用户自治数字身份安全白皮书

CSA GCR cloud security alliance
GREATER CHINA REGION

致谢

云安全联盟大中华区（简称：CSA GCR）区块链安全工作组在 2020 年 2 月份成立。包括 100 多位安全专家们，分别来自中国电子学会、耶鲁大学、北京大学、北京理工大学、武汉大学、世界银行、华为、腾讯、知道创宇、赛博英杰、元界 DNA、慢雾科技、安比实验室、启明星辰、天融信、联想、OPPO、零时科技、安永、阿斯利康等五十多家单位。

区块链安全工作组有 9 个项目小组，包括智能合约安全、数字钱包安全、共识算法安全、交易所安全、Dapp 安全、去中心化数字身份（DID）安全、网络层安全、数据层安全，AML 技术安全等方向。

本白皮书主要由去中心化数字身份安全小组专家撰写，并由 DID 安全小组及 IAM 工作组的专家共同审核，感谢以下专家的贡献：

区块链安全组组长：黄连金

DID 安全小组的领军人物：初夏虎

白皮书作者名单：程伟强、初夏虎、黄连金、李程、李腾飞、刘洁、王登辉、于继万、袁运亮、周庆松（按照字母排序）

如本白皮书有不妥当之处，敬请读者联系 CSA GCR 秘书处给与雅正！联系邮箱：info@c-csa.cn；云安全联盟 CSA 公众号：

序言

数字身份是保障数字经济安全的信任基石，业界目前的数字身份体系一般都是中心化的，区块链作为解决可信问题的分布式技术，给数字身份自治的场景打开了天窗，比如分散云计算中心化身份数据大量聚合的泄露风险，在边缘计算分布式系统中使可信身份认证管理更加便捷私密等等。这本白皮书不是有关于用户自治数字身份（DID）的标准。DID 标准是 W3C 正在开发的一系列标准，包括 DID 数据模型，DID 方法，DID 通讯等等。本书主要是针对于希望用 DID 来进行技术开发或者应用落地的项目或公司需要注意的一些安全与隐私的问题，且分析为什么在新的数字化转型过程中 DID 能够解决的痛点问题，介绍目前国际已有的标准和案例。本白皮书目前是第一版本，因为 DID 本身的一系列标准还在开发之中，安全对于数字身份是第一要素，新的安全问题肯定会出现，欢迎读者专家们能够提出意见，使下一个版本的覆盖面更广，对于行业的发展能有更大贡献。

李雨航 Yale Li

CSA 大中华区主席兼研究院院长

目录

第一章：为什么需要用户自治的数字身份

1.1 传统的中心化数字身份

传统的数字身份是中心化的，主要可以按照下面的应用场景进行分类。

A. 企业内部使用身份（Enterprise Identity）

通常企业内用户身份主要是用于认证及授权。用户对于企业信息系统的访问，用户使用账号密码登陆，通过认证后，根据服务端的配置获得相关的访问授权。用户的账号及密码信息存储在认证及授权相关的服务器中（例如 LDAP 服务器），用户输入账号及密码，认证程序会对比相关服务器存储的信息，以确认该用户持有正确的用户账号及密码。从而允许用户访问，并授予相应的访问权限。

为了增强安全性，很多企业引入了多因子认证机制（ MFA: Multi-Factor Authentication），用户不仅仅需要提供所知道的用户账号及密码，还需要提供发送至个人持有设备（比如令牌、手机等）的随机信息或个人生物认证的信息（比如人脸、指纹、语音等）。还有一些企业采用 PKI（Public Key Infrastructure）体系，用户（主要是特权用户）持有私有密钥，用户在某些应用场景中需要使用私有密钥，或者采用私有密钥签名，做为身份的证明。

为了用户登陆企业内多个系统的便利性，很多企业引入了单点登陆机制。用户仅需要单点登陆服务器的一次身份认证，就可以访问具有相关授权的资源。

企业的用户访问授权主要关注以下几方面：

1. 职责分开（Separation of Duty: SOD）：为不同职责的用户分配不同的权限，避免单个职员有过多的、职责冲突的系统操作权限，从而在系统中进行业务授权之外的操作，产生业务风险。

2. 最小权限（Least Privilege）：授权采用按需最小的原则，仅授予职员工作所必须的操作权限。降低系统误操作、舞弊等风险。

3. 基于角色的访问控制（Role Based Access Control: RBAC）：企业应用系统，例如 ERP 等系统，一般采用基于角色的访问控制，通过对角色权限的一次性配置，然后将角色权限授予相关的用户。当用户的工作内容发生变更时，只需要调整相关的角色分配即可以完成相关访问权限的调整。

4. 基于属性的访问控制（Attribute Based Access Control: ABCA， 不同于基于角色的访问控制，ABAC 基于一组访问策略进行授权，认证授权服务器验证用户的身份后，基于一些特性（比如访问者职位、部门、登陆地点等）进行计算，根据计算结果，采用预定的访问策略给予授权。ABAC 可以更方便地实现细颗粒度的授权管理。

B. 个人或者消费者身份管理（Consumer Identity）：

通常个人/消费者用户要求更方便的认证及授权体验，访问负载通常变化也比较大。而且不同于企业用户访问通常有一定的内部网络边界，接入渠道受控（通过局域网 LAN，虚拟私有网络 VPN 加密通道等方式访问），消费者访问一般都是通过互联网传输，这对于应用服务商带来更多的挑战，应用服务商在提升用户体验的同时，还需要兼顾相关的安全性。

HTTPS 在 App 后台为了避免每次验证用户身份都需要传输用户名和密码，一般采用以下机制。App 后台接收到 App 发送的用户名和密码后，验证用户名和密码是否正确。如果错误则返回错误信息。如果 App 后台验证正确，生成一个随机的不重复的 token 字符串。token 字符串作为用户的唯一标识，在 Redis（Remote Dictionary Server)中建立 token 字符串和用户信息的对应关系。App 后台把 token 字符串和用户信息返回给 App，App 保存这些数据作为以后身份验证的必备数据。为了减少 token 被盗产生的安全风险，用户每隔一段时间，需要重新输入用户名及密码，生成新的 token。

提供用户身份验证除了基本的用户名，密码之外，也采用其它身份验证方式，例如短信验证码，指纹，人脸生物特征识别等，通过对于设备等其它相关信息的综合判断，给与登陆用户相应的认证及授权。为了对抗一些暴力破解，还会采取一些用户行为验证（例如：图形选择，数学计算，拼图等）。

应用服务商集中存储大量用户个人信息及相关验证信息。为避免用户隐私泄露，认证信息泄露，用户信息数据库的安全防护，就成为应用服务企业重要的控制领域。除了对服务器数据库的基本防护措施，相关用户存储信息一般需要采用加密，或摘要等方式，避免万一数据库被盗，引发用户个人信息及认证信息泄露的问题。

C. 物联网设备的身份：

由于物联网设备硬件能力相对比较弱，需要采用轻量级的身份认证方式。提供 IOT 服务的云厂商，都有一些物联网身份的解决方案，比如采用 X.509 证书等。目前还缺乏成熟的身份管理系统。

用户与设备的认证协议主要为了实现用户匿名和互认证；设备与服务器的认证协议主要为了实现轻量级认证和隐私保护；设备与设备的认证协议主要为了实现隐私保护、高效认证和移动认证。

1.2 传统的数字身份系统的痛点

传统的数字身份系统有下面一些主要的痛点：

1.中心化数字身份容易遭到黑客进攻，由于黑客一旦成功攻破一个中心化的身份数字系统，就可以或得几千万乃至几个亿的数字身份。去中心化数字身份相对进犯本钱高，因为黑客需要攻破每一个去中心化的身份。

2.中心化数据可能被平台或者第三方不合理使用。例子包括剑桥分析(Cambridge Analytics)师公司对于美国和英国民意的控制，也包括国内在 2019 年一些数据公司因为卖隐私数据被审查事件。

3.目前，我们正在进入数据技术时代（DT：Data Technology），在数据技术数据与工业时代的汽油一样重要。但是，目前的 IT 架构和中心化的数字身份生态对于数据的确权，数据的价值定价，和数据在基于所有权不变的前提下进行公平交易没有好的解决方案。 数据时代，数据私有化是数据是否能够被合理，有效，安全地使用的必要条件。那么，数据私有化本身的必要条件是什么呢？数字

身份私有化就是这个必要条件。没有数字身份的私有化或者说去中心化，公链或者联盟链生态体系的数据很难确权。

4.数据私有化以后，如何鼓励用户有偿地分享数据呢？中心化的架构不可能合理，安全，有效地解决这个分享的问题。这个也需要基于去中心化数字身份下的通证激励机制在身份私有化和数据确权的情况下才能发挥价值。

5.基于用户的信誉的生态系统是包括金融，电子商务，保险，房地产，政务，物流等等产业系统的技术和数据支撑。区块链项目的落地，也需要与之对应的用户的信誉的生态系统。传统的数字身份系统下，用户的信誉存储在中心化的数据库存在被盗用和被篡改的情况。去中心化的数字身份在对于用户的数据进行确权的前提下，可以叠加用户的信誉分数， 在保护隐私的条件下有偿地存储和分享信誉分数，并需要满足各国的隐私法律和监管条例（如欧盟 GDPR， 美国加州 CCPR， 英国 DPA2018 等等）和提供用户能管理他们个人的信息和私隐保护。

第二章：用户自治的数字身份（SSI）和去中心化数字身份（DID）

用户自治的数字身份（SSI：Self Sovereign Identity ）和去中心化数字身份（DID: Decentralized IDentity）没有本质上的区别主要区别是在字面意义和实务应用两个层面。从字面意义的角度，DID 更加强调去中心化特性，更加强调身份系统中每个用户通过标识符实现点对点的交互，没有单独某个或一群节点可以控制所有流程产生的数据，因此 DID 更加侧重于技术的实现方式与系统的架构；SSI 则更加倾向于对用户权利的主张，表达了用户对个人数据隐私保护和对数据的使用具有许可权的诉求。比较 SSI，DID 在国际是更加通用，因此在这篇白皮书，我们会使用 SSI 和 DID 技术术语表示同一个概念。

DID 规范（Specification) 可确保 DID 的发行者和验证者都可以在公共区块链上查找必要的公钥进行验证，而不管他们是否属于同一身份证书颁发机构（CA-Certificate Authority）系统还是同一个身份联邦系统（Federated Identity）。

这种进展与局域网的“网络孤岛”到互联互通的全球互联网相似。从各自具有自己的 PKI 的断开的“身份孤岛”到基于去中心化 PKI（Distributed PKI）的全球身份网络。这样的数字身份系统从对于中心化的身份证书颁发机构（CA）的依赖转变为更具弹性，安全和隐私保护的系统。DID 协议有下面的 3 个重要组成部分，或者说是三大支柱：

第一个支柱：安全连接协议，一种标准的开放协议，用于在两方之间建立唯一，私有和安全的连接，而无需诸如 Google，WhatsApp，电子邮件提供商或电话运营商之类的中间“连接代理”的协助。

安全连接是由两个对等方创建的，它们创建并交换 DID 标识符。W3C (World Wide Web Consortium) 现在正在研究开放的 DID 标准（https://www.w3.org/TR/did-core/）。有“公共” DID 和“私有”或“对等”（Pairwise）DID。公用 DID 可用作“跳出”点，以触发私有 DID 的交换。私有 DID 只能在连接中的两方存储或交换，任何第三方不可见。一旦两方交换了私有 DID，他们就可以安全地进行通信，就像通过一条其他人看不到的专用隧道一样。用户可以为每个数字关系使用不同的 DID 来保护隐私。任何人都可以随时创建 DID，而无需第三方。 这个安全的连接不能提供信任，提供信任需要第二个支柱。

第二个支柱：可验证证书协议，W3C 和 DIF (Decentralized Identity Foundation) 发行了一种标准的开放式“可验证证书（VC - Verifiable Credentials）”协议，用于签发，持有和验证数字证书，例如驾驶执照，会员卡，机票和医疗资格。这样，任何人都可以验证任何数据的来源，完整性和有效性。这种 VC 协议结合了成熟的公钥加密技术对每个数据元素进行数字签名，并且可以增强隐私和避免个人在线数据被关联的可能性。类似于使电子邮件使用 SMTP 和 Internet 使用 TCP / IP 协议，任何人都可以在 VC 协议的基础上进行信任的构建。VC 协议可以参照链接（https://www.w3.org/TR/vc-data-model/）。任何人都可以出于任何目的将任何数据放入可验证的凭证中。在初步应用中，证书的颁发者可以是权威机构，比如大学学位证书可以由大学颁发。信任的建立是基于证书发行者的签名和 VC 协议层的密码学。

