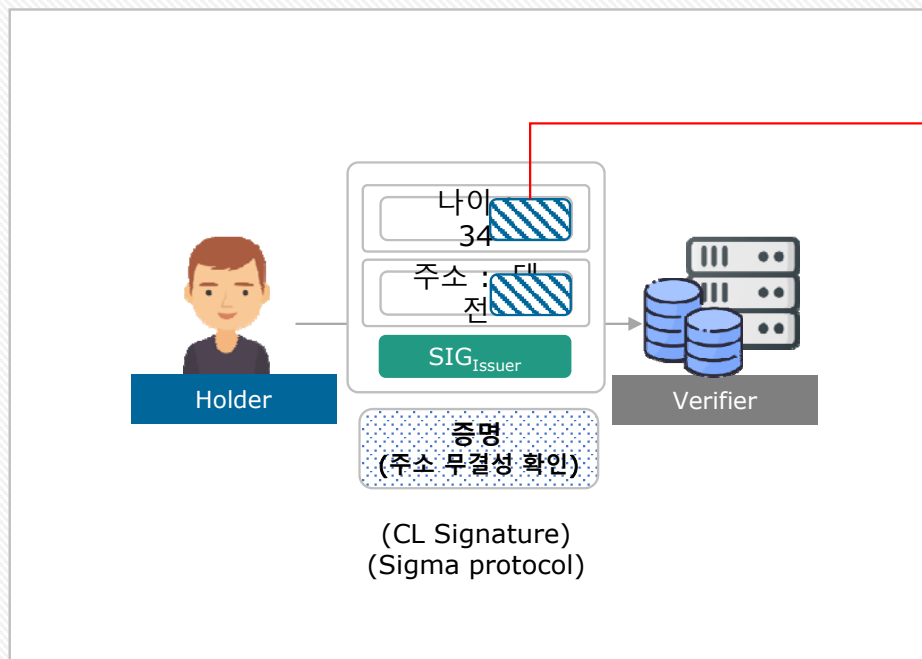


DID 영지식 증명 + 서명

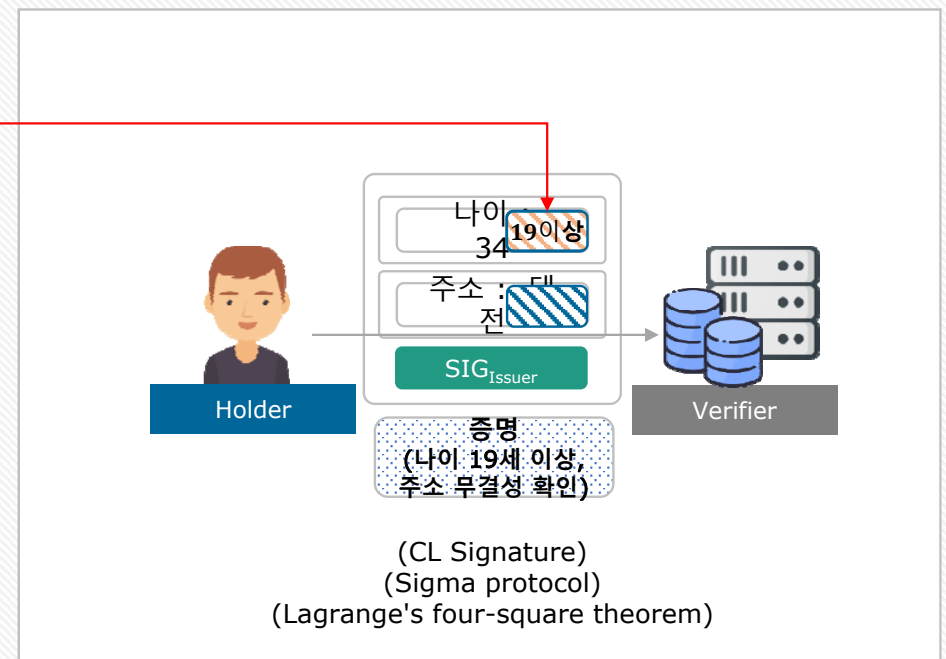


DID에서 사용하는 영지식 증명

Selective disclosure

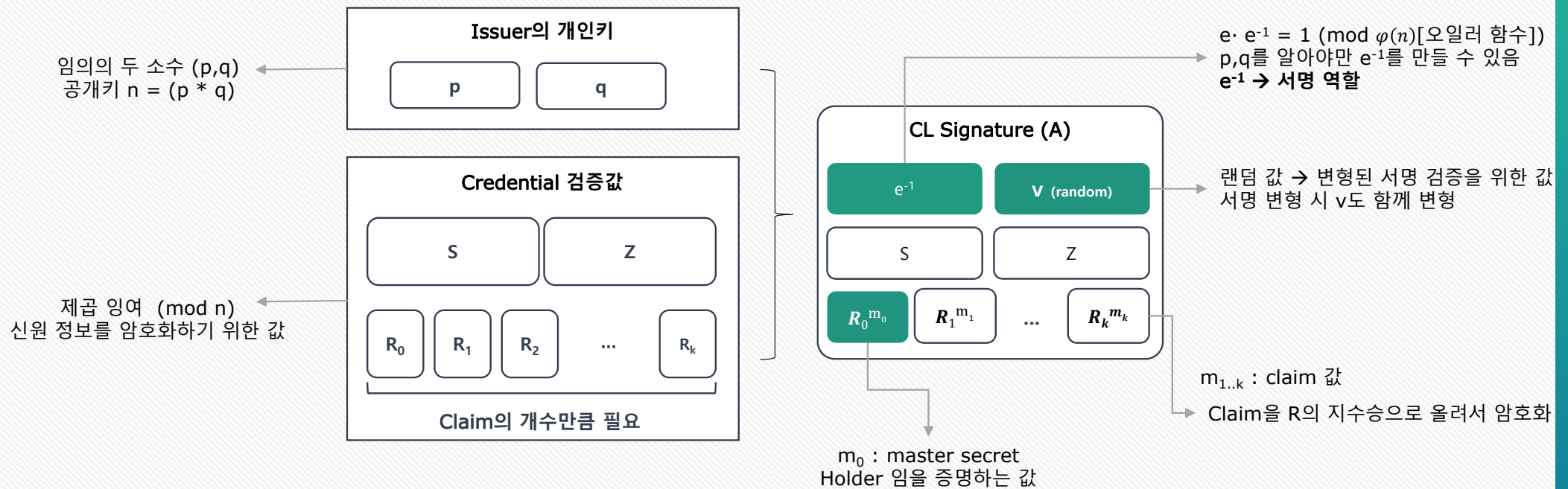


Selective disclosure + range proof



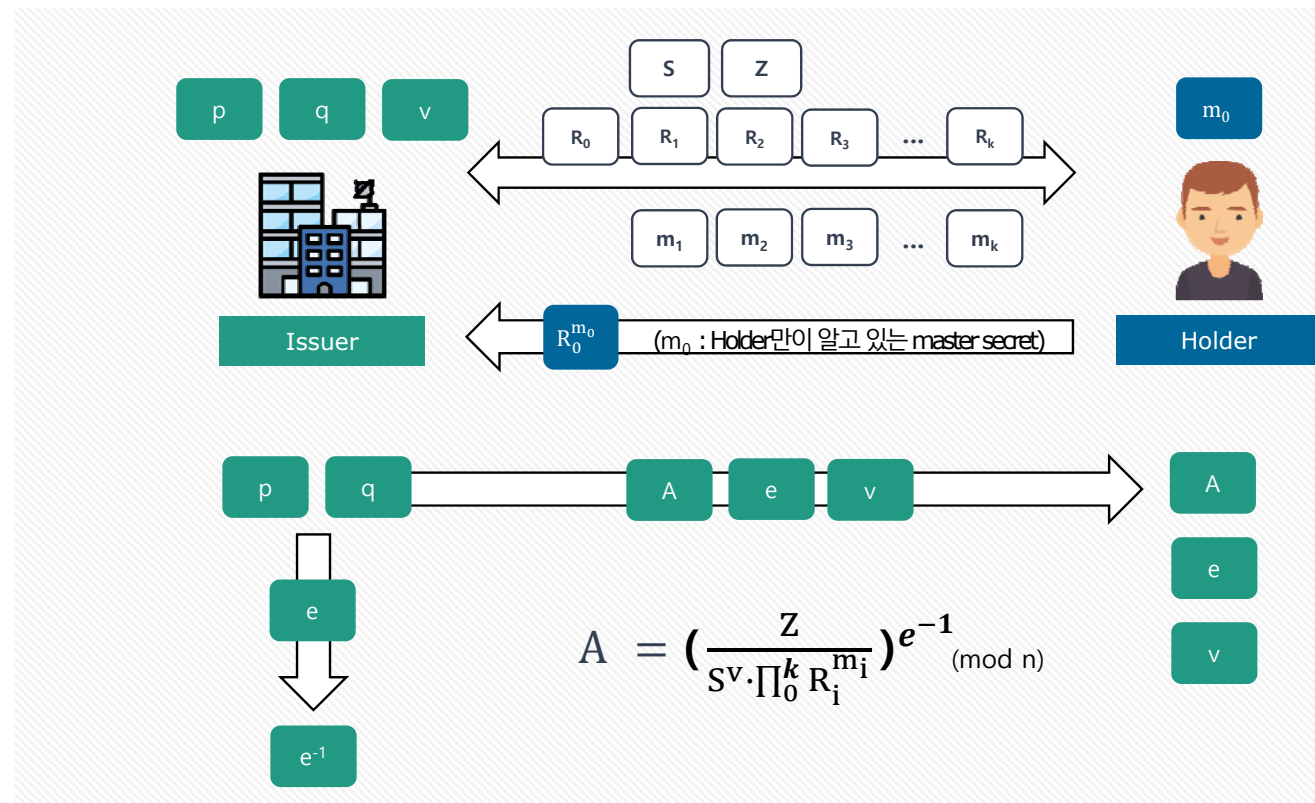
Selective disclosure

- CL(Camenisch-Lysyanskaya) Signature(1) - 구성요소



Selective disclosure

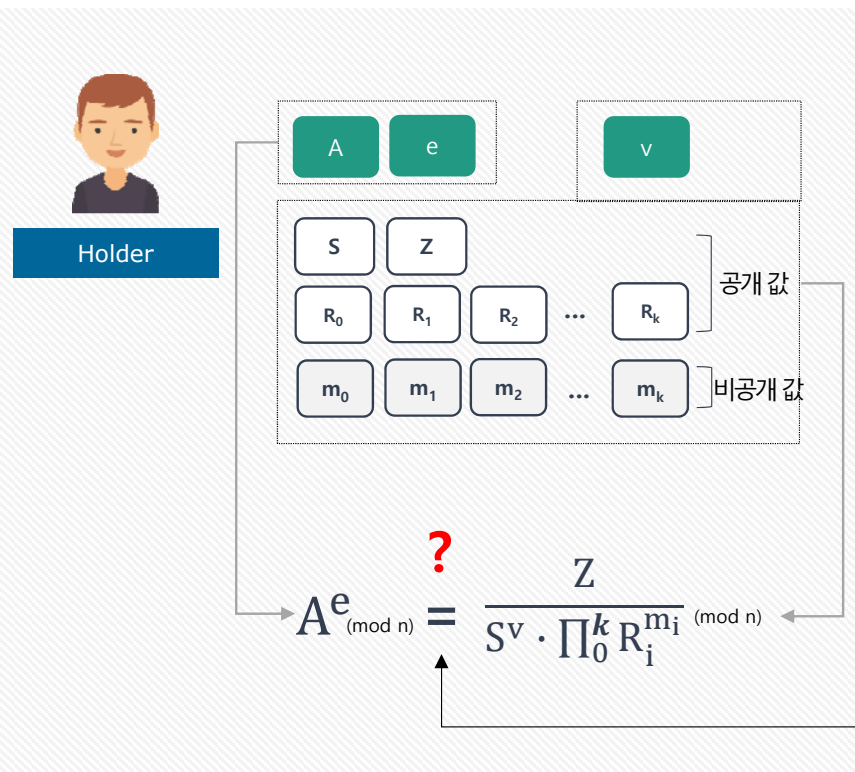
- CL(Camenisch-Lysyanskaya) Signature – Credential 발급



