Fidelius: 面向数据合作的隐私保护区块链解决方案

Fidelius: YeeZ Privacy Protection for Data Collaboration —

A Blockchain based Solution

熠智科技

https://yeez.tech

2022 年 3 月

目录

1 背景介绍

1.1 数据合作

进入大数据时代，人类获取、管理和利用数据的能力空前提升，社会各界对数据的价值愈发重视[1]。在数字经济时代，数据已经成为关键生产要素，就像在农业经济时代和工业经济时代中，土地、劳动力和资本是关键生产要素[2]。

相比传统生产要素，数据有着非常强的网络效应，当不同维度、不同来源的多种数据相结合的时候，其中蕴含的社会价值、经济价值往往能达到 1+1>2 的效果。例如，医疗数据与位置信息的结合能够更快的控制传染病的发生规模；设备传感器的数据与用电数据的结合能够极大的提高用电效率；企业订单、物流数据与金融数据的结合能够为中小企业带来更精准的金融政策等。

可以说，在大数据时代，数据社会和商业价值的变现，需要多方参与，我们将这种**为获得数据的综合见解而对多个独立数据进行的跨域数据处理**称为数据合作 [3]。

数据合作通常会涉及到多方参与，但一般可分为两类角色：**数据提供方与数据使用方**，前者提供数据，后者使用数据提供方的数据。

一般情况下在某个数据合作场景中可能存在多个数据提供方或者数据使用方，甚至某个主体即是数据提供方也是数据使用方。

而更大范围的数据合作则通常需要引入**中间人**，后者帮助完成数据的展示、转发甚至必要的撮合、交易。中间人一般以中心化平台的形式呈现，近年来各地纷纷成立的大数据交易中心和平台[4]，正是承担着上述数据合作中间人的角色。图 1描述了这两种数据合作的模型。

虽然企业、部门之间需要进行数据合作，但是在数据合作的实践上却差异很大，这是因为 1) 参与数据合作的主体之间的关系的影响；2) 数据合作内容、目的的区别 3) 数据合作技术的限制等。

同时数据合作在表现形式上有多种形式，包括但不限于：

- 数据共享：通过某种方式实现多方原始数据的互通，例如彼此开放数据访问的

[1] 《开放的数林——政府数据开放的中国故事》，郑磊，上海人民出版社

[2] 《中共中央国务院关于构建更加完善的要素市场化配置体制机制的意见》，2019

[3] What is data collaboration，https://medium.com/infosum/what-is-data-collaboration-6519e604a365

[4] 全国大数据交易所及数据交易平台汇总，http://www.tanmer.com/blog/541

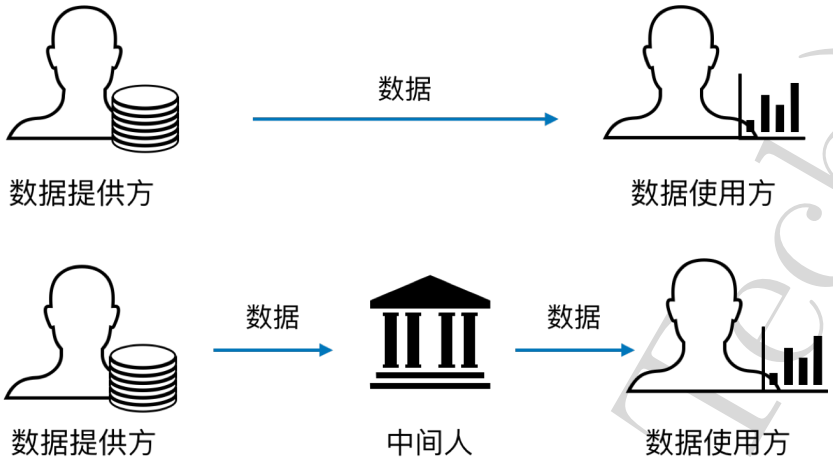

图 1: 数据合作模型

API、搭建专线、物理介质传输等；

- 数据交易：多方之间的数据交易行为，类似于数据共享，常见于存在中间人的情况，由中间人进行交易的撮合、数据的交付及支付；
- 数据分析：数据分析通常需要基于多方数据进行联合建模从而挖掘更多数据价值，例如近年逐渐受到关注的联邦学习。