第三个支柱：用于存储证书颁发者的公钥的分布式账本或者是 DID 的底层区块链技术和账本。这样，任何人都可以随时查找和检索公钥来验证数据并且验证符合 VC 标准的任何数据的来源，完整性和有效性。这些密钥和其他密码数据保存在 DID 文档和凭证定义中，并固定在证书颁发者的公共 DID 中。

这三个组件的组合定义 DID 的新身份范式。它将改变我们今天所知道的数据格局。这些组件一起提供了一种新的方法，可将数据从 A 可靠地移动到 B，而无需中间人窥探，而且接收者可以验证数据的来源，完整性和有效性。各地的每个人都可以颁发，持有和验证有关任何事物的任何凭证。这意味着不再有专有的数据库“拥有”你的身份。在这种环境中，人与密码的信任相结合意味着你最终可以同时提高安全性和减少摩擦。

2.1 DID 标准介绍

目前在去中心化数字身份的标准制定方面有下面的一些标准组织在做努力，他们之间有一些沟通，但是缺乏统一的行动，因此可能因为不同的应用场景（比如：公链和联盟链的不同的应用）可以预见以后多种标准的共存：

有关单位·	参考资料·
1： DID 的核心架构，数据模型和表示， 版本 1.0 （DIF 和 W3C）	https://www.w3.org/TR/did-core/
2： 欧洲的 SID	https://www.eesc.europa.eu/en/news-media/presentations/european-self-sovereign-identity-framework
3： 基于 IP 的信任网路 （Trust over IP- ToIP 基金会,和 Linux 基金会 ）	https://trustoverip.org/
4： ID2020	https://id2020.org/

5： ERC725	https://erc725alliance.org/
6: 中国公安部的 EID	https://eid.cn/

所有这些标准的共同属性如下：

1. 数字身份的全球唯一性。
2. 用户对于数字身份的可控制性。
3. 身份没有中心化发行或者认证机构。
4. 隐私保护。
5. 基于非对称加密算法。
6. 有一些标准没有规定具体的底层区块链技术（比如 DID 的协议），但是基本都假定底层的技术是基于区块链。具体区块链技术可以用协议的方法比如 DID Method 来规定。

下面我们主要对于其中 3 种标准进行介绍因为他们一个代表北美的标准（DID）也有国内和欧洲的有关机构支持，另外一个主要是欧洲的标准，最后一个是中国公安部的标准（EID）。

2.1.1 W3C 的 DID

基于区块链建立符合 W3C 标准的数字身份系统，为企业、用户提供去中心化的数字身份， 保证数字身份的绝对可控和绝对拥有，解决企业和用户隐私泄漏难题。

DID 是数字身份的基础，其核心是基于区块链去中心化、不可篡改的特性而创建的。多方通过接入工具接入分布式网络，以区块链为依据，建立不同身份标识之间的安全通讯，为可验证声明（一种数字证书）的流转建立前提。

所有用户都可以根据自己的需要，通过任意一个实现了 DID 算法的平台都可以创建新的 DID，是完全由用户自主控制的。生成 DID 的同时，也会生成一对秘

钥，并把公钥与 DID 的绑定关系发布在分布式存储上，而私钥则由用户保管，最后只需把身份相关数据锚定在区块链上即可。在应用方验证用户身份信息时，只需要根据分布式系统中形成共识的用户的公开秘钥，进行验证计算即可验证用户的真实性。

DID 是一个协议标准，主要定义了实现的格式与验证方法。下面为一个基本的 DID 文档的内容：


```
{
    "@context": "https://www.w3.org/ns/did/v1",
    "id": "did:example:123456789abcdefghi",
  "publicKey": [{
      "id": "did:example:123456789abcdefghi#key-1",
      "type": "Ed25519VerificationKey2018",
      "controller": "did:example:123456789abcdefghi",
  "publicKeyBase58":"H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV
"
    }, ...
  ],
  "authentication": [{
      "id": "did:example:123456789abcdefghi#keys-1",
      "type": "Ed25519VerificationKey2018",
      "controller": "did:example:123456789abcdefghi",

"publicKeyBase58":"H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
    }],
  "assertionMethod": [
      "did:example:123456789abcdefghi#keys-1",
      {
        "id": "did:example:123456789abcdefghi#keys-2",
        "type": "Ed25519VerificationKey2018",
```

```
            "controller": "did:example:123456789abcdefghi",
        "publicKeyBase58":"H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV
"
            }
          ],
        "proof": {
            "created": "2020-05-01T03:00:02Z",
            "creator":
"did:example:12D3KooWMHdrzcwpjb********cgqX3b5dpuPtPa9ot69yew;
example:key=id=bafyreicubtx5w*******zrkctfhwd6rewezgpwoe4swirls4ebdhs2i",
            "signatureValue":
"o9r6LxgoGN8FoaeeUA6EdDcv12GvDzFEmCgjWzvpur2YSQyA8W2r0SSWUK+nH5tMq
zaFLun6wwZ1Eot37amGDg==",
            "type": " LinkedDataSignature2015"
          },
          "service": [{
            "id":"did:example:123456789abcdefghi#vcs",
            "type": "VerifiableCredentialService",
            "serviceEndpoint": "https://example.com/vc/"
          }]
        }
```


这里主要分为：id、publicKey 、authentication、assertionMethod、proof、service 等属性。其中：

1、did:example:123456789abcdefghi 就是 DID DID 的完整格式为：did:<method>:<method-specific-id>，DID 的生成是根据 DID 文档的基本信息来加密生成的其中 <method-specific-id>=base58(ripemd160(sha256(<Base DID Document>)))；pubickey, DID 文档可以表达加密密钥和其他验证方法这些方法可以用于验证或授权与 DID 主题或关联方的交互。所表达的信息通常包括全局明确

的标识符和公钥材料，可用于验证数字签名。可以表示其他信息，例如密钥的状态信息（例如，密钥是否被挂起或吊销），或者其他属性，使人们可以确定它是否是硬件支持的加密密钥。关于加密密钥材料，可以根据其用途将公共密钥包含在 DID 文档中，例如，使用 publicKey 或身份验证属性。每个公共密钥都有其自己的标识符（id），类型和控制器，以及取决于其类型的其他属性。

2、authentication，身份验证是一种机制，通过该机制，实体可以证明自己是 DID 主体。 身份验证尝试的验证者可以检查身份验证方是否正在提供有效的身份验证证明，即他们是他们所说的身份。authentication 属性是 DID 主题和一组验证方法（例如但不限于公钥）之间的关系。 这意味着 DID 主题已出于身份验证目的（根据 authentication 属性的值）授权了一些验证方法集。 authentication 属性的值应该是一组验证方法。由 DID 文档的 authentication 属性指示的验证方法只能用于认证 DID 主体。为了认证 DID 控制器(在 DID 控制器不是 DID 主体的情况下),与控制器的值相关联的实体(参见§7.5 授权和委托)需要使用自己的 DID 文档和所附的 authentication 验证方法关系来认证自己。除了 authentication，其他如 capabilityInvocation, capabilityDelegation, keyAgreement 和 assertionMethod 也可以进行验证。

3、assertionMethod 属性也用于表示验证关系，该关系指示验证方法可以用于代表 DID 主体断言一个语句。判断方法尝试的验证者可以通过检查与该证明一起使用的验证方法是否包含在 DID 文档的判断方法中，来检查代表 DID 主体断言的陈述的证明。

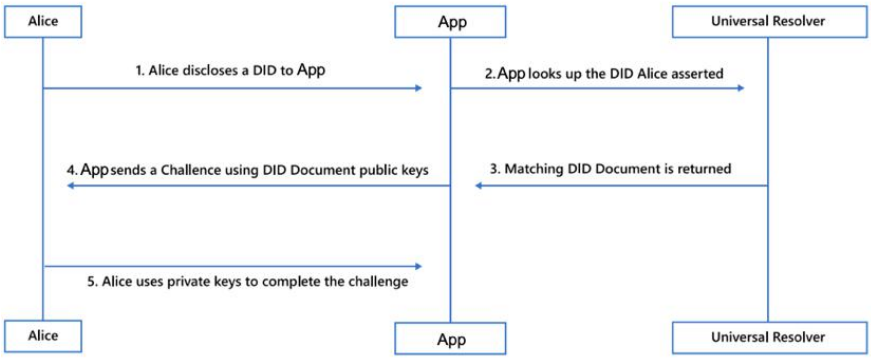
proof 属性，则该属性的值务必(MUST)是有效的 JSON-LD (LD-LinkData)证明，如链接数据证明所定义，通过使用链接数据证明[LD-PROOFS]，链接数据签名[LD-SIGNATURES]和各种签名套件，可以实现对多种类型的密码证明格式的支持。

4、service 属性，在 DID 文档中使用服务端点来表达与 DID 主题或关联实体进行通信的方式。DID 文档中列出的服务可以包含有关保护隐私的邮件服务(privacy preserving messaging)的信息，或者包含更多公共信息，例如社交媒体帐户，个人网站和电子邮件地址。与服务关联的元数据通常是特定于服务的。例如，与加密

的消息传递服务关联的元数据可以表示在消息传递开始之前如何启动加密的链接。

使用 service 属性表示服务的指针。 每个服务都有自己的 id 和 type，以及带有 URI 或其他描述服务的属性的 serviceEndpoint。

其中，利用公钥的验证应用流程示例如下，



用户在 App 上使用 DID 时得证明自己是 DID 的所有者，主要运用的机制是挑战-响应机制：App 首先根据用户提供的 DID 用从 DID Resolver 查到对应的 DID Document，然后 App 使用 DID Document 中的公钥加密自己随机生成的一串 nonce，发送给用户，用户用自己的私钥解密后得到这串 nonce，把 nonce 发送给 App 完成挑战。

2.1.2 欧洲的 SID

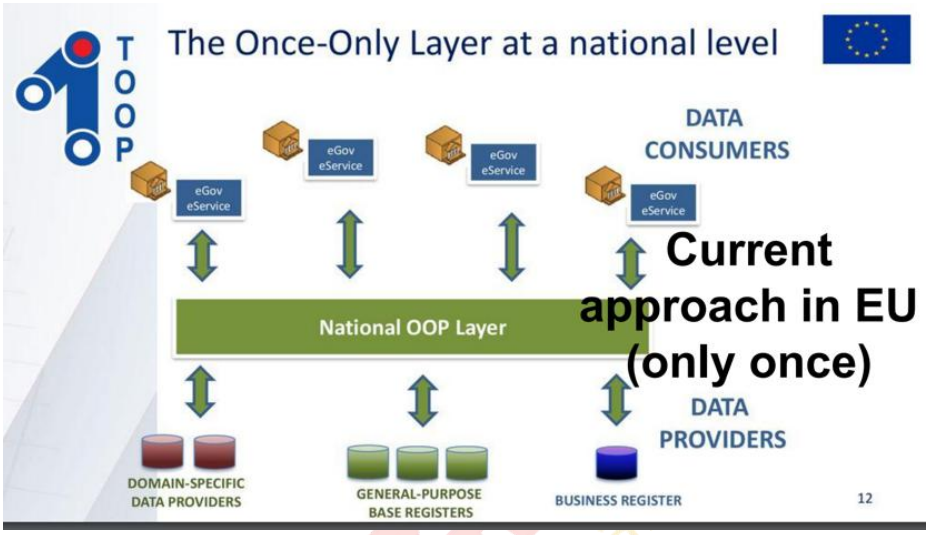
欧洲经济和社会委员会于 2019 年组织发表了 ESSIF-European Self Sovereign identity framework （简称 SID）的标准。SID 表明于对于数字身份在数字化生活与工作中都具有非常重要的作用，包括但不限于学历真假、难民身份、贷款或补贴、判断年龄、客户体验、地方身份、当前公私机构等。

传统的身份认证方式具有以下问题需要解决掉：

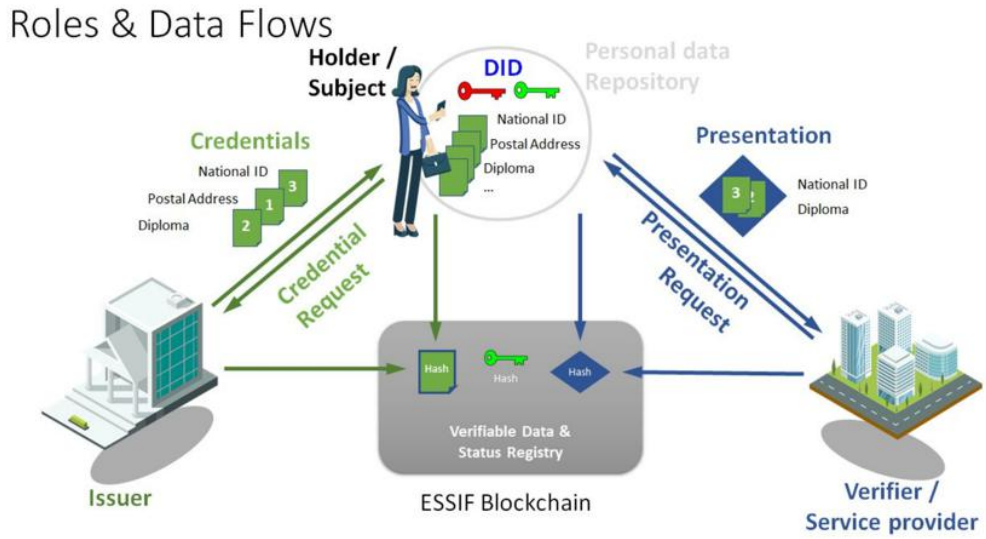
- 纸质文件的重复提交与审核需要浪费很多时间与金钱；
- 验证的结果不能进行有效共享，机构需要对纸质文件的真实性需要进行验证；