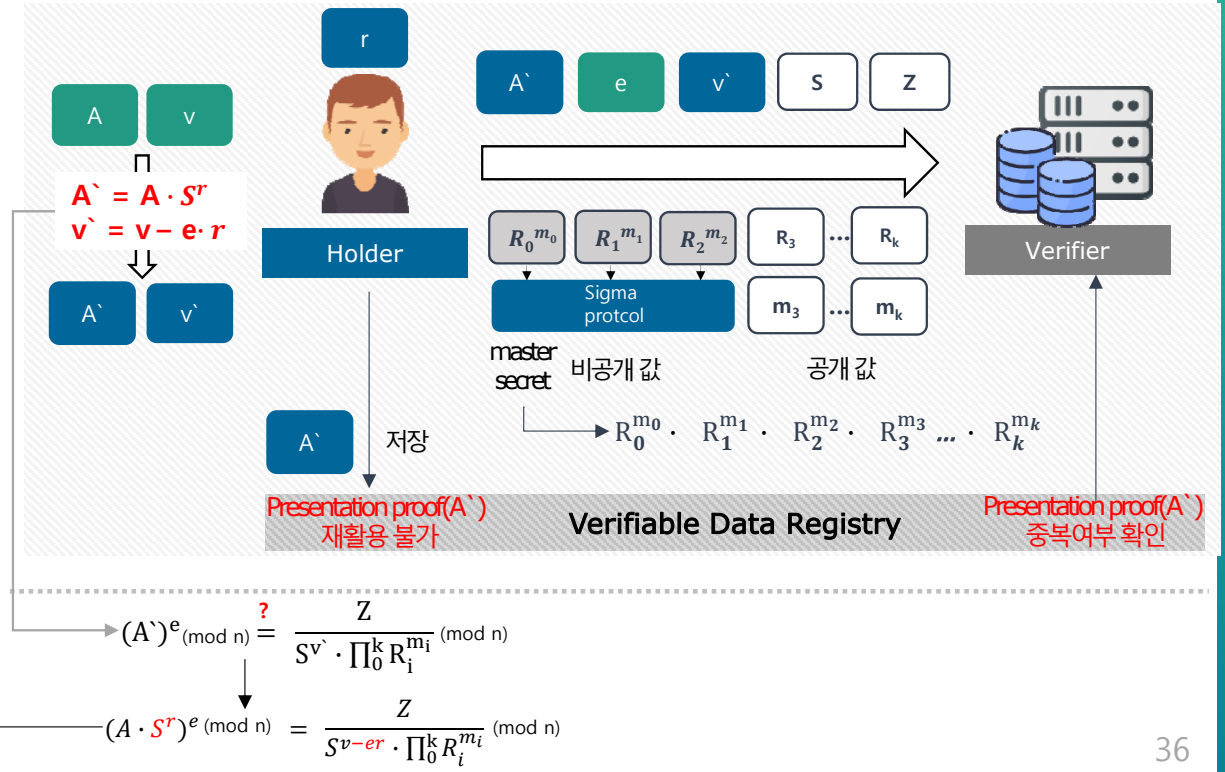
Selective disclosure

- CL(Camenisch-Lysanskaya) Signature – Presentation 생성 및 검증

Issuer의 서명 확인



Presentation 생성 및 검증



Selective disclosure의 성능

• 실행 환경



Issuer



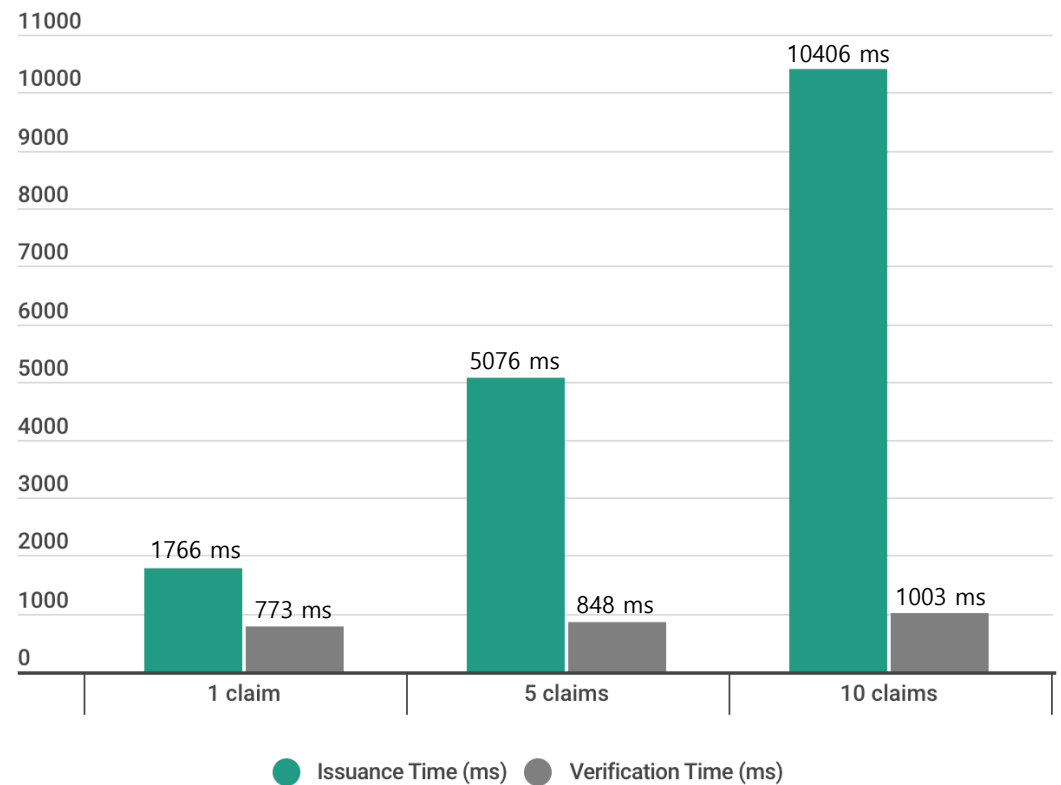
Verifier

Raspberry Pi

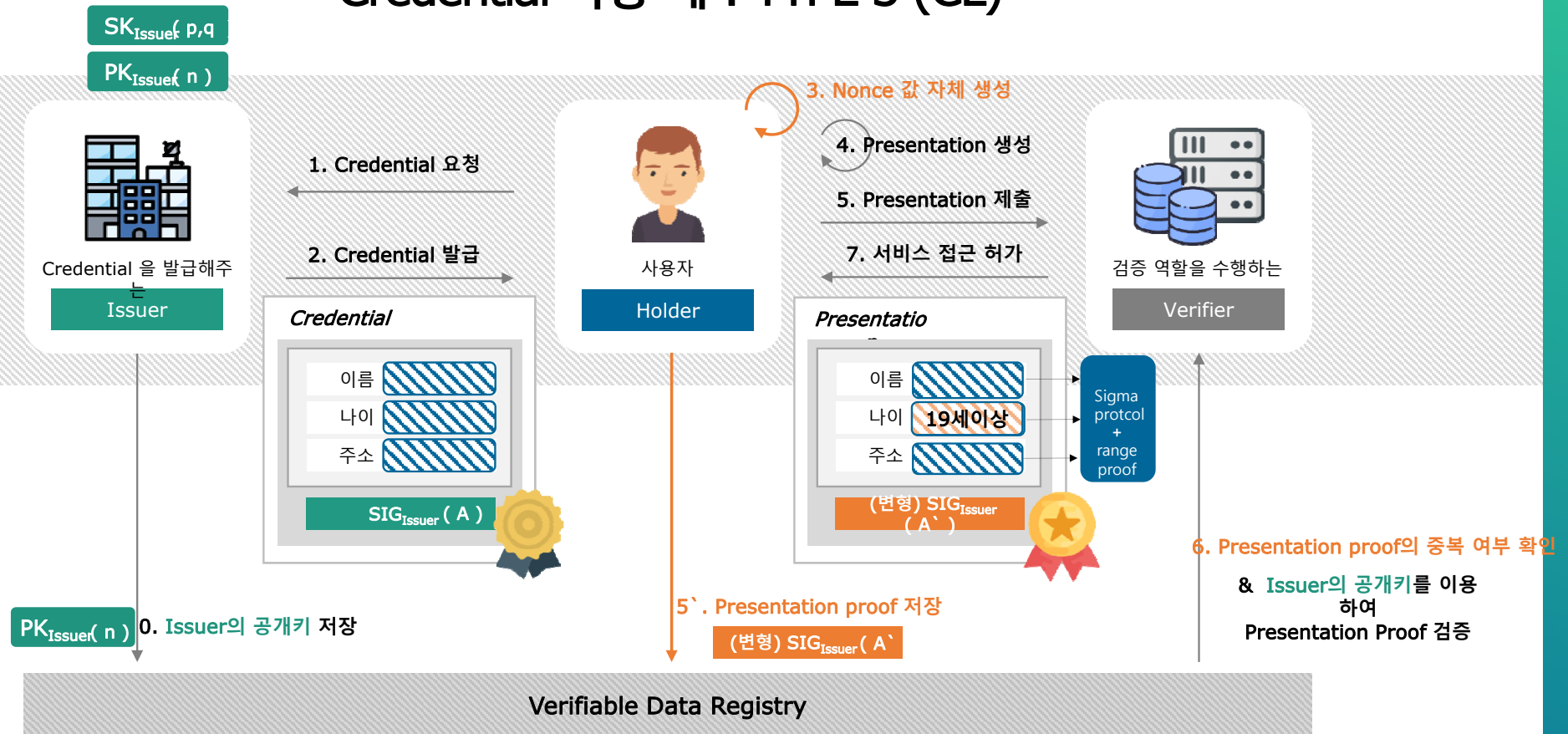
Model Pi 3 B+

1 Go of SDRAM LPDDR2

64 bit quad core processor 1.4GHz



Credential 사용 예 : TYPE 3 (CL)

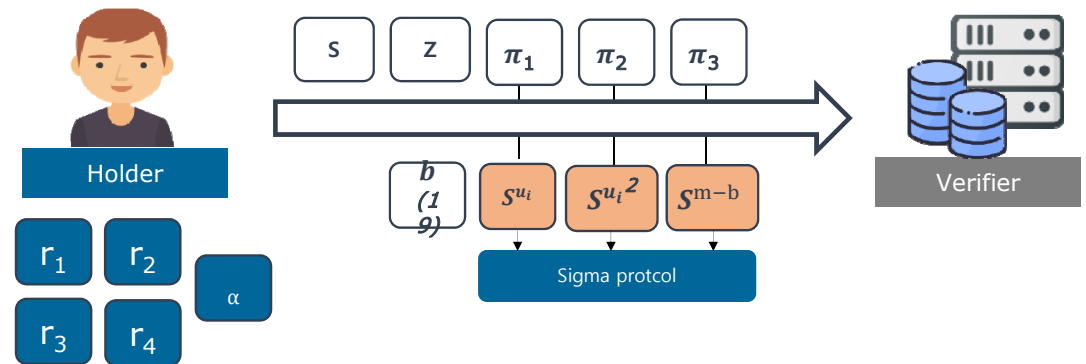


Range proof(2)

- 34세임을 드러내지 않고 19세 이상임을 증명 가능

- $m(34) \geq b(19)$, $\Delta = m - b$ m :비공개, b :공개
- $\Delta = u_1^2 + u_2^2 + u_3^2 + u_4^2$ <라그랑주의 네 제곱수 정리>
(Lagrange's four-square theorem)

모든 양의 정수는 네 개의 제곱수로 표현 가능하다.



