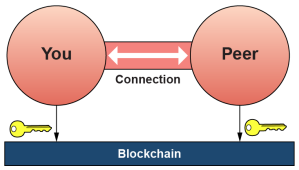
- From the book self-sovereign identity

디지털 신원인증 모델

- Centralized
  - e.g. Account-based website
  - [You] → [Org]
- Federated
  - service/identity provider (IDP) in the middle
  - "federation": collection of all the sites that use the same IDP (or group of IDPs)
  - [You] → [IDP] → [Org]
- **Decentralized**
  - A new model, inspired by blockchain technology, since 2015
  - peer-to-peer
  - *공개/개인키 암호화 기법* 기반 *블록체인* 사용
  - 블록체인 기술을 암호화폐가 아닌, DPKI (Decentralized PKI)에 적용
    - 공개키를 직접 교환 하여 private하고 안전한 peer-to-peer 연결 생성
    - 공개키를 블록체인에 저장하여 디지털 신원 자격(VC) 서명(signature) 증명
      - VC(verifiable credentials): 실생활에서 신원증명 제공을 위해 교환 가능한 자격 증명
    - 개인키는 디지털 지갑에 저장
- 블록체인 사용 이유



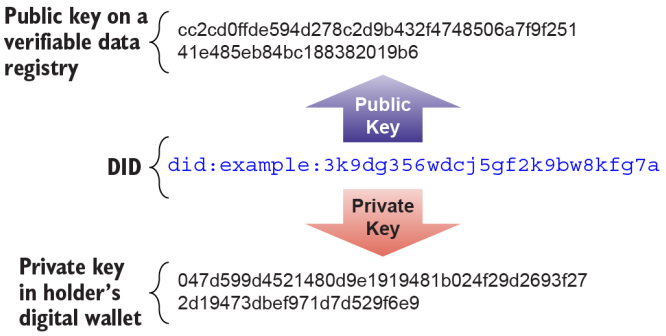
암호화 기법

- 비대칭 키 (Asymmetric-key cryptography)
  - e.g. DID의 개인키는 디지털지갑, 공개키는 블록체인에 저장
- 디지털 서명
  - 키페어 중 개인키로 생성, 공개키로 검증
  - *공개키 암호화 기법* 에 기반

DID 배경

- ip주소 자체는 해당 ip를 소유한 신원 대상에 대한 어떠한 정보도 제공하지 않음: 디지털 proof필요
- *공개/개인키 암호화기법* 으로 증명가능 한 디지털 proof 제공

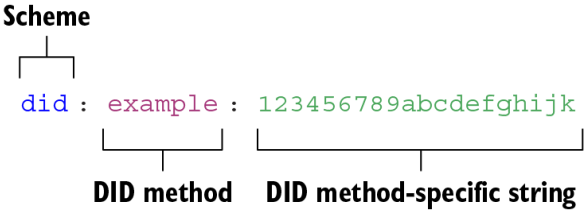
    - 개인키로 *메시지* 서명(sign), 공개키로 검증(verify)
- 공개 키 기반 구조 ***PKI*** 도입
    - 신원 검증자가 신원 대상자의 공개키가 실제로 소유자의 것인지 - *공개키의 소유권* 을 검증할 수 있게 됨
    - 검증된 CA기관들로 부터 공개키 인증서(public key certificates) 발행하는
    - 중앙집권형 시스템이므로 개인들이 여러개의 암호화 키페어를 가지고 있는 환경(SSI)에 한계가 있음
- 탈중앙 인증 식별자 DIDs 도입
    - 영속성 (permanent)
    - 분해성 (resolvable to document)
    - 암호화기법으로 검증 가능 (cryptographically verifiable)
        - 신원 소유자가 암호화 기법으로 개인키 검증가능
        - 암호화기법으로 DID 생성
        - DID는 1개의 공개/개인키와 연결되므로 개인키 소유자(controller)가 DID 소유자(controller)임 증명 가능
    - 탈중앙화 (decentralized)
        - 암호화기법을 사용하여 중앙 신원인증 기관들(CAs)의 통제 없이, 블록체인 등 탈중앙화 네트워크에 기반함
        - 공개/개인키 생성하는 암호화 알고리즘은 프라임넘버, 랜덤숫자생성기, 타원곡선 암호학에 기반하여 globally 고유한 식별자를 만들기 때문에 중앙기관 없이 고유성 검증가능