- 相同的信息需要向多个不同的机构进行提交，不同的机构也要进行相同的检查；
- 所有验证都是手工检查，不能进行自动化验证；
- 所有提交出为的信息，将不再由自己进行控制，容易导致滥用。

因此，EESC 提出了一个一次性认证的 SID 关系架构图，如下：



为了解决现有身份认证的问题，提出来自我主权的身份设计。自我主权的身份是超越以用户为中心的身份的下一步：用户必须是身份管理的中心。基于这个观点，SID 提出了采用区块链的技术来实现的数据流程，如下图：



- 用户的私有数据由用户自己存储，包括用户的公私钥、身份证、邮编、学历等；
- 用户向对应的验证机构进行验证，验证结果保存在区块链系统中；
- 数据使用方向用户申请获得用户的身份对应信息的凭证；
- 数据使用方使用用户的公钥解密用户的凭证后，与链上数据进行对比，如果一致即可验证用户的身份。

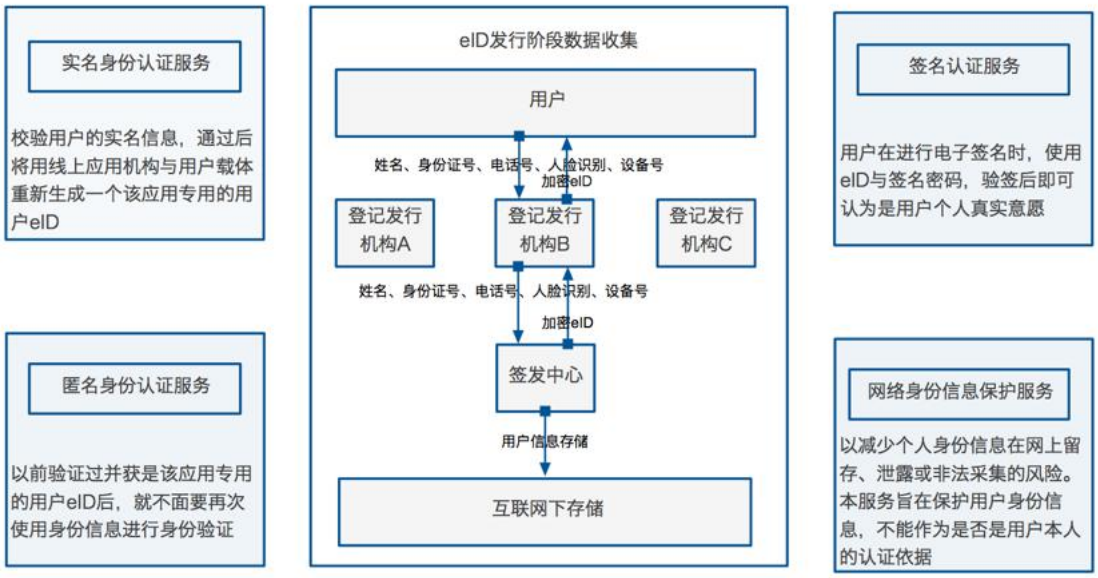
2.1.3 中国公安部的 EID

eID 是以国产自主密码技术为基础、以智能安全芯片为载体，采用空中开通或临柜面审的方式，依据对法定身份证件核验的结果，由“公民网络身份识别系统”签发给公民的网络电子身份标识，不仅能够在不泄露身份信息的前提下在线识别自然人主体，还能用于线下身份证明。

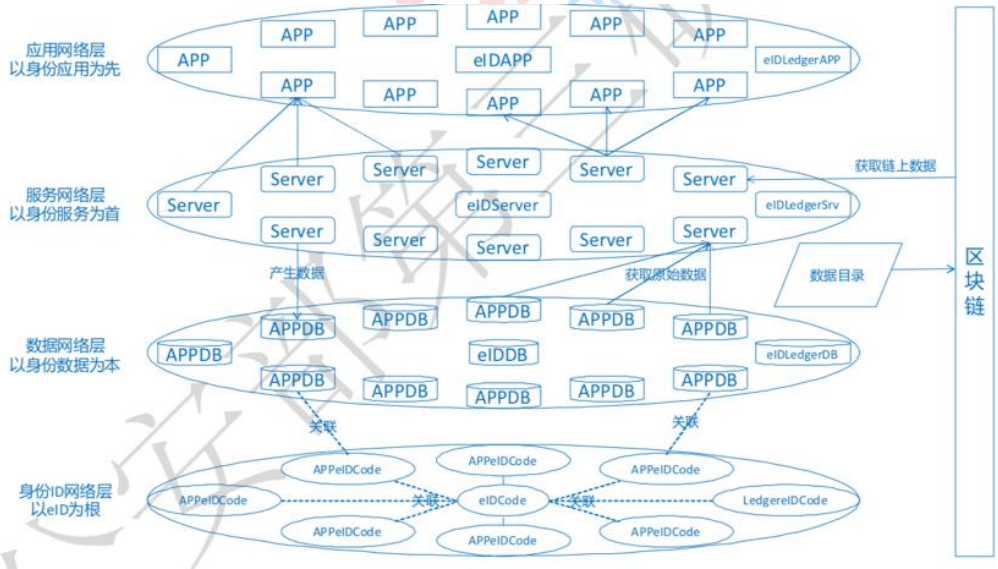
公安部第三研究所于十二五期间承担了国家 863 计划“网域空间身份管理”等信息安全重大专项，研发了“网络电子身份标识（eID）”技术并形成了相关标准体系。并于 2018 年，从我国数字身份发展的突出需求出发，结合以上科研成果和近年在该领域的实践和理论研究，推出《eID 数字身份体系白皮书（2018）》。在 2019 年 5 月，推出了 eID 数字身份链，eID 数字身份链是以 eID 数字身份为索引的基础链，在此基础之上自下而上形成 eID 数字身份网络、数据网络、服务网络、应用网络，面向政务、企业、社会、行业等提供身份证明、声誉证明、资产保管、支付记账、数据存证、数据管理等服务。eID 数字身份链继承了 eID 和区块链的特点，具有权威、中立、多方、普适、可控、隐私等特点，有力支撑了 eID 数字身份生态的建设。

在发行与服务流程上，通过泛用户与应用场景、多登记发行机构，中心化签发中心、内网存储原始数据的方式，再结合实名身份认证服务、签名认证服务、匿名身份认证服务，网络身份信息保护服务等方式为各应用提供服务。

eID 在发行阶段与服务阶段的说明，如下：



在技术分层架构上，依赖 eID 数字身份链，在四个层次上构建了相互关联的网络，分别为:eID 数字身份网络、数据网络、服务网络、应用网络。 数字身份链技术框架图如下：



在功能上，eID 具有在线身份认证、签名验签和线下身份证明等功能，能够在保护公民个人信息安全的前提下准确识别自然人主体身份，可以运用在网上签约授权、交易支付、航旅服务、酒店住宿等多种场景。目前已经有一些角行、数据交易、政务等应用在使用。

总结表

	W3C 的 DID	欧洲的 SID	中国公安部的 EID
1：数字身份的全球唯一性。	是	是	是
2：用户对于数字身份的可控制性。	是	是	是
3：身份没有中心化发行或者认证机构。	分布发行	分布发行	需要中心化发行
4：隐私保护。	可以	可以	可以
5：基于非对称加密算法。	是	是	是
6：底层区块链技术	是	是	是

2.2 DID 代表性项目介绍

下面主要介绍 5 个具有代表性的 DID 项目。元界 DNA 的 DID 数字身份是第一个在链上实现数字身份的区块链项目，其他项目把数字身份作为第二层协议实现，主要原因是在区块链设计过程中没有认识到去中心化数字身份的重要性。例如以太坊区块链就没有在链上实现的原生的区块链，必须在第二层进行拓展才能实现。Civic 数字身份是美国的一个非常有影响力的早期项目，值得参考。Evernym 与 Linux 基金会合作把 Evernym 的数字身份项目贡献给 HyperLedger。Uport 是基于以太坊的数字身份项目。微软代表着大公司对于去中心化数字身份的努力。

2.2.1 元界 DNA 的 DID 数字身份

元界 DNA 的 DID 数字身份又称呼为“Avatar”，是全球第一个在链上实现的原生的 DID 数字身份。用户在元界 DNA 的客户端或者钱包可以建立完全可控制的身份，这就意味着不必依赖中心化实体或第三方进行身份验证。用户拥有真正意义上的自主身份，可以创建、签署、验证，同时与用户进行交互的人也能够证明其身份。此外，这些拥有自主数字身份的用户能够选择性地披露他们的信息。元界 DNA 认为数字身份是虚拟世界不可分割的一部分。就其本身而言，数字身

份可以采取任一形式，如个人或价值中介（机构和实体）。因此，个人在不同的场所可以拥有不同的数字身份，如职场身份和家庭身份，但这些最终都是以用户的现实身份为基础。用户可以通过数字身份在元界 DNA 生态系统中建立自己的声誉，同时这也会改进交换价值的方式。它可以通过数字签名、验证要求和交易来实现这一目的，并逐步树立起可以被市场上其他数字身份和价值中介（Oracle）检查和验证的声誉基础。对于一些中心化实体，如果他们的服务器崩溃，那么它们多年树立的身份和声誉将永远消失。元界 DNA 则不同，用户的数字身份及其声誉将受到区块链的保护。

元界提出数字资产、数字身份和价值中介这三大核心要素，建立一个具备智能属性的网络，从而为元界上的所有去中心化应用提供协议级的支持。

数字身份

数字身份作为用户拥有的主私钥所对应账户的 Profile 信息的统称，主要包括了 Oracle 角色以及普通用户角色，任何数字身份都可以作为 Oracle 和普通用户参与到数据身份的应用中。Profile 拥有一个全网唯一标识（DID），其主要包含以下信息：个人交易记录、资产信息、自定义描述字段。其操作流程如下：

1. 创建：任何用户都可以创建数字身份，并与自己的主私钥绑定。如果创建完成后并未绑定任何主私钥，那么该 DID 相当于一个未经认证的账号，无法使用数字身份的任何功能和应用。

2. 验证：Profile 能够提供有效的证明链用来证明该数字身份下的客观事实。

3. 授权：通过触发脚本去验证目标账户数字身份信息中的资产信息后方提供服务。

4. 查询：使用 DID 标识作为在场外交易的主体，可通过该 DID 标识在交易市场中查询交易请求以及历史交易记录。

数字身份与主私钥是一对多的关系，一个数字身份可以对应多个主私钥；数字身份与资产是一对多的关系。数字身份不可转移和销毁，但其对应用户的现实中的关系是可以变更的。

链下数据及资产管理

链下数据指的是没有记录在区块链上的数据，其具有数据结构复杂、数据体量庞大的特点，元界通过以下步骤将数字身份与其相关联的有效链下数据进行登记：

1. 拥有数据的用户在元界上建立其数字身份，并提供其自定义格式的数据，将数据打包提交给负责数据鉴定与背书的 Oracle；
2. 具备鉴定数据资格的 Oracle（数字身份的一种）鉴定该数字身份所提交数据的有效性、真实性；
3. 经数据所有者和 Oracle 签名后的资产将通过主私钥与数字身份绑定；
4. 其他用户可以在数据所有者的授权下，查看该数据资产所代表的详细信息。

下面的图介绍数字身份与价值中介 Oracle 在数据管理方面的作用。

应用管理

数字身份可以解决账户无法跨平台使用和账户资产无法跨平台流通的痛点。应用平台在元界中注册为数字身份并在自己的数字身份上定义出标识符，并使用该数字身份关联相应的主私钥，将元界钱包服务配置进气应用中。应用平台的用户在元界上注册数字身份，用户使用该数字身份在各应用平台登录时，可选择性的将身份信息授权给应用平台，而无需再次注册和认证身份信息。用户只要使用

该数字身份即可在各应用平台间进行访问，而无需再次注册。因数字身份不隶属于任何一个中心化的应用平台，故无需担心数字身份被删除、泄露或者篡改。用户可以选择性的将自己的数字身份中的绑定信息授权给其他应用平台，未经授权的信息，平台则无法读取。下面的图介绍 Oracle 角色和用户角色下的数字身份。

数字身份与 BaaS（Blockchain as a Service）

元界支持基于公链的 BaaS 概念，即企业或个人可以根据自己的实际需求，向区块链解决方案供应商定制区块链服务。BaaS 服务主要包括：1、基于数字身份进行对象管理；2、对链上数据进行深度挖掘和检测；3、提供一体化的资产管理底层服务；4、为关联交易方提供透明的信息授权。下面的图总结元界的 BaaS 概念.