数据合作的应用场景非常广泛，这里列举一些应用示例：

- 金融风控：银行在开展互联网贷款业务时，坏账率对于业务成败非常关键，因而需要对用户进行更精准的风控刻画，也需要针对整体的风控算法和参数，与外部多方协同合作；
- 营销分析：在广告营销领域，广告商和流量方需要接入第三方数据，从而实现精准定位客户来提升投放以及转化效果；
- 供应链金融：商业银行等金融机构通过结合供应链中的信息流、物流、资金流数据，来缓解金融机构和中小型企业之间信息的不对称问题，从而更有效的向核心企业以及供应商提供融资和贷款等业务；
- 园区综合能源服务：电网公司通过对园区内能源生产和消耗的数据的分析，可以实现包括发电和冷热负荷的协同利用，通过优化、经济分析，进而保证能源供应的安全性和经济性的平衡管理；
- 医疗数据共享：医疗信息对保险公司、医保机构和流行病监测部门具有至关重要的作用，医疗信息的共享可极大地提高保险的精准化水平、医保部门政策的精细化和流行病防治的及时性；同时诊疗流程中有极多的信息点可以帮助后期

的临床质量提升或新治疗方案的产生，医疗机构可以通过共享医疗数据提供临床辅助决策支持、应用管理决策支持等；

- 共享征信：当前我国获取主体信用信息的模式有数据中心模式、第三方征信模式、共享查询模式三类。传统征信机构通常采用数据中心模式，出于对个人信息的保护，央行往往审慎从严下发个人征信牌照，类似地第三方征信机构利用自身系统或技术优势，构建中心化平台向授信机构提供服务；而近年来的共享征信则是数据合作的典型应用：业务机构无需事先将数据上报给共享中心，数据由机构自行管理，当机构需要获取数据时，通过中心发送到其他机构，有数据的机构回应信息，返回查询机构。

当前业界已经逐渐意识到数据合作是数据作为生产资料的必然诉求[5]，然而，如何保证数据提供方的数据产权及如何保护数据的隐私是企业、部门之间开展数据合作的关键前提。

1.2 数据隐私保护

隐私保护是一个十分庞杂的概念，通常是指使个人或集体等实体不愿意被外人知道的信息得到应有的保护。对于个人来说，一类重要的隐私是个人的身份信息，即利用该信息可以直接或者间接地通过连接查询追溯到某个人; 对于集体来说，隐私一般是指代表一个团体各种行为的敏感信息[6]。不难看出，对于个人和集体（企业）而言，隐私保护的概念和范畴不尽相同。

对于个人而言，隐私数据是自身和周边相关环境的个人数据，同时个人隐私数据的使用受到严格监管，这不仅仅依靠信息安全技术来解决，更多的是依赖于相关法律法规的约束。

> 欧盟《通用数据保护条例》（General Data Protection Regulation，简称 GDPR）定义了任何收集、传输、保留或处理涉及到欧盟所有成员国内的个人信息的机构组织所需要受到的约束；而刚刚于 2020 年新颁布的《中华人民共和国民法典》，以“隐私权和个人信息保护”专章方式，对隐私权和个人信息定义、保护原则、法律责任、主体权利、信息处理等问题作出规定。

对于企业而言，情况则更为复杂。尽管企业数据包含了其收集的个人用户的数据（例如移动 APP 用户信息、电商平台用户数据、保险公司用户数据），但企业数据也

[5] 《蚂蚁金服在大数据合作上的创新实践》，ArchSummit 2016
[6] https://wiki.mbalib.com/wiki/隐私保护

包括了自身和合作伙伴的关键商业数据（例如经营收入、经营利润、市场占用率等）以及业务数据（例如产品核心数据、生产线数据等）。上述数据既属于企业隐私，也是企业的重要资产，因此，这类数据的泄露会进一步造成企业隐私的泄露；同时，由于数据具有容易复制、不易追踪的特点，其边际使用成本极低，而可能出现二次贩卖会减少企业在数据合作中的收益，也会削弱企业核心竞争力。

隐私泄露问题越来越引起社会重视，对于企业而言，内部员工利用职责便利泄露数据等行为一直难以避免[7]。然而，即便看似"安全"的第三方数据合作平台，在利益驱使下也可能出现数据倒卖的问题[8]。