π_1 : T_1 T_2 T_3 T_4 $T_i = S^{u_i} \cdot Z^{r_i}$, r_i : 랜덤값
 u_i 를 드러내지 않고 S^{u_i} 증명 $i = 1, 2, 3, 4$

π_2 : T_Δ $T_\Delta = (T_1^{u_1} \cdot T_2^{u_2} \cdot T_3^{u_3} \cdot T_4^{u_4}) \cdot Z^{4a}$, a : 랜덤값
 u_i 를 드러내지 않고 $S^{u_1^2 + u_2^2 + u_3^2 + u_4^2}$ 증명

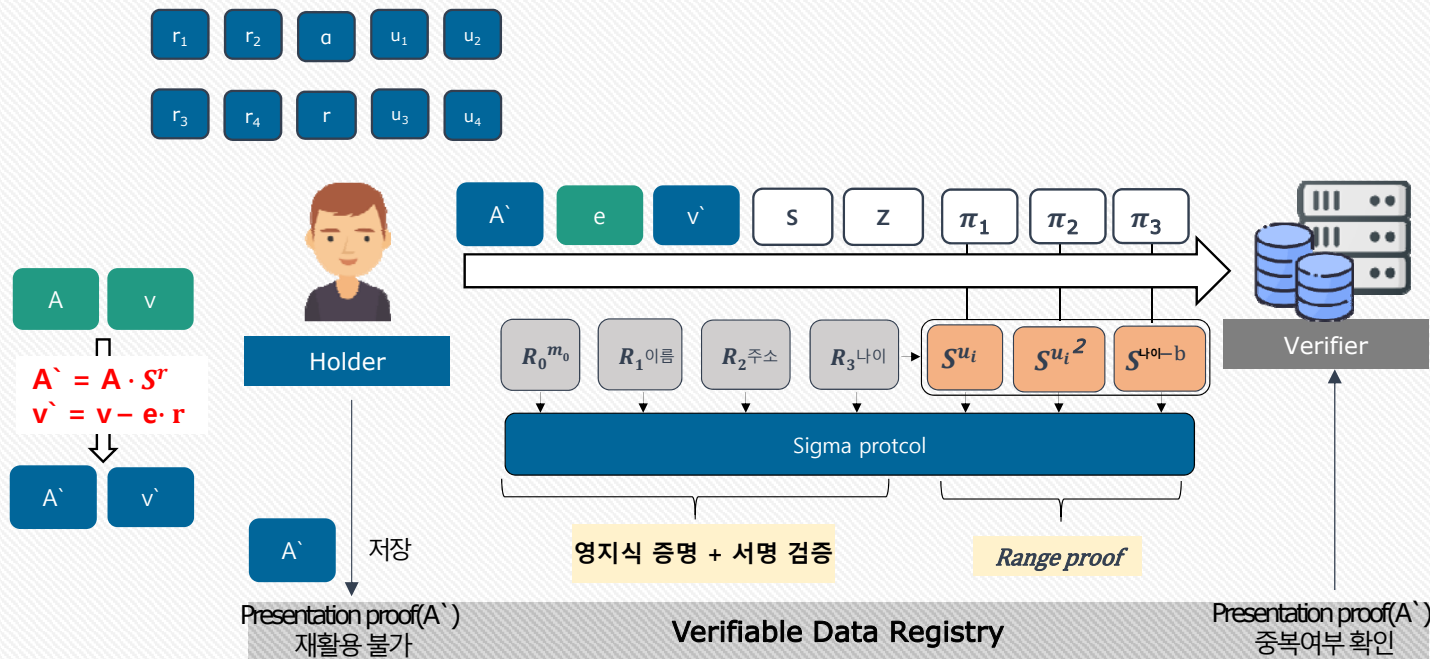
π_3 : $S^b \cdot T_\Delta = S^m \cdot Z^{r_\Delta}$
 $m - b$ 를 드러내지 않고 $S^{u_1^2 + u_2^2 + u_3^2 + u_4^2} = S^{m-b}$ 증명

$$\begin{aligned}
 & \xrightarrow{\text{by } \pi_2} S^b \cdot \{(T_1^{u_1} \cdot T_2^{u_2} \cdot T_3^{u_3} \cdot T_4^{u_4}) \cdot Z^{4a}\} = S^m \cdot Z^{r_\Delta} \\
 & \xrightarrow{\text{by } \pi_1} S^b \cdot \{(S^{u_1} \cdot Z^{r_1})^{u_1} \cdot (S^{u_2} \cdot Z^{r_2})^{u_2} \cdot (S^{u_3} \cdot Z^{r_3})^{u_3} \cdot (S^{u_4} \cdot Z^{r_4})^{u_4}\} \cdot Z^{4a} = S^m \cdot Z^{r_\Delta} \\
 & \downarrow \\
 & S^b \cdot \{(S^{u_1^2} \cdot Z^{r_1 u_1}) \cdot (S^{u_2^2} \cdot Z^{r_2 u_2}) \cdot (S^{u_3^2} \cdot Z^{r_3 u_3}) \cdot (S^{u_4^2} \cdot Z^{r_4 u_4})\} \cdot Z^{4a} = S^m \cdot Z^{r_\Delta} \\
 & \downarrow \\
 & S^b \cdot \{(S^{u_1^2 + u_2^2 + u_3^2 + u_4^2}) \cdot (Z^{r_1 u_1 + r_2 u_2 + r_3 u_3 + r_4 u_4 + 4a})\} = S^m \cdot Z^{r_\Delta} \\
 & \downarrow \\
 & S^{b + u_1^2 + u_2^2 + u_3^2 + u_4^2} \cdot Z^{r_1 u_1 + r_2 u_2 + r_3 u_3 + r_4 u_4 + 4a} = S^m \cdot Z^{r_\Delta} \\
 & \downarrow \\
 & b + u_1^2 + u_2^2 + u_3^2 + u_4^2 = m \longrightarrow \boxed{u_1^2 + u_2^2 + u_3^2 + u_4^2 = m - b}
 \end{aligned}$$

* r_Δ = 검증값

19

Selective disclosure + range proof



영지식 증명 + 서명 검증

- $R_0^{m_0}$... $R_3^{\text{나이}}$

Sigma protocol (영지식 증명)
- $(A)^e \pmod n \equiv \frac{Z}{S^v \cdot \prod_0^k R_i^{m_i}} \pmod n$

Range proof

$$u_1^2 + u_2^2 + u_3^2 + u_4^2 = \text{나이} - b$$

(Lagrange's four-square theorem)

Sigma protocol (영지식 증명)









PART 4.