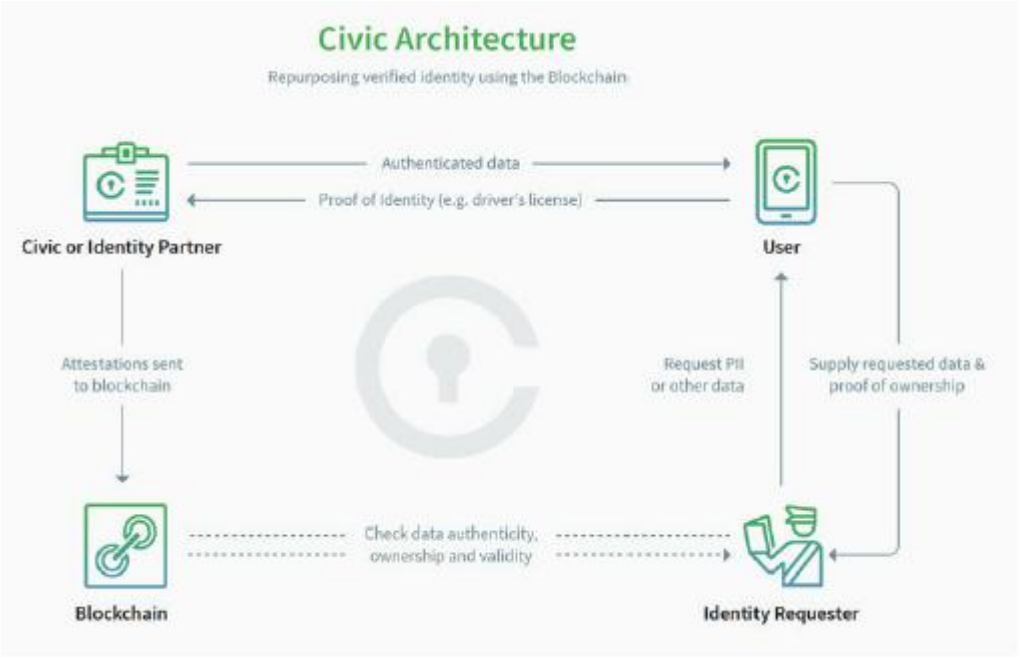
2.2.2 Civic 数字身份项目

Civic 是一个全新的基于区块链技术的身份验证平台，它为用户提供了按需定义、安全和低成本的数字身份隐私服务。Civic 允许人们控制其身份信息的使用，使用区块链身份验证技术来保障和保护个人信息传输，身份验证服务应该安全、可获得并且按需提供。

Civic 使用区块链和生物识别技术的分散式架构（区块链 & 生物识别 & 数字身份），搭建安全身份平台（SIP-Secure Identity Platform），提供多因素身份验证，无需用户名，密码，第三方身份验证器或物理硬件令牌。

- 由第三方钱包生成密钥；
- 在用户设备上的应用程序中的身份数据完全加密，只能使用生物识别技术进行访问
- 验证授权机构的公共标识符，散列标识数据和指示数据的标志仍然有效

身份数据的真正所有者可以使用 SPS（安全私人注册）创建新帐户。同时，用户可以在没有用户名或密码的情况下使用 Civic 的 SPL-Secure Personal（安全私人登录）登录网络和移动应用程序。SPL 提供更好的用户体验和多因素身份验证，无需使用弱密码，有效规避密码重置漏洞或传统双因素身份验证的麻烦。下面是 Civic 的架构图。



Civic 身份体系有如下优势：

1. 没有特定软件或基础设施：因 Civic 使用公共区块链，故身份请求者不必投入大量资金来建立技术基础设施来支持 Civic 安全身份平台解决方案。
2. 数据可撤销：身份数据可由身份验证机构撤销。
3. Civic 不存储用户数据：身份数据被加密并存储在用户移动设备（主要是手机）上的 Civic 应用程序中。
4. 全球保护：使用第三方认证的身份数据，Civic 不能被外国政府或犯罪组织强迫使身份数据失效。
5. 全球兼容：用户可以在世界任何地方存储和分享自己的身份，他们的数据可以在全世界任何地方被访问。
6. 安全：Civic 利用区块链的强大功能确保为企业提供最高质量的隐私和安全。
7. 身份盗窃保护：Civic 通过身份盗窃监控、身份监控报警、欺诈支持、身份盗窃保险等措施提供事件之前、期间和之后完整的身份盗窃保护。

2.2.3 Evernym 数字身份项目

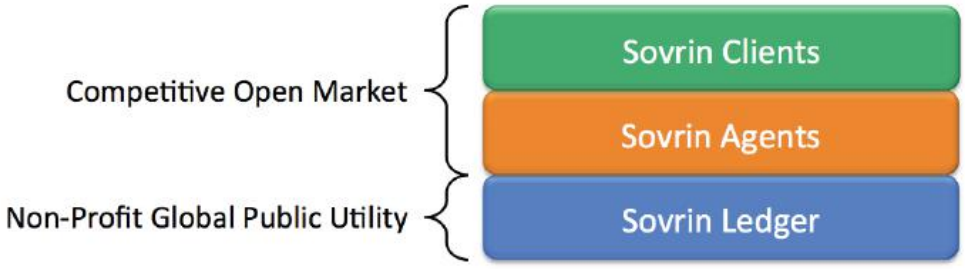
Sovrin 是 Evernym 的旗舰产品，项目的代码已捐赠给 Hyperledger。Sovrin 的身份系统可以应用于许多不同的应用程序。Sovrin 的目标是用安全的通用协议来取代当今的中介，从而简化交互并增强信任度。

Sovrin 的一个关键特征是它的“自我主权身份”，即 SSID，它是一个由个人或组织 100%拥有和控制的身份。未经主人明确同意，任何人不得阅读、使用、关闭或带走。这是一种简单的身份形式，是私人的，安全的。这将使个人、机构和企业之间发生可信的互动。

使用 Sovrin，医疗行业的成员将能够直接、有效地沟通，减少责任，并得到患者的立即同意。同时，政府可以使用 Sovrin 为土地记录、财产所有权、出生登记、疫苗接种、难民身份识别等提供数字身份验证。

Sovrin 架构

作为 Internet 的标识层，Sovrin 可以看作是一个三层软件堆栈，如下图所示：



1. Sovrin 分类账本是一个基础的组成部分，是一个全球分布的根身份记录分类账本，由全世界受信任的机构维护；

2. Sovrin 代理是一种新型的网络服务，它为 Sovrin 身份所有者（个人和组织）提供一种永久的、隐私保护的方式来执行身份和数据管理事务。Sovrin 代理并不是 Sovrin 体系结构所严格要求的，它们只是使 Sovrin 身份更容易、更有效地使用；

3. Sovrin 客户端是 Sovrin 身份所有者（通常在本地设备上如智能手机、笔记本电脑等）使用的应用程序，用户可以使用该客户端与 Sovrin 代理商

和 Sovrin 账本进行沟通，以进行各种身份交易。从安全和加密的角度来看，Sovrin 客户端是 Sovrin 密钥管理的“密钥”。

Plenum 共识协议

RBFT- Redundant Byzantine Fault Tolerance（冗余拜占庭容错）在早期 PBFT-Plenum Byzantine Fault Tolerance 和 Aardvark 协议的基础上进行了改进，在不考虑系统以前或将来的性能/条件的情况下，并行执行多个具有不同主验证节点的协议实例，实时检测任何性能问题。Plenum 协议在如下几个方面对 RBFT 协议进行了改进：

1. 用于节点间通信的数字签名（RBFT 使用 MAC 认证器，其速度更快，但不支持不可抵赖性）；
2. 只向 f+1 节点分发请求（f 是故障节点的数目）；
3. 主验证节点的两种选择机制（一种是确定性的，另一种是非确定性的）；
4. gossip protocol（允许集体共识在部分分区的网络中更快地进行）；
5. 多个明确的黑名单策略（Plenum 考虑故障的严重性并应用适当的黑名单策略）；
6. catch-up mechanism（用于新的或崩溃/恢复的节点有效而安全地重新获得完整状态）。

验证和观察节点

验证节点运行 Plenum 共识协议来验证新的 Sovrin 事务，对于 Sovrin 分类账本中的每个“写入”都必须发送到验证节点进行验证。在 Sovrin 网络的早期，所有节点都是验证节点，因为它们需要处理网络的读写负载。随着网络规模的扩大，逐步引入了观察节点。从 Sovrin 客户端的角度来看，观察节点只是 Sovrin 分类账本的只读副本。观察节点主要实现以下三个功能：

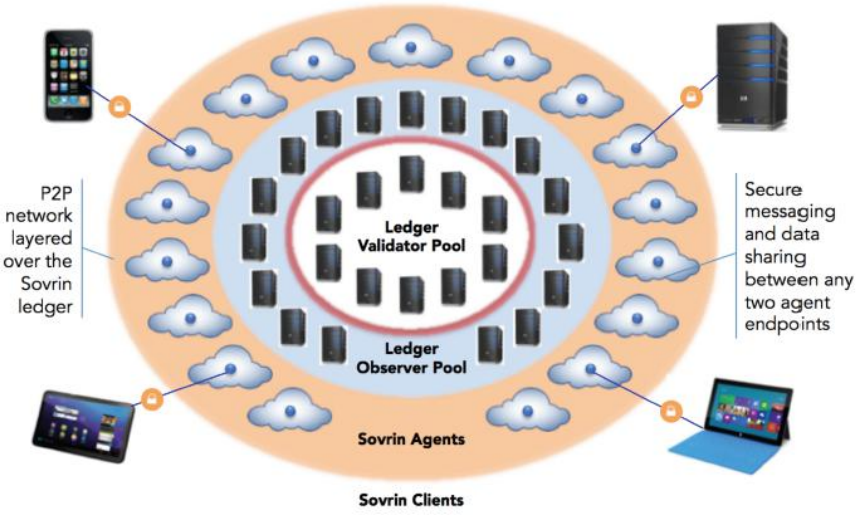
1. 分流读取请求：与 DNS、LDAP 和其他大型标识系统一样，读请求通常比写请求多出一个数量级。Sovrin 观察节点使 Sovrin 身份记录的需求能够在不影响运行 Plenum 共识协议的验证节点性能的情况下进行扩展；

2. 热备份：由于观察节点也由 Sovrin stewards 操作，因此如果另一个验证程序节点出现故障或受损，它们可以作为活动验证程序节点进行替换；

3. 推送订阅：不同的参与者可能希望订阅特定的 Sovrin 分类账事件。观察节点提供了一种将事件通知推送到没有增加验证器节点负载的订阅服务器的能力。

Sovrin 代理

Sovrin 代理构成了 Sovrin 体系结构的“中间层”，如下图所示



Sovrin 代理是 Sovrin 网络的 P2P 端点。Sovrin 代理和 Sovrin 客户端都充当 Sovrin 分类账的客户端，其主要的区别在于，Sovrin 代理不仅仅是一个客户端；它还是一个服务器，或者至少是一个具有可寻址网络端点的服务器进程。通常，此端点将具有与其他关键网络基础设施（路由器、DNS、电子邮件等）相同的高可用性。代理在 Sovrin 网络中提供以下核心功能：

1. 持久的 P2P 消息传递终结点；

2. 多个客户端的协调终结点；

3. 加密的 Sovrin 密钥环备份；

4. 加密数据存储和共享。

需要注意的是 Sovrin 体系结构并不严格要求具有 Sovrin 代理，任何 Sovrin 客户端都可以直接与 Sovrin 分类账进行交互。在某些情况下建议这样做是为了确保特定交易的完整性或验证 Sovrin 代理的完整性。

Sovrin 客户端

Sovrin 客户端是 Sovrin 架构的“最后一英里”，其实际控制权掌握在身份所有者的手中。Sovrin 客户端在其体系架构中从如下几方面进行了保证：

1. Keychains：Sovrin 身份所有者的私钥作为最重要的单一数据资产，需要 Sovrin 客户端进行管理和保护。为了维护身份所有者 Sovrin 交易的安全和隐私，每个 Sovrin 客户都必须保留所有者 Sovrin 密钥链的全部或至少一部分的副本。