因此，数据隐私保护是企业之间、部门之间开展数据合作的关键前提，除了监管和相关法律的制定等制度层面的措施之外，也应该加强技术手段从根本上杜绝隐私泄露的可能。

值得注意的是，隐私保护也并非单一的技术方案，从数据的发布到数据存储、数据使用过程中，涉及到不同类型的隐私保护技术。

目前主要的隐私保护技术通常包括：

- 数据发布隐私保护：在数据发布时，需要保证用户数据可用的情况下，高效、可靠地去掉可能泄露用户隐私的内容。k-匿名 [1]、l-diversity 匿名 [2] 等针对数据的匿名发布技术可以实现发布数据时的匿名保护。

- 数据存储隐私保护：在数据存储在云平台或者第三方存储服务商时，存储并不能保证是完全可信的。用户的数据面临着被不可信的第三方偷窃数据或者篡改数据的风险。对称加密技术 DES、AES [3] 以及不对称加密技术 RSA [4]、Elgamal [5] 等是解决该问题的传统思路，此外混合加密技术、同态加密技术 [6] 等也是针对数据存储时防止隐私泄露而采取的一些方法。

- 数据使用隐私保护：在数据的使用（例如数据查询、数据分析等）过程中，需要在尽可能保证数据可用性的前提下，采用合适的数据隐藏技术，以防范利用数据发掘方法引发的隐私泄露。现在的主要技术包括：基于数据失真和加密的方法，比如差分隐私 [7] 以及多方安全计算（MPC） [8]、可信执行环境（TEE） [9] 等技术。

[7]《实名举报! 脱口秀演员池子起诉中信银行: 未经授权泄露个人隐私》，新浪财经

[8]2018 年大数据交易平台"数据堂"被查出倒卖 4000GB 的隐私数据给 Google、三星、Microsoft 等境外企业牟取暴利

1.3 数据可用不可见

在企业的数据合作场景中，由于数据分布在多个主体中，并且数据的使用权和拥有权通常并非同一主体，因此数据的隐私保护也更为复杂。

一方面传统的隐私保护技术存在缺陷无法完全避免隐私泄露[9]；另一方面，数据提供方将数据（即便经过匿名处理）交付给数据使用方后，则丧失了数据的拥有权，例如数据的二次贩卖等行为将难以阻止。

实际上对于很多数据使用方而言，其本身诉求并非原始数据，而是基于数据分析做出数据驱动决策（data-driven decision-making）[10]。例如在供应链金融场景中，金融机构并不关心供应商实际的物流、销售数据而是需要基于上述数据判断后者是否符合融资或者贷款要求。这为数据合作提供了新的解决思路。

一个直观的思想是，将对于数据的使用或者计算进行迁移，数据提供方将数据存储在“安全域”内，提供数据处理服务而并非原始数据，从根本上避免数据出域造成隐私泄露，上述思想通常被称为“数据可用不可见”或“数据不出域”。

目前，业界实现数据“可用不可见”的技术路线主要有三条：

1. 联邦学习：一种分布式的机器学习框架，其核心思想是在多个数据源共同参与模型训练时，不需要进行原始数据流转的前提下，仅通过交互模型中间参数进行模型联合训练，原始数据可以不出本地。训练数据不发生转移，从而降低用户数据隐私泄露的风险。

2. 基于密码学技术的密态计算：以安全多方计算、可搜索加密、同态加密、零知识证明等密码学技术为代表。其中一种解决方案是数据提供方本地使用原始数据计算分析结果的同时生成一个基于零知识证明技术的证明，该证明只有在分析结果正确的前提下才能够被生成，同时不会泄露原始数据的任何信息，这样能让数据不出域的同时保证正确性；另一种方案是通过设计特殊的加密算法和协议，从而支持在加密数据之上（不用解密）直接进行计算，在不接触数据明文内容的情况下得到所需的计算结果。

3. 可信计算：通过在基于硬件的可行执行环境（TEE，Trusted Execution Environment）中执行计算来保护数据应用的隐私安全技术，可信执行环境市值在数据机密性、数据完整性和代码完整性三方面提供一定保护水平的环境[10]，目前引入可信执行环境较为成熟的技术有 ARM 的 TrustZone 和 Intel 的 SGX（Software Guard Extensions）。