활용 사례

PART 4. 활용 사례

DID 플랫폼

DIGITAL IDENTITY STANDARDS AND PLATFORMS

BLOCKCHAIN	DID PLATFORM	DID STANDARD	ECOSYSTEM TOKEN	FOCUS / KEYWORD
 ethereum		----	Minime ERC20 (BLT)	Credit Scoring International Accessible
		 ERC7056 ERC780	----	Identity Claims Usability Developer Ease of Use
 HYPERLEDGER		 Indy	----	Global Adoption Standardizing/Protocol Creation Government Involvement
	 civic	Unknown	ERC20 (CVC)	Protocol Reusability ID Replacement Global Adoption

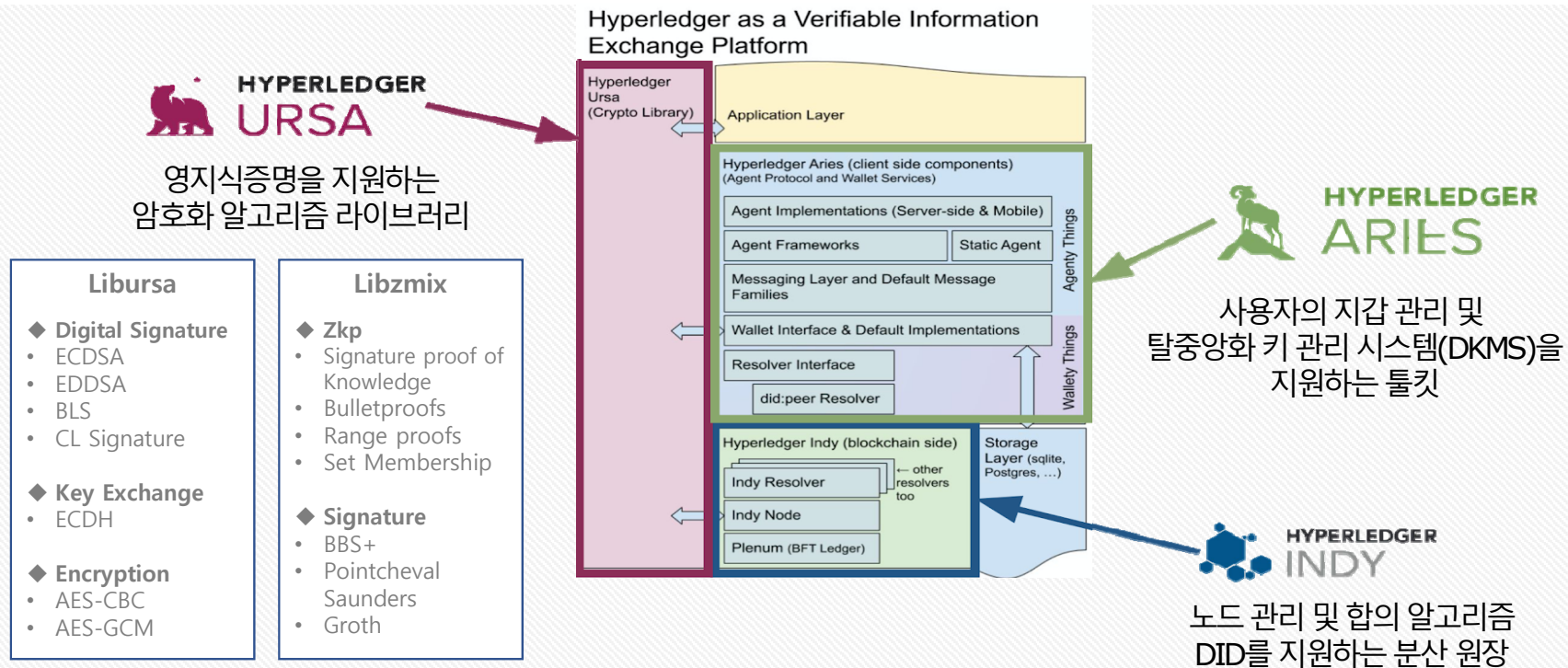
 LINUMLABS

출처 : LINUMLABS
(2018)

PART 4. 활용 사례

Sovrin

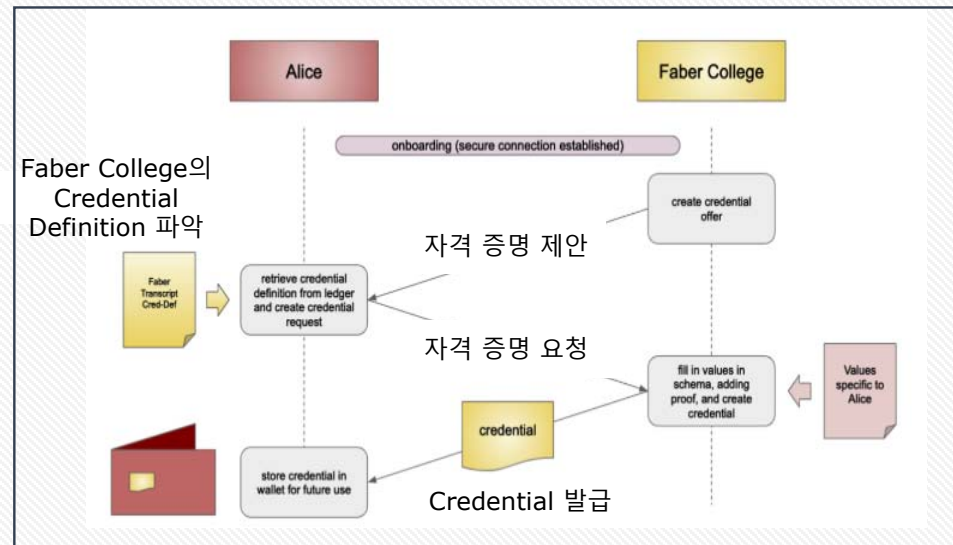
- 2016년에 미국에서 설립되었으며 DID 표준 및 Hyperledger Indy를 주도
- Hyperledger Indy와 ARIES, URSA를 활용하여 다양한 기능을 제공



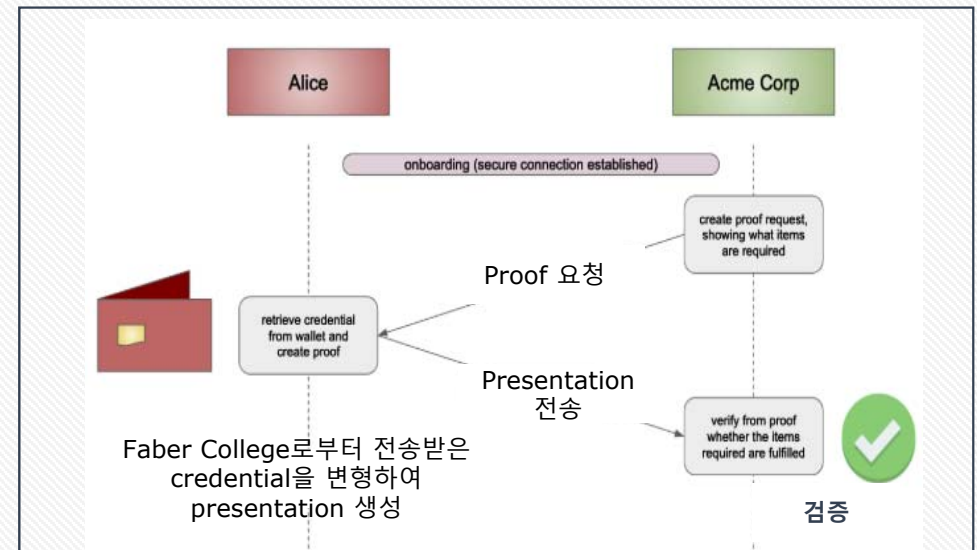
Sovrin

- 적용 예제 : Alice가 College로부터 Credential을 발급받아 Acme Corp에 사용 시나리오

College의 Credential 발급



Corp 에서의 Presentation 검증



PART 4. 활용 사례

Uport

- 2016년에 미국에서 설립되어 이더리움을 기반으로 한 DID 솔루션 제시
 - 블록체인 최대 개발 회사인 Consensys의 지원을 받고 있음
 - DID Library에 많은 기여를 하였으며, 개발 SDK를 제공

DID Library

DID-JWT

자바스크립트에서
JWT 형태로 DID를 생
성하고 검증하기 위한
library

DID-JWT-VC

자바스크립트에서
Verifiable credential
& presentation을
생성하기 위한 library

ETHER-DID- RESOLVER

이더리움 환경을
이용해 DID를
관리하기 위한 library

WEB-DID- RESOLVER

http 환경에서의 DID
관리를 위한 library

DID Solution

THE CITY OF UPORTLANDIA

City ID
4 credentials included

사용자의 정보를 바탕으로 시민권 생성

YOU'LL BE ASKED TO SHARE
First Name
Last Name
Address
Date of Birth

GET CITY ID

THE UNIVERSITY OF UPORTLANDIA

Diploma
4 credentials included

시민권을 바탕으로 대학교 졸업증 생성

YOU'LL BE ASKED TO SHARE
City ID

GET DIPLOMA

DREAM JOB LLC.