2. 本地容器：Sovrin 客户端维护着身份所有者 Sovrin 数据容器的全部或部分本地副本，故需要其管理特定操作系统来讲这些数据安全地在物理存储中存储。

3. 首次配置：Sovrin 客户端必须连接到网络进行网络身份验证之后方可使用。若身份所有者还没有 Sovrin 身份，则需要在提供第一个 Sovrin 客户端时连接到名为信任锚点的现有 Sovrin 身份所有者并进行信任关系确认后方可注册。一旦身份所有者同时拥有一个 Sovrin 身份和一个 Sovrin 代理，Sovrin 客户端就准备好开始创建连接和共享数据。

4. 配置额外的 Sovrin 客户端：身份所有者可以在其所有设备、应用程序、服务和网络中使用数字身份。身份所有者在新的 Sovrin 客户端和所有者的当前 Sovrin 代理之间执行身份验证过程。然后批准从所有者的现有配置的 Sovrin 客户端之一添加新的 Sovrin 客户端即可。

2.2.4 uPort 数字身份项目

uPort 是一个安全、易用的自主身份识别系统，致力于创建一个由用户控制的，基于区块链的数字身份，用户可以基于不同情况授予或废除对其信息的访问权。uPort 将身份归还给个人，其开放式身份系统允许用户在以太坊上注册自己的身份、发送和查询证书、签署交易以及安全地管理密钥和数据。

uPort 身份

uPort 身份完全由创建者拥有和控制，并且不依赖于集中的第三方进行创建或验证。身份可以通过加密方式链接到链外数据存储。每个标识都能够存储属性化数据 blob 的散列，与该标识关联的所有数据都安全地存储在该散列上。由于它们可以与区块链交互，因此 uPort 身份还可以控制数字承载资产，如加密货币或其他标记化资产。

uPort 标识的核心是 uPort 标识符，它是一个 20 字节的十六进制字符串，充当全局唯一的持久标识符。此标识符定义为以太坊智能合约（称为代理合约）的地址。代理合约可以中继交易，正是通过这种机制，身份得以与以太坊区块链上的其他智能合约进行交互。

智能合约

当用户想要与特定的应用程序智能合约交互时，可以通过代理合约、包含主访问控制逻辑的控制器合约发送事务。代理合约将此事务转发到应用程序智能合约。

使用代理合约作为核心标识符的目的是允许用户在维护持久标识符的同时替换其私钥。如果用户的 uPort 标识符是与其私钥相对应的公钥，并且用户丢失了存储私钥的设备，那他们将失去对其标识符的控制。

在设备丢失的情况下，控制器合约可以通过维护的恢复委托列表帮助 uPort 用户恢复其身份，这些代表可以是个人或者机构。可以通过仲裁委托使用户恢复其身份并将其连接到新设备。

uPort 通过代理合约及控制器合约希望实现用户能够在不更改其核心 uPort 标示符（与信誉、资产和历史等相关）的情况下更新器智能合约逻辑。其中控制器合约维护核心访问控制功能，允许用户对代理合约使用其私钥进行身份验证。

隐私保护

uPort 系统支持选择性公开或者选择与谁共享经过用户允许的数据，该数据在默认情况下进行了加密保护。通过此方式可以增加用户数据的隐私保护。

系统依赖于每个具有公共加密密钥的 uPort 标识。若用户需要对属性进行共享，则可以使用对称密钥对属性进行加密，然后使用允许读取此属性的标识的公钥对该对称密钥进行单独加密。

移动应用程序

移动应用程序是最终用户与其 uPort 交互的方式，也是管理用户私钥的主要手段。其主要思想是将用户密钥保存在其设备的安全存储中，并在使用密钥进行签名时通过本地生物认证进行访问。密钥保留在设备上，无法从设备导出私钥，从而保证了密钥的安全性。

2.2.5 微软的 DID 数字身份项目

微软的 DID 方法的实现是基于身份覆盖网络（ION，Identity Overlay Network）。ION 是一个基于比特币的双层网络，通过侧树（Sidetree）协议访问比特币网络。ION 通过在第二层网络进行批量合并的方式，将大量 DID 操作合并成一个上链操作，并将数据存于 IPFS，另外将数据的哈希存在比特币网络上的方式，从而实现 DID 数据的不可篡改和可信存储。ION 规避了比特币网络的性能问题，可以支持比较好的数据吞吐量。微软的 Microsoft Authenticator App 的使用也是 Microsoft DID 方案的一部分。微软在 2020 年 6 月表示，用于管理 DID 加密密钥的 Microsoft Authenticator App 代码将作为开放源代码发布。此外，Microsoft 表示发布与 Azure 服务一起使用的 Verifiable Credential 软件开发工具包，作为开源代码。

微软基于 W3C 的 DID 标准为个人和组织构建一套开放的、可信任的、可互操作的解决方案，使用户能够更好的控制自身的数字身份和数据。该系统主要由用户标示符，用于管理身份标识、加密等密钥的用户代理，用户控制的数据存储等 7 部分组成。

W3C 标准去中心化身份标识（DIDs）：W3C 标准去中心化身份标识作为独立于任何组织或政府的 ID 用户创建、拥有和控制。DID 是与分散公钥基础设施（DPKI）元数据链接的全局唯一标识符，该元数据由包含公钥材料、身份验证描述符和服务端点的 JSON 文档组成。

去中心化系统（例如，区块链和分类账）：DID 基于为 DPKI 提供所需机制和特性的去中心化系统。

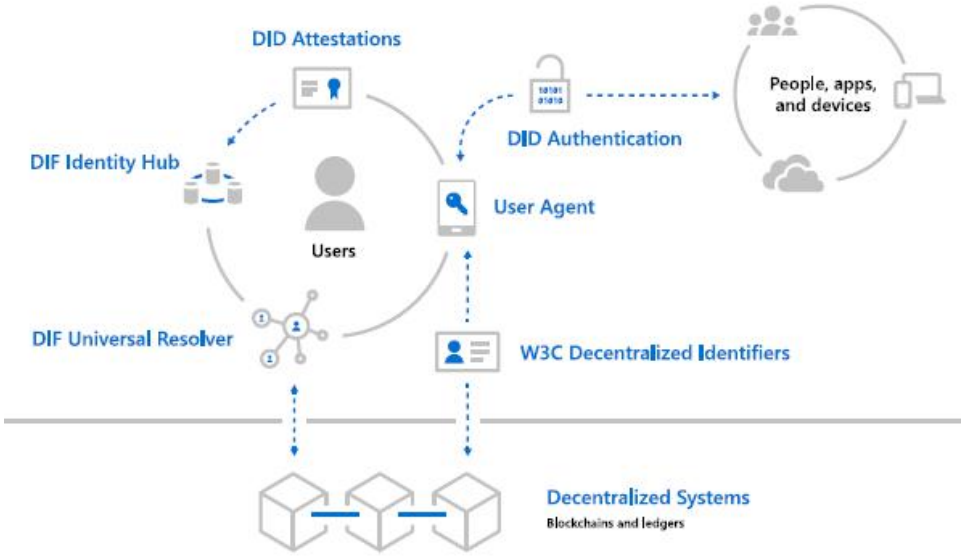
DID 用户代理：微软提供一款类似钱包的应用程序，用于充当管理 DID 和相关数据的用户代理，具有创建标示符、身份验证、数据加密以及密钥和权限管理等功能。

DIF 通用解析器：DIF 通用解析器为 DID 跨系统提供标准的查找和解析方法，并返回封装了与 DID 相关联的 DPKI 元数据的 DID 文档对象（DDO）。

DIF 身份中心：一个由云和边缘实例（如移动电话、PC 或智能扬声器）组成的支持个人数据加密存储的网络，可用于身份数据存储和身份交互。

DID 认证：基于标准格式和协议，使身份所有者能够生成、呈现和验证声明。这构成了系统用户之间信任的基础。

去中心化应用程序和服务：DID 与身份中心个人数据存储结合，可以创建一类新的应用程序和服务。它们使用用户的身份中心存储数据，并在授予它们的权限范围内操作。下面是微软的 DID 架构图。



DID 创建

用户可以通过下载 DID 用户代理应用程序来获取 DID。DID 用户代理也可以帮助用户管理 DID 创建标识符、身份验证、数据加密以及密钥和权限管理的所有方面。微软认为 DID 的实现应该严格使用去中心化系统来标定标识符和非 PII DPKI 元数据，以便为 DID 所有者启用路由和身份验证，从而保证不存在审查风险。用户的实际身份数据在由用户以加密的“链外”方式进行存储。用户可以根据 DID 实现的协议选择用户代理应用程序是否生成设备密钥。

DIDs 的查找与发现

用户代理应用程序是否与 DIF 通用解析程序实例通信以查找 DID。当一个 DID 被传递给通用解析器时，解析器使用适当的驱动程序与分散的系统接口并检索匹配的 DID 文档。它允许应用和服务确定地生成 did 的通用目录。

在 DIDs 之间建立信任

因 DID 创建之初仅代表这一个空洞的身份，只有所有者才能证明其拥有的 DID。为了获得合法性，DID 需要现有的信任提供者（如企业、教育机构和政府）和流程进行背书。该系统提供了一种创建证明的机制，其中包括了谁签发以及何时签发的独立验证。通过从多个信任系统中积累这些证明，DID 可以随着时间的推移建立逐渐建立起信任，以匹配其访问的应用程序或服务所固有的风险水平。

与电子邮件标识符和其他基于账户的系统不同，DID 是自有的且与加密密钥绑定，并基于维护 DPKI 操作的去中心化系统。系统基于此可支持进一步的身份活动，如创建和验证已签名的 DID 证明。证明是一个或多个 DID 用它们的密钥签名以生成关于另一个 DID 可独立验证的声明。数据的时间状态可以通过区块链账本进行记录，并在不信任其他实体或组织记录发生时间的情况下进行独立验证。

DIDs 信息披露与认证

用户可以使用 DID 与其他人、应用程序或服务进行交互（其中包含身份验证）。

1. DID 所有者向第三方提供 DID；
2. 第三方通过 DIF 通用解析器查找 DID，解析器返回匹配的 DPKI 元数据；
3. 第三方通过使用 DPKI 元数据中的公钥引用生成挑战，并与用户执行握手流程；
4. 用户若能够完成挑战-响应握手，则可以证明该用户是该 DID 的所有者。

使用存储的个人数据进行 DID 交互

DIF 身份中心基于用户控制的、链外的个人数据存储，旨在支持广泛的身份交互，为无服务器、提供者无关的、分散的应用程序提供基础。用户可以通过 DID 用户代理应用程序，决定与谁共享数据以及共享数据的粒度级别。身份中心是一个多实例的个人网络，其中的数据使用边缘加密以及用户许可来确保用户隐私。

管理数据访问权限

身份中心主机用于存储和消息转发，其无法解密用户数据，用户可以通过任何实体进行登录并撤销或者删除加密的数据。用户可以使用 DID 密钥对身份中心主机中存储的数据进行加密，并对权限进行签名。

数据同步与复制

DIF 身份中心的一个关键特性是用户可以跨提供者和基础设施边界利用多个实例来同步和复制数据以实现共享，但不需要使用标识中心的提供者。从而确保身份数据不绑定到任何组织，以实现支持分散化、自主权和用户控制的目的。

恢复 DID

用户作为数字身份的真正拥有者，可以使用可靠的设备来恢复其自身的DID。

DID 代表性项目总结表

	元界 DNA DID 数字身份项目	Civic 数字身份项目	Evernym 数字身份项目	uPort 数字身份项目	微软 DID 数字身份项目
1：链上原生	是	是	第二层次	第二层次	第二层次
2：数字身份的全球唯一性。	是	是	是	是	是
3：用户对于数字身份的可控制性。	是	是	是	是	是
4：身份没有中心化发行或者认证机构。	是	是	是	是	是
5：隐私保护。	是	是	是	是	是
6：基于非对称加密算法。	是	是	是	是	是

第三章：DID 安全与隐私考虑

因为 W3C 和 DIF 的 DID 标准相对成熟，我们主要参考这个标准来描述安全和隐私的考虑和解决方案或者说应对措施。DID 标准由于规定必须用户控制数据和身份信息，在隐私和安全比传统的数字身份有许多改进。例如没有单点的风险，也没有中心化的拖库的可能性。但是还是需要考虑新的安全和隐私。这一章就介绍 DID 需要考虑的新的隐私和安全。