[9]例如经过匿名等处理后的数据，通过大数据关联分析、聚类、分类等数据挖掘方法，依然可以分析出部分隐私数据

[10]Confidential Computing Deep Dive v1.0

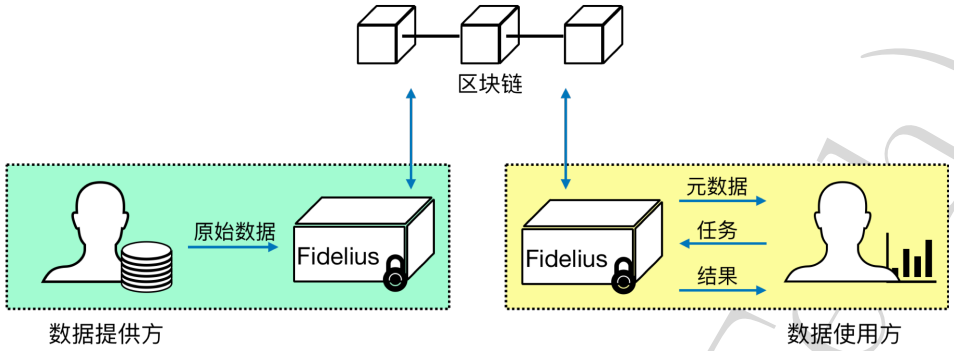

图 2: 隐私保护解决方案

“数据可用不可见”的思想理论上可以有效实现数据合作中的隐私保护，但是仍然存在一些关键问题阻碍了这一技术的实践：

1. 原始数据的一致性：即如何保证计算过程的输入数据是数据提供方声称提供的数据，且没有被篡改；
2. 计算逻辑可控：即如何保证结果不包含数据提供方意愿之外的信息，从而造成数据不可见情况下的隐私泄露；
3. 计算结果的正确性：即如何保证计算结果是由数据使用方提供的计算程序（且未经篡改）生成的，而不是数据提供方随意生成的；
4. 计算结果的隐私性：即如何保证计算结果仅数据使用方可见，数据提供方不能查看计算结果。

这些问题在数据使用方获得原始数据的情况下都是天然保证的，而在“数据可用不可见”的情况下，这些问题的重要性便凸显出来，甚至在一定程度上阻碍了数据合作的可能性。

2 解决方案

为了赋能企业间的数据合作，助力企业利用数据提升自身核心竞争力，熠智科技推出了面向数据合作的一站式隐私保护解决方案。

图 2描述了基于 Fidelius 实现数据合作的抽象流程。

该解决方案最重要的核心组件是 Fidelius 中间件，该中间件分别运行在数据提供方和数据使用方中，双方通过与 Fidelius 中间件交互实现数据合作操作，数据提供方和数据使用方之间没有直接的数据交互，并且原始数据不会离开数据提供方的安全

域，这从根本上避免了隐私数据的泄露问题。关于 Fidelius 中间件将在下一章节有更详细的介绍。同时，整个方案也结合了区块链技术。由于区块链本身具有去中心化网络、公开可验证等特性，其可以作为可信的传输通道和数据计算验证平台[11]。

2.1 典型场景

数据合作表现形式有多种，在图 2基础上我们拓展系统在不同场景下的执行流程。

2.1.1 本地数据服务

场景描述 数据提供方在本地部署并管理相关的存储、计算服务器，数据使用方使用算法提供方的算法请求数据提供方对数据进行计算，并获得计算结果。

可信要求

- 数据不可篡改；
- 请求参数加密且不可篡改；
- 请求结果加密且不可篡改；

注意，由于在数据存储在本地，在数据管理运行的情况下，可以不加密原始数据，以保证计算性能。

执行流程

1. 数据注册：数据提供方传输原始数据至本地的 Fidelius 中间件（此处简称 FP），后者自动生成相关元数据（即原始数据的描述信息），同时元数据被发布至区块链网络；
2. 任务发布：数据使用方所在的 Fidelius 中间件（此处简称 FC）通过区块链获取元数据后提交给数据使用方，后者根据数据描述信息提供相应的数据分析任务（通常为执行程序的二进制文件），FC 将数据分析任务通过区块链网络转移到至 FP；
3. 数据计算：FP 对任务进行检测后，基于原始数据和任务完成数据计算，生成计算结果和相应的证明，并将加密后的计算结果和证明发布至区块链网络；

[11]关于区块链功能详细的介绍，参见附录 B

4. 结果返回：分析结果（加密）被保存在区块链网络中，区块链中的智能合约基于证明和结果进行验证，当计算结果通过验证后，FC 将计算结果解密后返回给数据使用方。

2.1.2 托管数据服务

场景描述 数据提供方出于成本、管理等考虑，本地没有相关的存储、计算服务器，而是委托给第三方（云服务厂商）进行数据的存储和计算；数据使用方使用算法提供方的算法请求第三方（数据托管服务方）对数据进行计算，并获得计算结果。