Employment Verification
3 credentials included

시민권과 졸업증으로 고용 증명서 발급

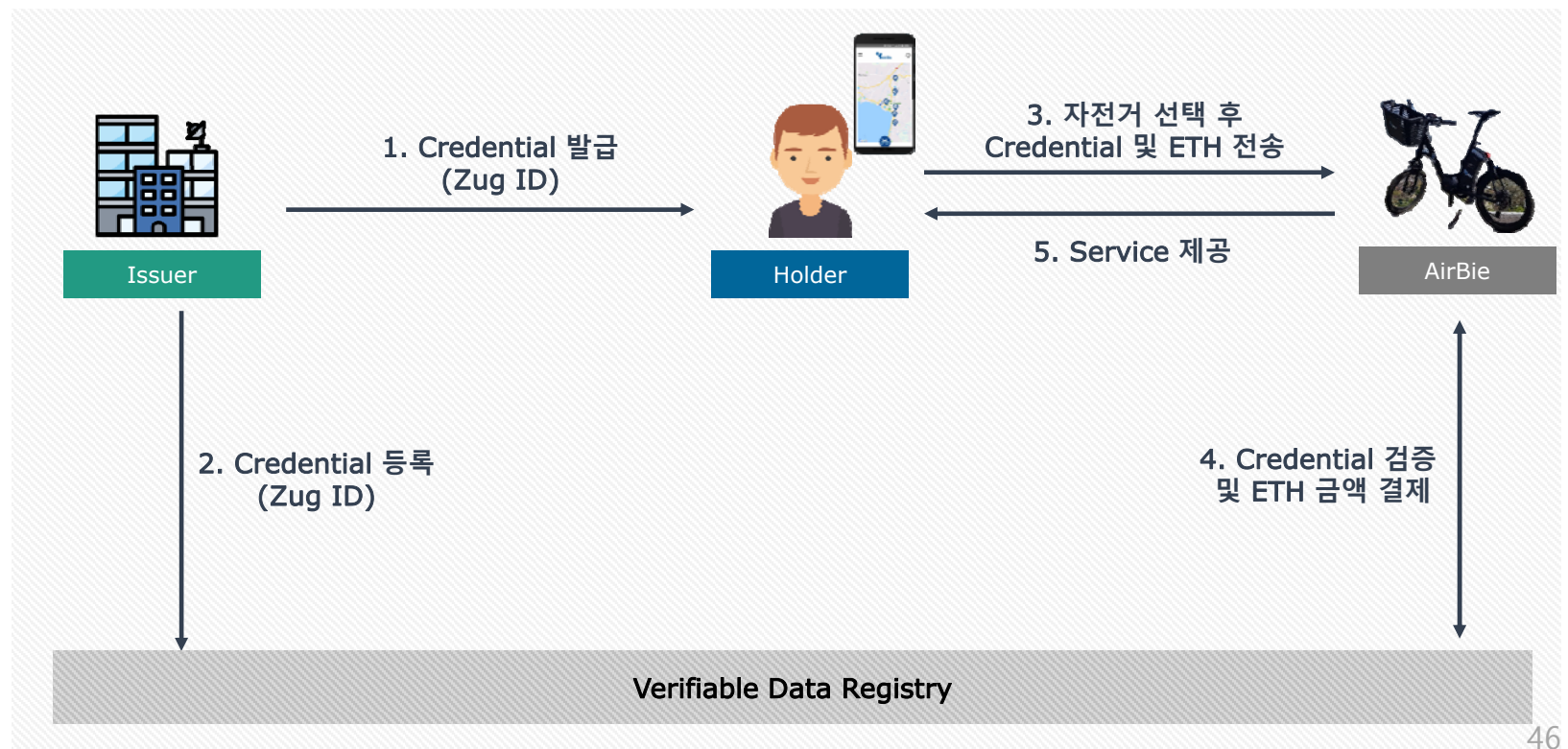
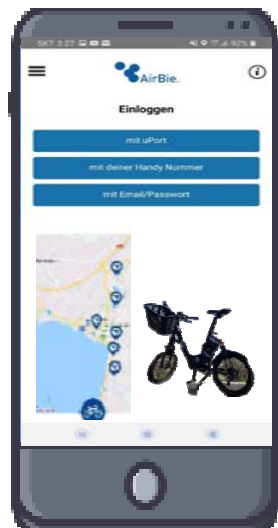
YOU'LL BE ASKED TO SHARE
City ID
Diploma

GET EMPLOYMENT VERIFI...