先介绍一下需要用到的技术术语：

DID 标识符

全球唯一的标识符，不需要中心化的注册机构，因为它是通过分布式帐本技术（DLT）或其他形式的分布式网络进行注册的。

DID 主体（Subject）

DID 文档（Document）所涉及的实体。即由 DID 标识符和 DID 文档描述的实体。

DID 控制者（Controller）

能够更改 DID 文档的实体。一个 DID 可以有多个控制者。 DID 控制者由 DID 文档顶层的 Controller 属性表示。需要注意的是 DID 控制者可能包括 DID 主体。

DID 文档 （Document）

描述 DID 主体的一组数据，包括 DID 主体可以用来认证自己并证明其与 DID 关联的机制，例如公钥和伪匿名生物验证数据。 DID 文档可能还包含描述主体的其他属性或声明。DID 文档是基于图形的数据结构，通常使用 JSON-LD 表示，但是可以使用其他兼容的基于图形的数据格式表示。

DID 方法（Method）

有关如何在特定的 DLT 或者区块链网络上实现特定的 DID 方案的定义，包括 DID 的创建（Create），读取（Read），更新（Update）和删除（Delete）或者

统称为 CRUD 方法。典型的例子包括 uPort, Soverin，Hedera 等等。详细请见：https://w3c-ccg.github.io/did-method-registry/

DID 读取函数（Resolution）

该函数把 DID 的标识符和一组元数据作为输入，去匹配对应的 DID 文档，并加上附加的元数据返回 DID 文档。此功能依赖于具体的 DID Method 方所定义的“读取” 操作。

DID 读取者（Resolver）

DID 读取者是一种软件和/或硬件组件，它以 DID 作为输入，并通过执行 DID 标识符与 DID 文档的匹配（利用读取函数）生成合格的 DID 文档作为输出。

DID 服务端点（Service End Point）

服务端点是 DID 主体提供的网络地址。特定服务的例子可以包括发现服务，社交网络，文件存储服务和可验证的声明存储库，数字资产服务。服务端点也可以由通用数据交换协议（例如可扩展数据交换）提供。

DID 用户代理程序（User Agent）

诸如浏览器或其他 Web 客户端，或者移动客户端之类的程序，它在 DID 所有者（就是 DID 的主体），DID 发行者和 DID 验证者之间进行通信调解。

DID 发行者

DID 发行者可以发行或者创建可验证的凭证，并且进行数字签名，并将该可验证的凭证传送给 DID 主体。DID 主体拥有 DID 发行者所发行的凭证。

DID 验证者

对可验证凭证进行验证。这包括检查：凭证是否符合规范；证明方法是否得到满足，验证过程中需要对于 DID 发行者提供的凭证和数字签名进行验证，因为凭证的和数字签名的 Hash 应该存储在链上，验证者不需要一个中心化的数据库进行验证。DID 验证者可以是 DID 服务的提供方(Service Provider)或者中继方（Relaying Party), 对于 DID 主体提供的凭证进行验证以后，可以提供客户化的服务。

根据 ITEF 的 RFC 3552 规定，DID 做为规范必须在安全方面考虑以下内容：

- 哪些攻击不在范围内
- 协议的弱点和强项。
- 至少必须考虑以下形式的攻击：窃听，重放，消息插入，删除，修改和中间人攻击。还必须确定潜在的拒绝服务攻击。
- 如果协议包含密码保护机制，则应明确指出数据的哪些部分受到保护以及保护的内容（即，完整性，机密性和/或端点身份验证等）。还应该给出一些说明描述密码保护可以受到哪些攻击。应该保密的数据（密钥，随机种子等）应该清楚地标记。
- 如果该技术涉及身份验证，尤其是用户与服务器之间的身份验证，则必须明确指定身份验证方法的安全性。
- DID 的底层的安全，比如区块链， 或者点对点网络的安全不属于 DID 安全和隐私的范畴。CSA GCR 另外工作小组在以后会做另外的研究。

我们按照上面的要求和其他因素的考虑，总结以下方面的安全考虑。按照可能的安全威胁和应对措施进行分析：

3.1 DID 安全注意事项：窃听

威胁：窃听

DID 由于没有规定传输中的机密性要求，窃听方可能会在相对应的响应（Response）DID 文档中获取敏感信息。DID 请求的详细信息将削弱请求方和响应方的安全性。

对策：

加密 DID 文档的某些属性，例如 DID 主题，服务端点以及授权和委托信息。利用 TLS 或 IPSec 作为 DID 事务的传输层将提供一定程度的保护。目前，最佳实践是在需要传输的 DID 文档里面不写任何诸如 PII 的敏感信息。

3.2 DID 安全注意事项：重放攻击

威胁：重放攻击

重放攻击有两种：

（1）重放会话（Session）攻击：黑客可能会使用经过验证的身份信息和会话内容进行重放会话攻击从 DID 交易方提取资金或进行其他未经授权的活动。

（2）硬分叉引发的重放攻击：在两条链（尤其是硬分叉链）上进行相同的交易。例如，由于最初的 BCH 和 BSV 硬分叉没有对重放攻击的限制，因此用户可以在 BCH 和 BSV 链上发起相同的交易，从而导致在两个链上重放交易或者双化。

对策：

（1）会话管理：DID 协议未指定任何会话管理要求。一种可能的对策是使所有 DID 连接变为无会话（Sessionless）连接。如果在某些应用场景会话是必需的，则一种可能的设计是允许“一次性使用密码(OTP)“或每次交易的往返都使用随机数(nonce)作为这个交易的唯一标识符来防止重放攻击。

（2）硬分叉引发的重放攻击：在每个 DID 方法规范中，可以在 DID 文档的交易上下文中引入命名空间。发送到硬分叉链的交易可以包括命名空间标识符，以指定交易要使用的硬分叉链，以便只有一个链的矿工可以接受该交易。

3.3 DID 安全注意事项：消息操纵

威胁:

日食攻击（Eclipse Attack）：在极少数情况下，如果区块链的基础层具有相关漏洞，那么许多矿工可能会阻止，插入，删除或修改 DID 文档或文档碎片（日食攻击）。

合谋攻击（Collusion Attack）：DID 控制者们在不经过 DID 主体同意的情况下合谋更改 DID 文档。

超前运行攻击（Front RUNNING Attack）：DID 对等方可以支付更高的 Gas，或者矿工（共识节点）可以超前运行某些 DID 交易（如果 DID 服务端点可以执行某些数字资产交换交易），超前运行的效果类似于消息删除。

对策：

Eclipse 攻击：应对这种攻击的对策是在 P2P 网络层检查并修复对等点发现方法，这与基础层 DLT 技术有关。

合谋攻击（Collusion Attack）：当 DID 所有者或主体丢失密钥时，DID 控制者们可被用于社会恢复。但是，如果 DID 控制者们合谋，则 DID 所有者可能会丢失与 DID 文档关联的数字资产或凭证。使用时间戳记（BIP 0113）和时间锁定（BIP 65）可以作为缓解策略。

超前运行攻击：（1）带外通信（out of band）是一种对策。这个对策利用其它非区块链频道，比如利用电子邮件进行预先的交易匹配，匹配完成以后再利用区块链技术进行链上的点对点定向交易 （2）也可以用可信的中继系统防止超前运行攻击。可信的中继系统是一个相当于比特币闪电网路的第二层技术（Layer 2）可以增强性能和隐私，一定程度上防止超前运行攻击。但是不能防止中继系统本身的超前运行攻击。

3.4 DID 安全注意事项：中间人攻击

威胁：中间人攻击（Man in the Middle Attack）

恶意用户代理（Malicious User Agent）：

用户代理是包含 DID 文档和私钥的客户端或者数字钱包。恶意用户代理可能会被用户错误地下载，恶意用户代理可能盗取私钥或者代理可以替换剪贴板中的目标地址或 URI，如果 DID 交易包括取款，则可能存在风险（类似的攻击发生在 Ledger 钱包上：https：// www .ledger.com / academy / hack-5-malicious-wallet）。

恶意 DID 读取者（DID Resolver）：DID 读取者是一种软件和/或硬件组件，它以 DID 标识符作为输入，并通过执行 DID 标识符与 DID 文档的匹配（利用读取函数）生成合格的 DID 文档作为输出。DID 读取者本质上是分布式的，不应部署为中心化的服务。在某些情况下，多个 DID 读取者程序可能会合谋或被攻击者欺骗，从而发生中间人攻击。

对策：

恶意用户代理：请勿从未经验证的来源下载用户代理应用，例如钱包，交易所或其他 DLT 应用，请始终使用双因子验证（2-FA）。使用 2-FA 并不能保证完全的防御能力，但会使黑客更难。

同时，我们需要保持用户代理应用应用的更新，防止安全漏洞因为更新不及时被利用。

恶意 DID 读取者（DID Resolver）：请勿使用未经验证的 DID Resolver，如果可能，请构建和部署自己的 Resolver 群集，并在 DID Resolver 群集上加固安全。

3.5 DID 安全注意事项: DID 的 CRUD 安全

威胁：DID 的 CRUD 安全

DID 文档的创建，阅读，更新，和删除（CRUD）是 DID 文档的生命周期，具体如何实现与底层的 DLT 或者区块链技术有关。每一个具体的定义是由 DID Method （方法）来决定。需要考虑的安全威胁包括如下：

- 创建（Create）：DID 文档创建的的客户端是否安全？有没有产生弱的私钥？ 用户是否因为操作失误引起私钥的丢失和泄露？
- 阅读（Read 或者 Resolve）：未经验证的 DID Resolver 可能发起中间人攻击。
- 更新（Update）：经验证的 DID Resolver 如果合谋串通，则可以恶意更新 DID 的元数据或控制者。

- 删除(Delete)：DID 文档删除意味着 DID 标识符与 DID 文档的关联被删除了。删除是永久的，不可撤消。因此用户的失误操作会产生不可逆转的安全风险。

对策：

- 创建：DID 用户代理应用的安全必须经过第三方安全测试防止安全漏洞，并且对应的应用必须及时更新。同时需要教育终端用户安全使用 DID 用户代理应用。

- 读取：请勿使用未经验证的 DID Resolver，如果可能，请构建和部署自己的 Resolver 群集，并在 DID Resolver 群集上加固安全。DID Resolver 的代码应该使用经过第三方安全测试的开源代码。

- 更新：使用时间戳（BIP 0113）和时间锁定（BIP 65）的相似方法对于更新进行时间戳的检查和时间锁的审查。防止对 DID 文档进行未经授权的更改，在发生更改时监视并积极通知 DID 主体。这类似于密码重置通知发送到电子邮件地址来提醒用户关于密码的修改。对于 DID 文档的修改，没有中介注册商或帐户提供商可以生成此类通知。但是，如果利用可验证数据注册表支持更改通知，则可以向 DID 控制者提供订阅服务。通知可以直接发送到现有 DID 文档中列出的相关服务端点。

- 删除：删除前请仔细检查，或者利用时间戳和控制者多签名的方法防止单方面的失误。

3.6 DID 安全注意事项：密钥和签名到期

威胁：在 DID 的标识符体系结构中，没有中央授权机构来强制执行密钥或签名到期策略。因此 DID 读取者和其他客户端应用程序需要验证密钥在使用时没有过期。使用过期密钥或丢失密钥会引发安全问题。

对策：

时间戳（BIP 0113）和签名到期时间可以是与 DID 私钥部分关联的元数据的一部分，并由 DID 文档的控制者签名。验证因此可以用于时间戳和到期时间。

由于某些应用场景下可能有合理的理由可以扩展已经过期的私钥，因此可能需要一种安全的机制使用过期的私钥。

3.7 DID 安全注意事项：抵赖攻击（Repudiation)

威胁：如果用户丢失了私钥或被欺骗签名不应该发生的交易，则可能会产生抵赖攻击。使用过期的密钥或丢失的密钥也会带来抵赖攻击。

对策：对于高价值（High Value）的交易，设置监视并通知 DID 主体关于交易的细节。 也可以对于 DID 文档实现访问控制机制。 如果底层的 DLT 账本支持时间戳（BIP 0113），可以利用时间戳增加不可抵赖的能力。防抵赖还有一个重要措施就是自动或人工审计。

3.8 DID 安全注意事项：服务端点(Service End Point)

威胁: 在某些情况下，对于 DID 服务端点的访问需要具有特权，未经授权的访问可能是安全的隐患。

对策：需要为服务端点定义身份验证和访问控制。目前的 DID 规范对服务端点的身份验证和访问控制没有规划。具体实现需要按照应用场景来决定。

3.9 DID 安全注意事项：缓存

威胁： DID 文档缓存可能被黑客进行篡改。

对策： 缓存的 DID 文档可以有时间戳和内容的数字摘要（Digest）进行 Hash. 使用缓存的 DID 文档需要对于 Hash 进行比对。另外，对于 DID 版本可以检查。同时，需要注意缓存过期和刷新间隔的处理。