DID 정의

- 새로운 타입의 *globally unique identifier*
- DIDs are the cryptographic counterpart to verifiable credentials (VCs)
- DID는 블록체인에 공개키 주소로서 역할을 하며, DID subject의 agent를 찾는데도 사용
- DID 메소드를 통해 블록체인, DLT 등을 이용할 수 있도록 설계됨

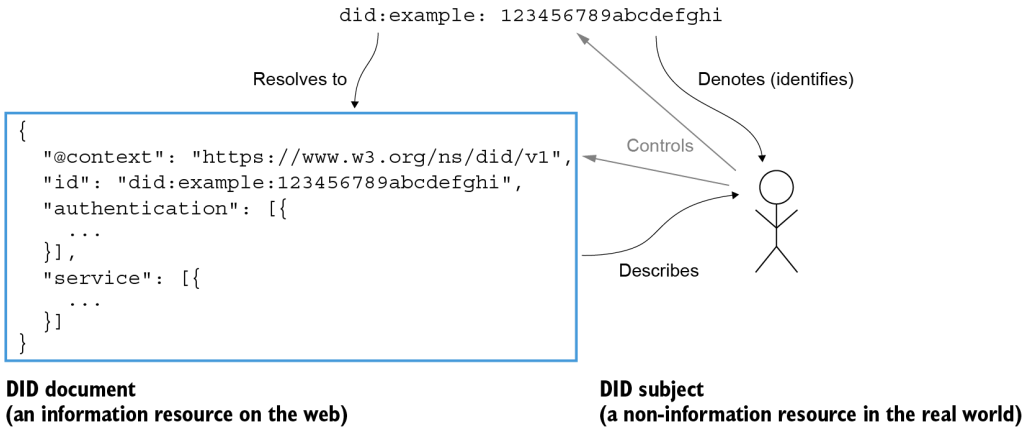
- 소프트웨어를 통해 누구나 DID 메소드(sov,btcr,ethr, ...)를 사용하여 중앙기관 통제 없이 DID 발행 및 사용가능
- DID를 생성하는 것은 비트코인이나 이더리움 블록체인에 공개 지갑 주소를 생성하는 것과 동일한 프로세스 - DID 탈중앙화 핵심



- DID 예시
  - 예시 1
    - 디지털 지갑 앱 개인키 저장
    - 공개키 → 블록체인 (sovrin, bitcoin, ethereum, ...)
    - 공개키를 암호화하여 트랜젝션을 통해 블록체인에 저장
    - 블록체인은 응답으로 DID 생성 및 반환
    - 은행 로그인 시 DID를 개인키로 서명하여 요청
    - 은행은 블록체인에서 DID와 연관된 트렌젝션 조회 & 공개키 조회
    - 공개키로 서명 검증 및 로그인 완료처리
  - 예시 2
    - 학생정보 입력하여 학교웹사이트 로그인
    - 나의페이지 Dashboard에서 제공하는 고유 디지털 ID를 디지털 지갑 앱으로 스캔및 bio인증하여 고유 식별자(DID) 생성
      (개인키 생성 및 블록체인에 공개키저장하여 DID 생성)
    - 온라인서적 사이트에서 DID로그인

1. DID documents

- *DID → DID resolver(software/hardware) → DID document*
  - 디지털 신원인증 앱, 디지털 지갑, 또는 에이전트 등에서 인증을 위한 기초 빌딩블록 으로 사용
  - DID ↔ DID document (1대1 대응)
- DID document는 표준화된 규격 구조(json)를 가지고 있으며 다음을 포함 :
  - 공개키: 거래시 DID subject를 검증하기 위함 - essence of DPKI
  - 서비스: 프로토콜을 통한 거래 시에 사용 할 DID subject 관련 서비스들
  - 메타데이터: 타임스탬프, 디지털서명, 암호학적proof, deleation 및 인증 관련 메타데이터




```
// 1개의 공개키와 1개의 서비스를 가진 DID document 구조
{
  // The first line is the JSON-LD context statement,
  // required in JSON-LD documents (but not in other DID document representations).
  "@context": "https://www.w3.org/ns/did/v1",
  // DID being described
  "id": "did:example:123456789abcdefghi",
  // public key for authenticating the DID subject.
  "authentication": [{
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "Ed25519VerificationKey2018",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyBase58" : "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }],
  // service endpoint for exchanging verifiable credentials.
  "service": [{
    "id":"did:example:123456789abcdefghi#vcs",
    "type": "VerifiableCredentialService",
    "serviceEndpoint": "https://example.com/vc/"
  }]
}
```