PART 4. 활용 사례

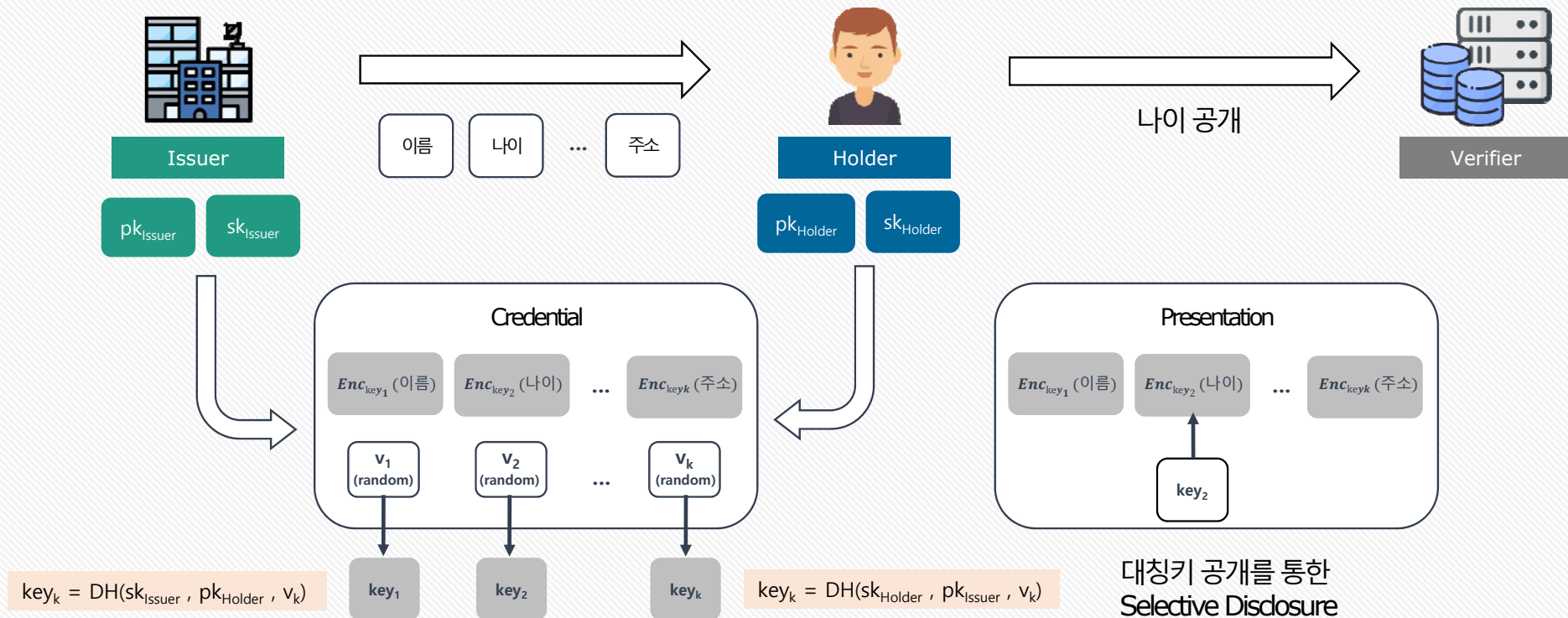
Uport

- AirBie: 블록체인 기반 시민 ID를 발급하여 자전거 대여 시스템 제공(스위스 Zug) – Uport 기반의 앱



Uport

- Uport의 서명 알고리즘
 - DH(Diffie Hellman key exchange)를 통한 Selective disclosure 지원



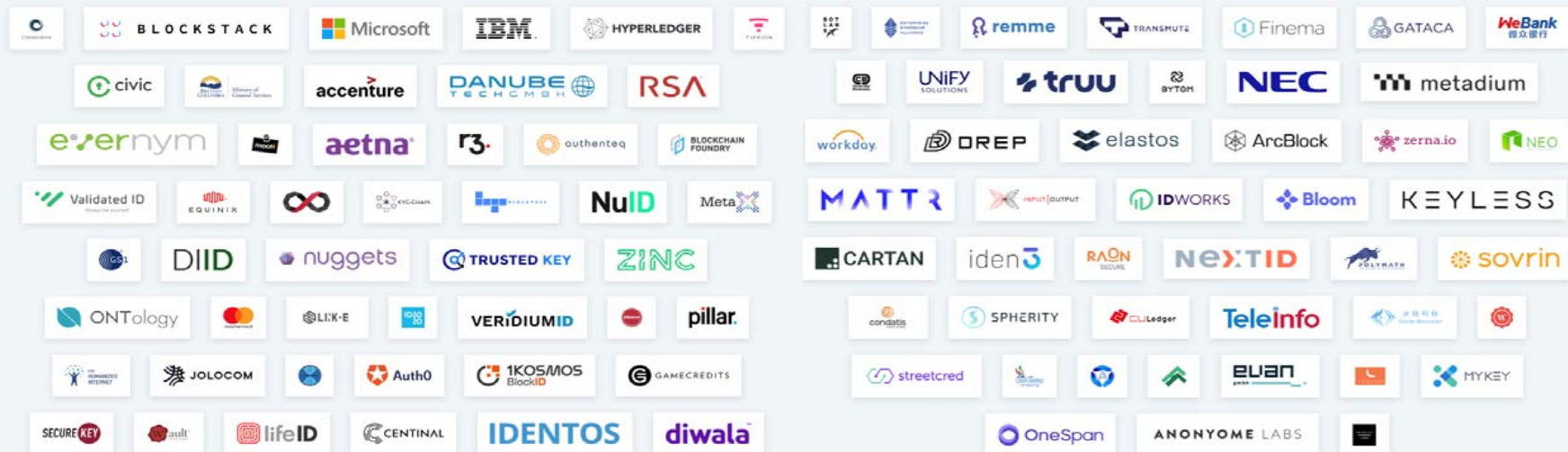
PART 4. 활용 사례

DIF

- Decentralized Identity Foundation



- 탈중앙화된 신원 증명을 개발하는데 중점을 둔 엔지니어링 중심의 조직
- 설립일: 2017년 5월
- 회원 수: 93 개 (2020. 06. 12 기준)



PART 4. 활용 사례

DIF

- Decentralized Identity Foundation



기술 SPEC

개발을 위한 spec과 표준 개발



구현 레퍼런스

프로토콜 구현 및 DID 개발을 위한 오픈소스 구현

GROUP

IDENTIFIERS & DISCOVERY

- DID 시스템
- 분산 식별자
- Universal Resolver
- 31개의 github repository

Authentication

- DID 암호화 알고리즘
- DID 인증 프로토콜
- 데이터 구조 spec
- 4개의 github repository

Claims & Credentials

- Claims, Credential의 생성, 증명
- Presentation 프로토콜
- 6개의 github repository

DID Communication