对于缓存的安全保护，也可以采用 Intel SGX 等技术或者针对分布式缓存系统的安全进行加固。

3.10 DID 安全注意事项：密钥吊销和恢复

威胁：虽然不是黑客攻击造成的安全威胁，用户可能不小心泄露 DID 私钥或者丢失私钥。

对策：可以利用 “社会恢复” （Social Recovery）：利用 DID 的 Controller 属性指定可以帮助 DID 主体进行密钥恢复的 DID 控制者。

另外一个方法是使用时间戳（BIP 0113）和时间锁定（BIP 65））来吊销和恢复私钥。也可以利用传统保护私钥的方法来防止私钥丢失，例如利用硬件安全模块（Hardware Security Module）或者可信计算环境（Trusted Execution Environment）和本地安全攻击检测措施。

3.11 DID 安全注意事项：中继方（Relayer）威胁

威胁：中继方或服务提供商可能在没有经过 DID 主体或者控制者同意的情况下在中继方本地的服务器存储用户 DID 文档，从而引发安全和隐私问题。

对策：

利用同态加密： 同态加密允许对加密数据进行数学运算（例如，DID 文档中的 DID 主体和 PII 数据元素在事务处理中被加密而无需解密）。

利用多方计算，允许多个实体安全地将其数据贡献到组合数据集中以进行基于 DID 的交易，同时保持其数据彼此之间的私密性（例如，DID 各方可以使用 MPC 进行供应链融资）。

也可以利用差分隐私（Differential Privacy）： 差分隐私允许通过向聚合查询结果添加随机化"噪声"来实现，以保护个人的条目，而不会显著改变查询结果。差分隐私算法保证攻击者能获取的个人数据几乎和他们从没有这个人记录的数据集中能获取的相差无几。

3.12 DID 安全注意事项：残留风险

威胁：残留风险是企业在尽一切努力确定和消除风险之后仍然存在的威胁。由于残留风险是未知的，因此许多组织选择接受或者转移残留风险-例如，通过购买保险将风险转移给保险公司。例如，如果企业使用开源的 DID 代码，开源 DID 代码库更新可能会带来持续的残留风险。

对策：

（1）针对使用的开源代码，做好开源软件的生命周期管控，尤其是开源软件中的安全漏洞，必须及时修复。

（2）业务场景发生变化时需要重新审视残留风险

（3）对于高价值的 DID 文档，有必要向有执照的保险公司投保。

3.13 DID 隐私注意事项：将个人身份信息（PII）保密

威胁：DID 文档，可能帮还个人隐私数据，如果存储到公开的分布式账本就有隐私泄漏的风险。

对策：

可能帮还个人隐私数据，如果存储到公开的分布式账本就有隐私泄漏的风险。

如果 DID 文档需要存储到公开的分布式账本，那么 DID 文档不能包含任何个人数据(PII)。

所有个人数据（PII）应保存在服务端点（Service End Point）之内，对于这些数据的访问控制，应该由 DID 主体的决定。

服务端点中的 URL 不应泄漏个人数据或相关信息，例如包含用户名的 URL 如果出现在 DID 文档中可能无意间泄露 DID 主体不愿意公开的敏感信息。

在点对点的 DID 交易中，DID 主体可以对于不同的交易方使用不同的 DID 公开标识符（Pairwise DID，按照不同的交易对，分配和使用不同的 DID 公开标识符），这样可以最大程度地增加隐私。也可以采用匿名化技术消除 PII 信息。

因为 PII 数据不会被存储在不可变的分布式账本上，GDPR 和其他有关数据隐私条例可以得到满足。

3.14 DID 隐私注意事项：DID 标识符的关联风险

威胁：DID，表识符如果多次使用，可能有被关联引起隐私信息泄露的风险。

对策：

像任何类型的全球唯一标识符一样，DID 标识符可以用于关联。DID 控制者或者主体可以通过使用成对（Pairwise）的唯一 DID 标识符来减轻这种隐私风险。

仅当 DID 主体明确授权其他方之间的关联时，才容许与多方共享 Pairwise DID。

在 DID 主体明确同意下，DID 标识符可以作为主体的公开身份发布在区块链账本上，在这种情况下，DID 主体是容许标识符关联的。

3.15 DID 隐私注意事项：DID 文档的关联风险

威胁：DID 文档被关联可能引起隐私泄露

对策：

如果可以将对应的 DID 文档中的数据进行关联，则容易破坏上面提到的 Pairwise DID 的反关联风险保护。例如，在多个 DID 文档中使用相同的公共密钥描述或 DID 服务端点可以提供与使用相同的 DID 标识符一样多的关联信息。因此，Pairwise DID 的 DID 文档还需要使用成对（Pairwise）的唯一公共密钥(Public Key)，并且需要使用成对的唯一服务端点（Service End Point）来增强隐私。但是，唯一的服务端点可以将两个 DID 之间的所有流量被双方存储权利，进行时序相关性和

类似的分析来增加关联性。因此，端点隐私的更好策略可能是在由许多不同 DID 主体控制的数千个或数百万个 DID 之间共享一个服务端点（Mixer 混合器）。

3.16 DID 隐私注意事项：群体隐私

威胁：

当 DID 主体与群体中的其他主体无法区分时，可以一定程度保留隐私。但是，当一个 DID 主体与另一方交往的行为本身可以成为识别的标志时，隐私就会大大减少。

对策：

可以利用下面的技术增强这方面的隐私：

1. 团体签名（Group Signature）允许一组实体在掩盖其身份的同时进行交易，仅揭示了“该团体中的某人”被交易了。
2. 秘密共享（Secret Sharing）或多重签名（Multiple Signature）方案可以确保仅在足够数量的实体（例如，五个中的三个）同意时才披露敏感数据。
3. 零知识证明（Zero Knowledge Proofing ）可以在不透露数据的情况下证明对数据的要求（例如验证 DID 主体是一个成人，但是不需要透露真正的年龄）。
4. 同态加密（Homomorphic encryption）允许对模糊数据进行数学运算（例如，DID 文档中的 DID 主题和 PII 数据元素在事务处理中被加密而无需解密）。
5. 多方计算（Multi-Party Computation）允许多个实体安全地将其数据贡献到组合数据集中以进行欺诈检测，同时保持其数据彼此之间的私密性（例如，DID 各方可以使用 MPC 进行供应链融资）。
6. 差分隐私：允许通过向聚合查询结果添加随机化"噪声"来实现，以保护个人的条目，而不会显著改变查询结果。差分隐私算法保证攻击者能获

取的个人数据几乎和他们从没有这个人记录的数据集中能获取的相差无几。

第四章： SID 在国内的应用案例

4.1 SID 在数字政务中的使用

4.1.1 中国数字政务的现状

《2020 联合国电子政务调查报告》数据显示，我国电子政务发展指数从 2018 年的 0.6811 提高到 2020 年的 0.7948，取得历史新高。其中，作为衡量国家电子政务发展水平核心指标的在线服务指数上升为 0.9059，指数排名大幅提升全球至第 9 位，国家排名位居第 12 位，在线服务达到“非常高”的水平。分析发现，本次联合国电子政务调查报告中我国在线服务全球排名的大幅提升，与我国不断深化“放管服”改革和大力推动全国一体化政务服务平台建设的决心与行动密不可分。

电子政务是指政府行政部门利用信息技术及现代通信技术变革传统工作模式，面向政府在社会服务中的各项业务所建立的集成化、综合化的政府信息系统。2018 年 7 月 31 日出台的《国务院关于加快推进全国一体化在线政务服务平台建设的指导意见》提出，要在 2022 年底前，全面建成全国一体化在线政务服务平台，实现“一网办”。

数字政务除了实现政务数字在线化，还包括社会治理机制的网络化。中国数字政务可分为以下四个阶段 ，描述如下：

（1）面向办公自动化-网络互联互通阶段

80 年代初到 2001 年以前 20 年，是使用计算机协同处理政务业务，部门内部网络互联互通的初级阶段。

（2）面向业务信息化-应用百舸争流阶段

2001~2011 年，国家围绕“一网四库十二金”进行了国家级应用和系统建设，形成了政府信息化的主要框架。各级地方政府也规划建设了相当数量的应用系统，其主要特征是通过信息化手段将政府管理事务进行数字化。但数字政务始终处于一个支撑，协助的地位，没有将政务管理事务的流程进行再造和创新。同时，这个阶段带有明显野蛮生长的特征，在顶层设计缺失的情况下，形成了部分信息孤岛和应用藩篱。

（3）面向资源集中化-基础架构进化阶段

2011~至今，这个阶段的发展模式以整合、集中为主。数字政务在政策导向下得到了长足发展，但是，资源集约化、数据集中化在各部委各地方政府，落地时阻力较大。从 2015 年开始，云计算、大数据、区块链等技术进入产业化阶段，为数字政务的集中化带来了契机。“进化”阶段结束后，数字政务基础设施将兼具规模、弹性、灵活等特性，形成健壮的 IT 基础架构和强大的中台特性，为未来的应用创新打下良好基础。

（4）面向业务创新-数字政务泛化阶段

未来随着布式数据库、微服务框架、大数据等平台的建立，应用的建设时间、成本、运维复杂度等都将极大降低，应用创新会进入一个高速发展期 。最终，随着数字政务的不断成熟，这些创新性应用将切入城市的管理与服务、引导和促进产业与经济的发展，真正形成基于数字政务的智慧城市。见下图，面向业务创新-数字政务泛化的发展阶段

4.1.2 目前中国数字政务的痛点

目前数字政务处于面向资源集中化-基础架构进化阶段末期阶段，实现为数字政务的集中化，要建成一站式政务平台，我国还面临以下一些痛点：

（1）信息资源不共享、不开放，信息资源使用效率不高

由于对信息资源的归属、采集、开发等的相关管理规则还不明确，造成了不少政府部门将政府信息资源的产权部门化，人为设置信息互联互通的壁垒，从而导致了资源归属上的“部门私有”。政府部门出于维护部门权威，以及部门主管出于部门利益考虑和避免担负数据泄露的责任，不愿进行信息交换与应用，信息资源共享。

各级政府部门掌握大量的政府信息数据库,这些库分别属于不同部门。由于缺乏对公众使用信息资源的服务意识和对信息资源的协调管理，普遍存在信息资源量不足、信息内容更新不及时、网络平台交互性差，突发事件应变能力较弱等问题，不少数据库缺乏开发维护，处于闲置状态。政府信息资源交流不畅，信息资源利用率低造成社会资源极大浪费。

（2）到政府部门办事存在办证办事难、奇葩证明多、证件重复提交等问题

群众到政府部门办事过程中，常常辗转于几个部门之间，其实递交的往往是同一信息，由于部门间没有进行数据共享，互联互通，只能由办事者一家家跑腿，

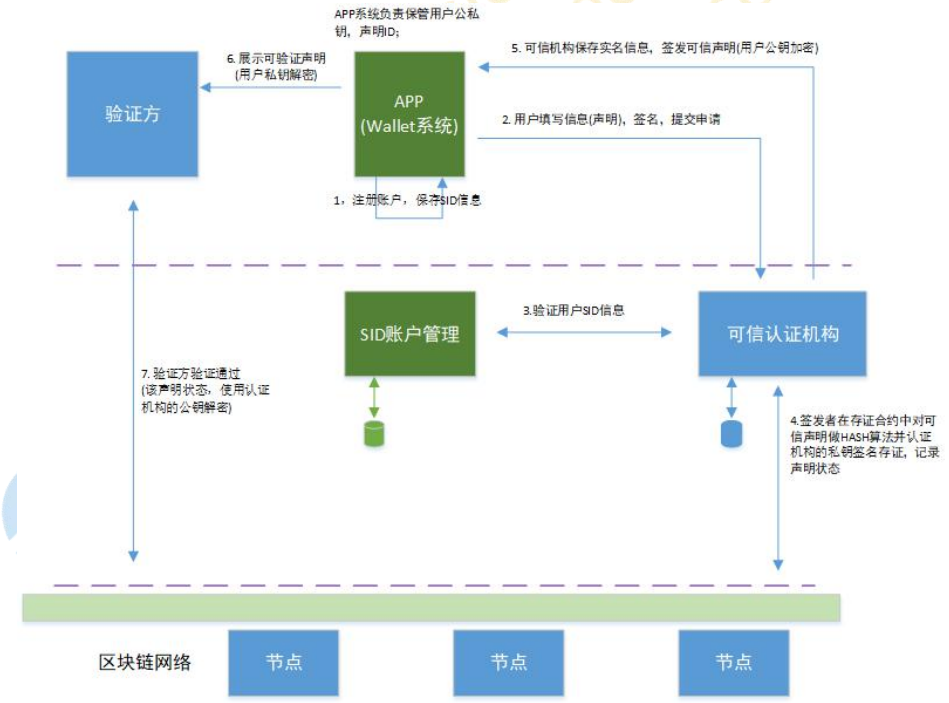
导致资料重复提交。更有一些部门里科室之间信息也不联通，到一个部门办事，还得一个个科室跑，这样造成办事者费时费力，体验差，老百姓颇多怨言，同时增加了行政成本。