2. DID methods

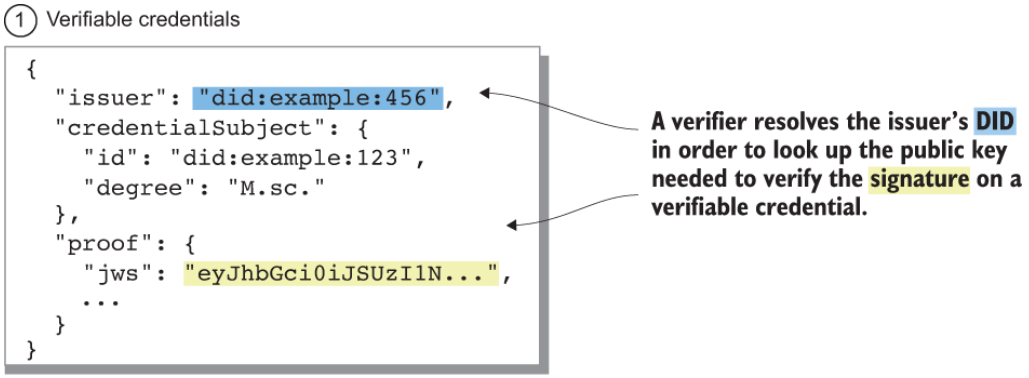
- 각 DID 메소드는 다음과 같은 기술적 스펙 정의가 요구됨:
  - 메소드 고유 식별 (예: sov,btcr,v1,ethr,jolo,...)
  - DID에 대한 CRUD 4가지 operation 수행 가능
    - 블록체인이나, 분산 ledger시스템에 기반한 DID메소드의 경우 create/update시 ledger에 트랜젝션 기록
  - 메소드에 따른 보안 및 개인정보보호 장치

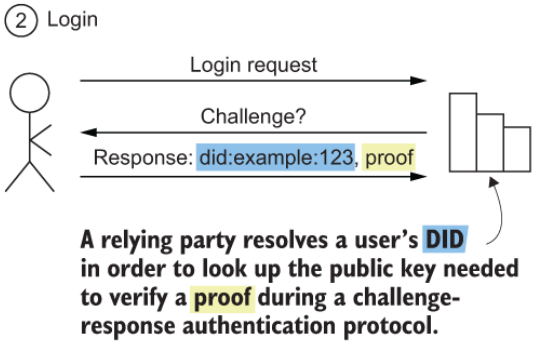
```
did:sov:WRfXPg8dantKVubE3HX8pw
did:btcr:xz35-jzv2-qqs2-9wjt
did:v1:test:nym:3AEJTDMSxDDQpyUftjuoeZ2Bazp4Bswj1ce7FJGybCUu
did:ethr:0xE6Fe788d8ca214A080b0f6aC7F48480b2AEfa9a6
did:jolo:1fb352353ff51248c5104b407f9c04c3666627fcf5a167d693c9fc84b75964e2
```

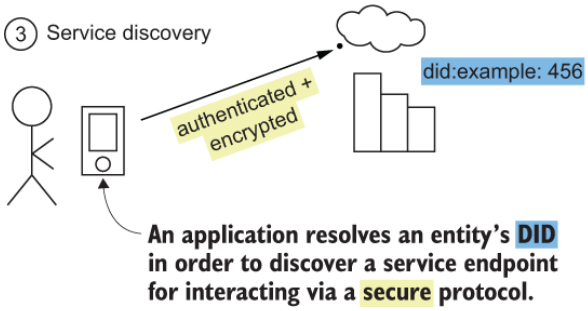
- https://w3c.github.io/did-rubric
  - “Rubric” document to help adopters evaluate how well a particular DID method will meet the needs of a particular user community:

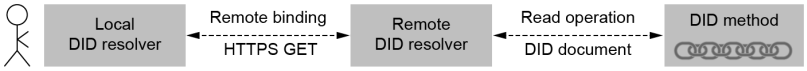
3. DID resolution

- DID로부터 DID document를 얻는 과정
- DID관련 앱이나 서비스가 DID대상(subject)와 관련된 메타데이터를 얻어서 다음과 같은 추가 상호작용 :
  1. VC 발행자로 부터의 디지털 서명을 검증할 공개키 조회
  2. DID 컨트롤러가 웹사이트나 앱에 로그인해야할 때 검증 진행
  3. 웹사이트, 소셜 네트워크 또는 라이선스 기관과 같은 DID 컨트롤러와 관련된 잘 알려진 서비스를 검색하고 액세스
  4. DID 컨트롤러로 DID-to-DID 연결을 요청









4. DID URLs

5. DIDs 타입

Category	Description and examples

Category	Description and examples
Ledger-based DIDs	The original category of DID methods involves a blockchain or other distributed ledger technology (DLT), which serves the purpose of a registry that is not controlled by a single authority. This registry is typically public and globally accessible. A DID is created/updated/ deactivated by writing a transaction to the ledger, which is signed with the DID controller's private key: did:sov:WRfXPg8dantKVubE3HX8pw did:btcr:xz35-jzv2-qqs2-9wjt did:ethr:0xE6Fe788d8ca214A080b0f6aC7F48480b2AEfa9a6 did:v1:test:nym:3AEJTDMSxDDQpyUftjuoeZ2Bazp4Bswj1ce7FJGybCUu
Ledger middleware (Layer 2) DIDs	An improvement to classic ledger-based DID methods, this category adds an additional storage layer such as a distributed hash table (DHT) or traditional replicated database system on top of the base layer blockchain. DIDs can be created/updated/deactivated at this second layer without requiring a base layer ledger transaction every time. Instead, multiple DID operations are batched into a single ledger transaction, increasing performance and decreasing cost: did:ion:test:EiDk2RpPVuC4wNANUTn_4YXJczjzi10zLG1XE4AjkcGOLA did:elem:EiB9htZdL3stukrklAnJ0hrWuCdXwR27TNDO7Fh9HGWDGg
Peer DIDs	This special category of DID method does not require a globally shared registration layer such as a blockchain. Instead, a DID is created and subsequently shared with only one other peer (or a relatively small group of peers). The DIDs that are part of the relationship are exchanged via a peer-to-peer protocol, resulting in private connections between the participants (see https://identity.foundation/peer-did-method-spec/index.html): did:peer:1zQmZMygzYqNwU6Uhmewx5Xepf2VLp5S4HLSwwgf2aiKZuwa
Static DIDs	There is a category of DID methods that are "static", i.e. they enable a DID to be created and resolved, but not updated or deactivated. Such DID methods tend to not require complex protocols or storage infrastructure. For example, a DID may simply be a "wrapped" public key, from which an entire DID document can be resolved algorithmically, without requiring any data other than the DID itself: did🔑z6Mkfriq1MqLBoPWecGoDLjguo1sB9brj6wT3qZ5BxkKpuP6

Category	Description and examples
Alternative DIDs	A number of other innovative DID methods have been developed that do not fall into any of the previous categories. They demonstrate that DID identification architecture is flexible enough to be layered on top of existing internet protocols, such as Git, the Interplanetary File System (IPFS), or even the web itself: did:git:625557b5a9cdf399205820a2a716da897e2f9657 did:ipid:QmYA7p467t4BGgBL4NmyHtsXMoPrYH9b3kSG6dbgFYskJm did:web:uport.me

DIDs가 작동하는 이유 (아키텍쳐 관점)

- Public Key Infrastructure (PKI)의 문제점
  - 해결책 1: 전통적 PKI 모델
  - 해결책 2: web-of-trust 모델
  - 해결책 3: 공개키 기반 식별자 (Public key-based identifiers)
  - 해결책 4: DIDs and DID documents
- DIDs의 4가지 장점 (that go beyond PKI)
  - 1: Guardianship and controllership
  - 2: Service endpoint discovery
  - 3: DID-to-DID connections
  - 4: Privacy by design at scale

DIDs의 의미

- 주소는 자체적으로 존재하지 않으며, 그것들을 사용하는 네트워크의 컨텍스트에서만 존재

Origin	Address type	Network
1994	Persistent address (URN)	World Wide Web (machine-friendly)
1994	Web address (URL)	World Wide Web (human-friendly)
2003	Social network address	Social network
2009	Blockchain address	Blockchain or distributed ledger network
2016	DID	DID network

- SSI Resources

- webinar