- 전송수단에 따른 통신방법
- 1개의 github repository

Side tree Development & Operating Group

- DID 확장성을 위한 Sidetree 프로토콜
- 3개의 github repository

Secure Data Storage

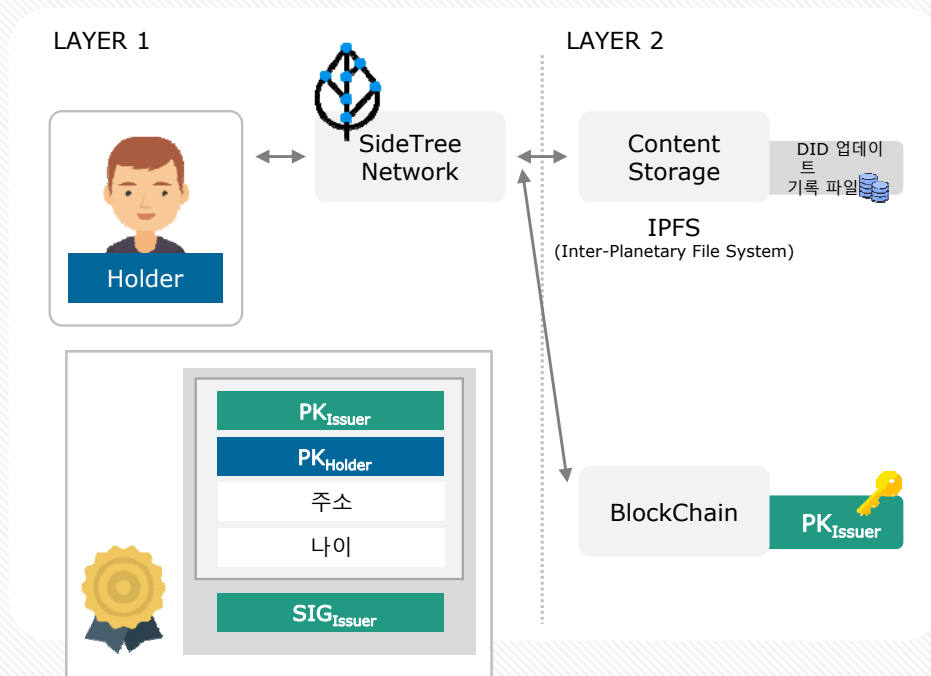
- 데이터 저장의 보안
- 저장 및 전송 프로토콜
- 액세스 제어
- 데이터 동기화
- 1개의 github repository

*아직 Identifiers & Discovery 그룹에 집중

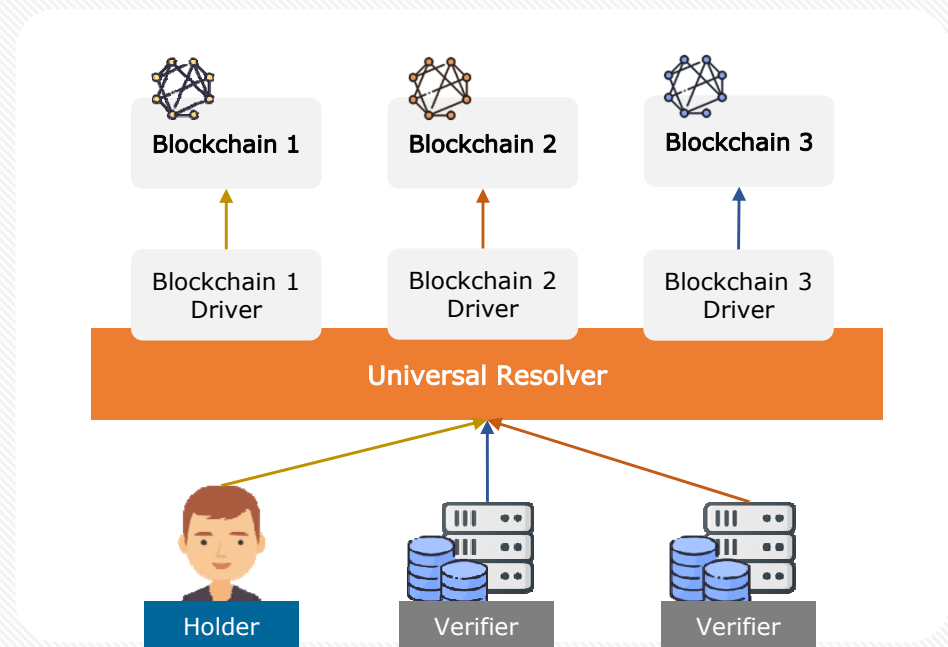
DIF

- Interoperability를 위한 SideTree, Universal Resolver

SideTree



Universal Resolver



PART 4. 활용 사례

국내 DID 연합



블록체인 연합	DID 얼라이언스	MYiD 얼라이언스	DID 이니셜	MYKEEPiN 얼라이언스
블록체인 개발사	라온시큐어	아이콘루프	SKT	코인플러그
플랫폼	라온시큐어 블록체인 (옴니원)	아이콘루프 블록체인 (루프체인)	SKT 블록체인 (스톤)	코인플러그 블록체인 (메타디움)
참여기관 및 기업수	56개	62개	14개	30여곳
출범일	2019년 11월	2019년 11월	2019년 7월	2020년 4월
특징	DID 기술 표준화 병무청 블록체인 인증	금융권 중심 디지털 신분증 개발	성적증명, 졸업증명, 모바일 전자증명 등	게임, 미디어, IoT 등 광 범위한 산업에 적용 예 정
주요 이력	KISA 시범사업 참여	금융규제 샌드박스 지정	KISA 시범사업 참여	부산 블록체인특구 DID 서비스