有时候，还出现需要证明“我是我”“我妈是我妈”，这样的奇葩证明让人哭笑不得，但遇上时却不知如何处理。

（3）网络安全存在隐患

目前的电子政务系统一般是中心化的数据库，很难防止内部或外部对存储的数据进行篡改。

4.1.3 SID 用户自治数字身份解决方案



上面的图是 SID 解决方案的架构图。SID 用户自治数字身份一般由 2 部分组成，一部分是 SID 信息，用于标识用户 ID，和控制公钥等信息，另一部分是可验证声明（Verified Claim），用于证明该用户的有效信息，例如是身份证信息，驾驶证信息，或者医学出生证明等。

SID 系统的角色介绍：

1. APP 系统

一般是数字钱包类 APP，可以是去中心化钱包，负责保管用户公私钥，可验证声明（加密）等信息，由用户自行掌管。

2. 可信认证机构

可信认证机构一般是公安局，交通管理部门，或者是卫生健康委员会等权威机构的在线平台，负责制发电子证书和信息验证的正确性。

3. 验证方

验证方一般是用户证书的使用方，需要验证用户身份或者证书的状态。

具体的注册，发证，验证的业务流程描述如下：

1. 用户安装/注册好 APP，产生 SID 数字身份信息。
2. 用户填写相关信息（声明），使用自己的私钥做签名，提交证书申请。例如填写身份信息，发起电子身份证信息认证给公安局平台；认证机构采用该用户的公钥进行解密，把信息内容发到 SID 账号管理系统进行验证。
3. 可信认证机构在存证合约中对可信声明做 HASH 算法并认证机构的私钥签名存证，记录声明状态，保存到区块链上。
4. 可信认证机构验证通过的声明，使用该用户的公钥进行加密，然后使用机构自身的私钥进行签名后发送给用户保存；
5. 当用户需要向证件验证方出示证件时，例如入住酒店需要出示电子身份证，使用自己的私钥解密第 5 步保存的可验证声明，展示给酒店。
6. 证件验证方收到信息后，采用可信认证机构的公钥解密获得该可信声明证书的 HASH 值和证书状态。对比获得的证书 HASH 值和保存在

链上的证书 HASH 值，如果一致则证明该用户呈现的证书和状态是正确的。

4.1.4 SID 用户自治数字身份应用列表

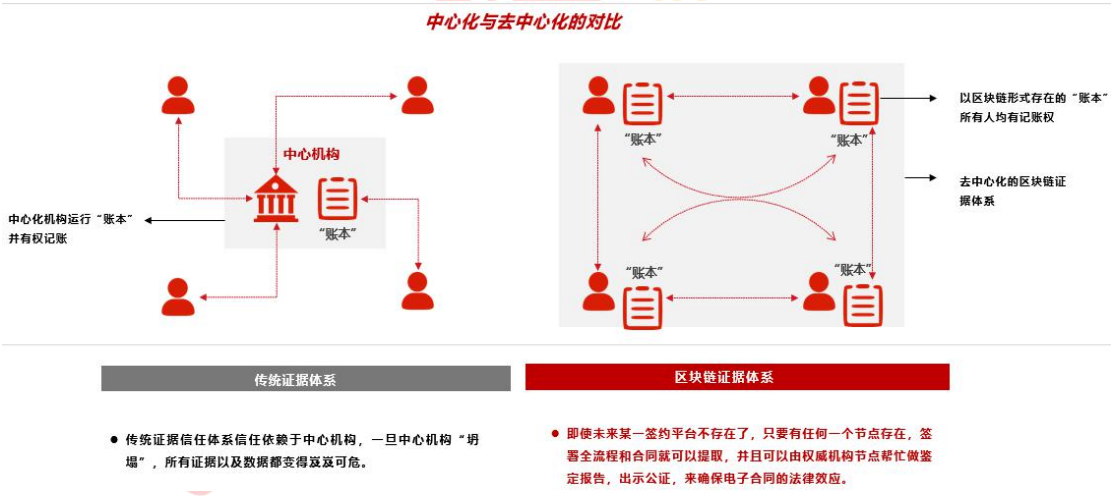
据互链脉搏做过分析，借“1024”政策东风，以及区块链技术在后疫情时代发挥的作用，区块链+政务渴望在 2020 年爆发。其统计了从 2017 年至今已建设完成、正在建设中共 96 项区块链政务应用，数字身份应用，也恰是政府区块链政务探索的主要方向，其项目数量占到总量的五分之一（20 项）。这些区块链数字身份管理应用大部分都使用分布式数字身份方案。见下面的表格。

省市	政务区块链应用	发起方	上线、披露时间	项目一级分类	项目二级分类
广东省佛山市	广东佛山禅城IMI数字身份平台	禅城区政府	2017年6月	数字身份	公民身份
台湾省台北市	“TangleID”公民身份证	台北市政府信息科技处	2018年1月	数字身份	公民身份
重庆市	“社区民警智能名片”区块链项目	江北区公安分局	2018年2月	数字身份	公民身份（警务人员）
河北省雄安新区	基于区块链技术的智慧垃圾收集样机器	雄安新区管委会	2018年5月	数字身份	公民身份（激励）
重庆市	区块链信用公示管理体系	巴南区政府	2018年8月	数字身份	公民身份（激励）
河南省开封市	“链政通”数字身份平台	兰考县政府	2018年	数字身份	公民身份
澳门	身份认证	澳门政府设立的澳门科学技术发展基金	2019年2月	数字身份	公民身份
广东省深圳市	网贷机构良性退出投票表决系统	深圳市互联网金融协会	2019年6月	数字身份	公民身份（投票）
广西省南宁市	“区块链+人社”应用平台	南宁市人社局	2020年1月	数字身份	公民身份
福建省福州市	疫情防控数据管控区块链系统	福州市马尾区政府	2020年2月	数字身份	公民身份
山东省济南市	身份健康码	济南市公安局	2020年3月	数字身份	公民身份
河北省雄安新区	个人积分系统	雄安新区管委会	尚未上线	数字身份	公民身份（激励）
江苏省南京市	基于区块链技术的电子证照共享平台	南京市政府	2017年5月	数字身份	电子证照
江苏省	中兴通讯GoldenChain政务服务平台	江苏省政府	2017年9月	数字身份	电子证照
山东省济南市	济南“区块链数字证照系统”	高新区政府	2018年9月	数字身份	电子证照
广东省广州市	市级电子证照系统	广州市政府	2019年7月	数字身份	电子证照
重庆市	企业开办网上服务平台	重庆市市场监管局	2019年6月	数字身份	电子证照
广东省深圳市	区块链电子证照应用平台	深圳市政府	2019年12月	数字身份	电子证照
广东省惠州市	惠城区智慧政务一体化商事登记秒批系统	惠城区政府	2020年2月	数字身份	电子证照
江西省	“区块链+电子证照”应用	江西省政府	尚未上线	数字身份	电子证照

4.2 SID 在电子签名中的使用

4.2.1 中国电子签名的现状

随着中国数字经济发展的不断深入，数字信息技术应用快速渗透。2019 年以来，国务院财政部、海关总署、交通运输部、人社部等多个政府部门发文表示要加强推进电子签名技术应用。艾媒咨询分析师认为，电子签名在政府部门的应用，是政务数字化变革的必然趋势，对电子签名在社会经济领域的广泛应用起到较强的指引和助推作用。在政务部门强力推进的示范效应下，电子签名市场将会迎来强需求爆发期。随着技术的成熟发展，电子签名平台已经广泛运用到合同签约、管理和法律服务全部环节，为企业多方位赋能。艾媒咨询分析师认为，随着电子签名平台服务内容的延伸，企业对电子签名平台的依赖程度将不断加深。电子签名为企业提供多方位赋能，但传统的电子合同和区块链合同有本质的不同，如下图电子签名中心化与去中心化对比所示：



政务信息化改革的深入开展将为电子签名行业带来持续性政策利好，加速推动电子签名自上而下渗透。电子签名发展至今可分为以下三个阶段 ，对比如下：

	纸质合同	电子合同	区块链合同
定义	合同 1.0：合同书面化 纸质契约时代	合同 2.0：合同电子化 远程签约时代	合同 3.0：可自动执行的程序 链签约时代
签约方式	**线下签约** 流程长、易出萝卜章	**线上签约** 可远程签约	**链上签约** 秒级签署，杜绝萝卜章
管理方式	**线下管理** 成本高、不易查询	**线上管理** 中心化系统管理	**链上管理** 无法篡改、不会丢失
履约方式	**线下履约** 履约成本高、维权成本高	**线下履约** 履约与签约缺少强关联	**链上履约** 区块链、电子签名、司法三重效力保障

4.2.2 当前电子签名的痛点

传统电子签名与区块链签约的区别，如下图所示：

区块链合同最大的挑战，比如：

- 身份认证
- 安全合规
- 商业信用
- 中心化的 CA 中心

4.2.3 SID 电子签名的解决方案

e 签宝和蚂蚁金服推出了业界第一个全流程上链的电子合同产品，如下图所示：

普通的电子合同打官司，需要：准备电子合同，各类出证证明、准备其他相关材料、提交给法院、法院进行审核，审查、判决等多个环节，全流程上链的电子合同，法院可以直接在线上核验合同及出证报告真伪，达到一键诉讼，在线判决的效果，电子合同的有效性包括 4 个要素，真实本人，真实意愿，签名未改，原文未改。区块链电子合同解决了保证签署材料上传区块链之后到出证这段时间内，无人篡改，公信力大增，但是没有解决身份认证的问题，传统的身份认证还是基于生物识别、三要素、四要素等方式进行身份识别，这些都涉及客户隐私，因此又会引入隐私保护的问题，所以亟需一套身份认证的解决方案，e 签宝主要

服务于 Tob 企业客户，因此 e 签宝从印章出发，联合蚂蚁集团、政府，互联网法院等，给每个企业颁发区块链印章，构建起电子印章从申领到使用，从使用到审判的闭环，这样可以大幅加速政企数字化转型，实现安全可靠的全流程在线服务，基于 SID 的区块链电子签名，如下图所示

“一链”指统一电子印章区块链，工商局、公安局等政府单位成为“统一电子印章区块链”监督节点，和“司法链”跨链打通。印章申请流程、企业基本信息、印章信息、电子签章全流程证据上链，构建起电子印章从申领到使用，从使用到审判的闭环。

“一系统”指电子印章核心系统，具备统一电子印章库、统一服务商接入、统一印章管理的能力，用户可以在此查看自己的印章以及所签署的文件。

“一身份”从公安印模库同步，面向印章制作单位和第三方电子签名服务平台提供电子执照、公民身份认证服务和印模查询验证服务，实现印章和身份的关联并上链，实现安全可靠的身份认证。

最后看下使用案例，SID 电子签名的案例如下图所示：

参考资料

1： http://www.jcr.cacrnet.org.cn/CN/10.13868/j.cnki.jcr.000352

杨婷，张光华，刘玲，张玉清．物联网认证协议综述[J]．密码学报, 2020, 7(1): 87-101.

2：YANG T, ZHANG G H, LIU L, ZHANG Y Q. A Survey on Authentication Protocols for Internet of Things. Journal of Cryptologic Research, 2020, 7(1): 87-101.

3：https://docs.aws.amazon.com/iot/latest/developerguide/authentication.html

4：https://azure.microsoft.com/zh-cn/blog/iot-device-authentication-options/

5：https://yq.aliyun.com/articles/697891

6：https://www.jianshu.com/p/4491e87b8e13?from=timeline&isappinstalled=0

7：https://www.ibm.com/developerworks/xml/library/x-xacml/

8：http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

9：https://www.evernym.com/blog/the-three-pillars-of-self-sovereign-identity/

10：https://github.com/w3c-ccg/did-test-suite/

11;github.com/hyperledger/aries-rfcs/blob/master/concepts/0207-credential-fraud-threat-model/README.md

12：https://medium.com/decentralized-identity/the-self-sovereign-identity-stack-8a2cc95f2d45

13：https://www.iimedia.cn/c1020/72485.html

14：
https://newmetaverse.org/white-paper/Metaverse-digital-identity-white-paper-v1.0-EN.pdf

15：
https://www.civic.com/blog/evolving-trust-with-applied-game-theory-recent-white-paper-update-describes-trust-creation-through-smart-contracts/

16：https://www.evernym.com/

17：https://www.uport.me/

18：https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE2DjfY