PART 4. 활용 사례 카카오톡 인증서비스

• 카카오톡, 산업인력공단과 MOU 체결(2020-07-07)

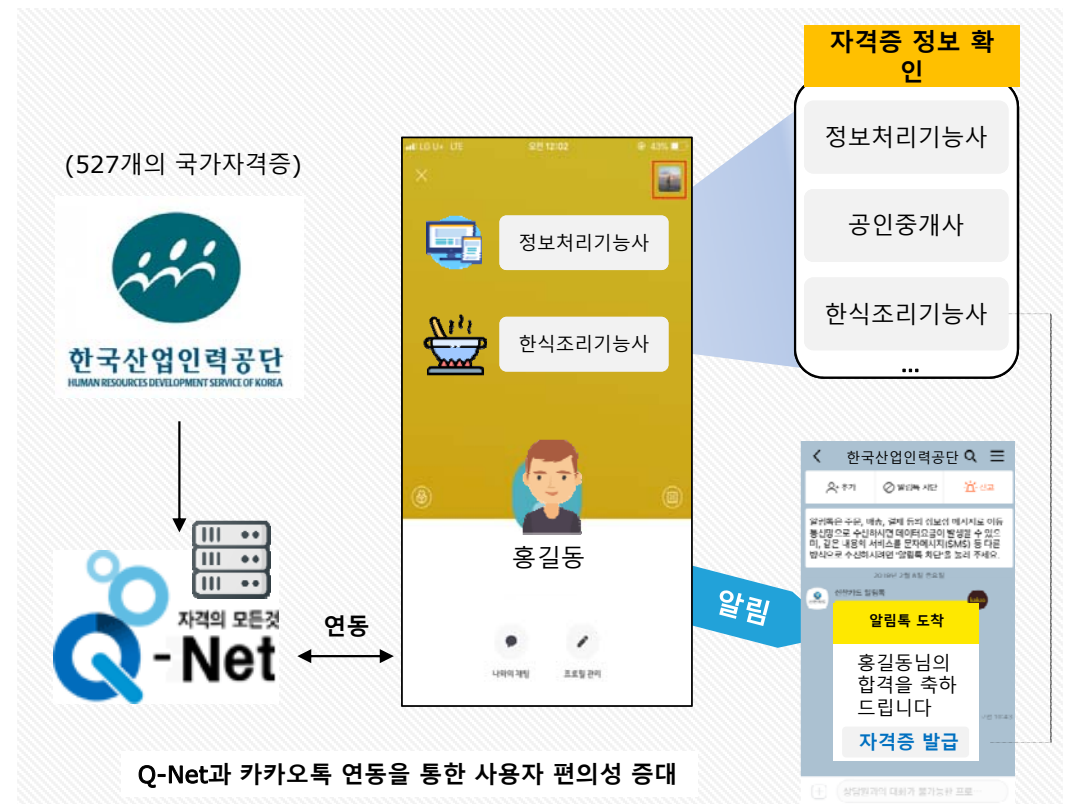
카톡으로 자격증 인증...카카오의 실험

‘생활형 플랫폼’을 지향하는 카카오가 자격 증명 서비스에 뛰어든다. 4519만명의 이용자를 확보한 카카오톡에서 정보처리기사, 공인중개사, 한식조리기능사 등 국가자격시험 관련 정보를 간편하게 확인할 수 있게 만든다는 구상이다. 카카오톡에서 자신이 보유한 국가자격증을 한번에 확인하고, 모바일로도 자격증 인증이 가능하게 하면 카카오톡 이용자의 편의성이 높아지고 이와 연계된 사업모델도 촉발될 수 있을거란 판단에서다.

7일 카카오(공동대표 여민수·조수용)는 한국산업인력공단(이사장 김동만)과 업무협약(MOU)을 체결하고 국가자격정보 연계와 활용을 위해 협력하기로 했다.

그동안 공단은 국가자격정보 누리집인 큐넷(Q-net)을 통해 자격정보를 제공해 왔다. 이번 협약으로 카카오톡과 포털사이트 다음을 통해 관련 정보를 제공할 수 있게 된다. 국가자격 수험자와 취득자는 시험 일정, 시험 장소, 합격 여부 확인 등 자격과 관련한 다양한 정보를 확인해야 한다. 이 같은 콘텐츠를 카카오톡 알림 혹은 카카오톡 내 서비스 개설로 확인할 수 있게 돼 접근성이 높아지는 것이다. 김동만 한국산업인력공단 이사장은 “이번 협약으로 카카오 플랫폼을 통해 국가 자격 수험자와 취득자에게 국가자격정보를 쉽게 전달할 수 있게 됐다. 국가자격 정보의 접근성을 높이기 위해 노력하겠다”고 말했다.

향후 카카오톡 내부에 본인이 취득한 국가자격증을 한눈에 확인할 수 있는 서비스 앱을 개설할 가능성도 있다. 예를 들어 본인이 공인중개사 자격증을 취득한 사람인지를 확인해주는 것이다. 이후 인증이 완료되면 카카오톡 프로필 화면에 취득한 자격증을 노출할 수 있게 하는 형태다. 카카오 플랫폼을 통해 이용자의 전문성과 경력이 공식적으로 인증되면 피싱이나 전문가 사칭 등 각종 사기를 줄일 수 있는 토대가 된다.



Login Form
Login to access your dashboard

email@example.com

password

LOGIN

아이디 & 패스워드

간편인증 문자인증

PASS 신규가입하면 이벤트 자동 참여!
추첨을 통해 신세계 상품권10,000원,
스타벅스 쿠폰까지~
*경품은 이벤트 상황에 따라 달라질 수 있습니다.

이름

생년월일/성별 -

휴대폰번호

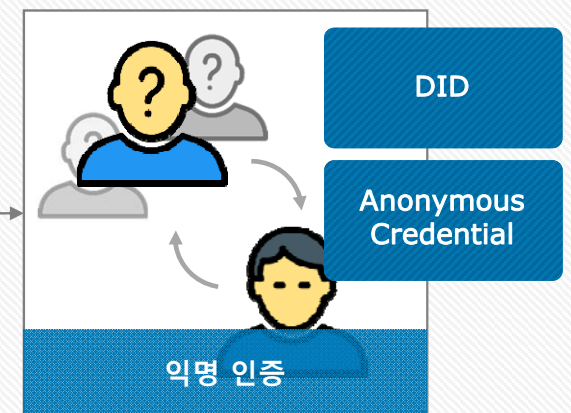
본인인증

졸업증명서

성명: 홍길동
생년월일: 2000년 01월 01일
학사: 컴퓨터공학과
학번: 200000001
졸업연월일: 2008-02-14
졸업유형: 일반
학위등록번호: 제215-2008-0001

이 사실을 증명합니다
2023년 02월 01일
브 대학교 총장

신원인증



Thanks for Listening

Q&A