可信要求

- 数据对第三方不可见；
- 数据不可篡改；
- 数据使用请求需要经过数据提供方授权；
- 请求参数加密且不可篡改；
- 请求结果加密且不可篡改；

执行流程

1. 数据注册：数据提供方传输原始数据至本地的 Fidelius 中间件（此处简称 FP），后者对数据进行加密，自动生成相关元数据（即原始数据的描述信息）和权限证明，同时元数据被发布至区块链网络，加密数据发送至第三方；
2. 任务发布：数据使用方所在的 Fidelius 中间件（此处简称 FC）通过区块链获取元数据后提交给数据使用方，后者根据数据描述信息提供相应的数据分析任务（通常为执行程序的二进制文件），FC 将数据分析任务通过区块链网络转移到至 FP；
3. 数据授权：数据提供方根据区块链网络中的请求，对该次任务进行数据使用授权，并将该授权提交到区块链网络中；
4. 数据计算：第三方对任务进行检测后，基于加密数据、任务及数据授权完成数据计算，生成计算结果和相应的证明，并将加密后的计算结果和证明发布至区块链网络；

5. 结果返回：分析结果（加密）被保存在区块链网络中，区块链中的智能合约基于证明和结果进行验证，当计算结果通过验证后，FC 将计算结果解密后返回给数据使用方。

2.1.3 多数据源数据服务

场景描述 由于算法本身的复杂性或业务的要求，计算结果需要基于多个数据源得出，且需要多个数据源集中到同一个计算环境中。这一场景类似于前述的数据托管服务，不同的是存在多个数据源，且计算环境可能部署在某一个数据源。

可信要求

- 数据对第三方不可见；
- 数据不可篡改；
- 数据使用请求需要经过数据提供方授权；
- 请求参数加密且不可篡改；
- 请求结果加密且不可篡改；

执行流程

1. 数据注册：每个数据提供方传输原始数据至本地的 Fidelius 中间件（此处简称 FP），后者对数据进行加密，自动生成相关元数据（即原始数据的描述信息）和权限证明，同时元数据被发布至区块链网络，加密数据发送至第三方；
2. 任务发布：数据使用方所在的 Fidelius 中间件（此处简称 FC）通过区块链获取元数据后提交给数据使用方，后者根据数据描述信息提供相应的数据分析任务（通常为执行程序的二进制文件），FC 将数据分析任务通过区块链网络转移到至 FP；
3. 数据授权：每个数据提供方根据区块链网络中的请求，对该次任务进行数据使用授权，并将该授权提交到区块链网络中；
4. 数据计算：第三方对任务进行检测后，基于加密数据、任务及所有数据授权完成数据计算，生成计算结果和相应的证明，并将加密后的计算结果和证明发布至区块链网络；

5. 结果返回：分析结果（加密）被保存在区块链网络中，区块链中的智能合约基于证明和结果进行验证，当计算结果通过验证后，FC 将计算结果解密后返回给数据使用方。

2.1.4 模型服务

场景描述 算法本身包含了专业知识、或通过机器学习得到的训练参数，因此具有相当的商业价值，需要保护其中的关键参数。数据提供方与数据使用方为同一方，此处称为算法使用方，在算法提供方的许可下使用算法对数据进行处理。计算环境部署在算法使用方本地。

可信要求

- 模型参数不可见、且不可篡改；
- 算法使用方需要在算法提供方的许可下使用；

注意，此处计算结果不需要加密，数据也不需要机密性或不可篡改的检查。

执行流程

1. 模型注册：每个算法提供方传输参数数据至本地的 Fidelius 中间件（此处简称 FP），后者对模型参数进行加密，自动生成相关元数据（即原始数据的描述信息）和权限证明，同时元数据被发布至区块链网络，加密数据发送至第三方；
2. 任务发布：算法使用方所在的 Fidelius 中间件（此处简称 FC）通过区块链获取元数据后提交给数据使用方，后者根据算法描述信息及本地数据生成相应的数据分析请求；
3. 模型授权：算法提供方根据区块链网络中的请求，对该次任务进行数据使用授权，并将该授权提交到区块链网络中；
4. 数据计算：算法使用方对任务进行检测后，基于任务及模型授权完成数据计算，生成计算结果。

2.2 系统架构

图 3简述了系统各个模块以及其之间的关系。服务平台为各个模块提供入口，并

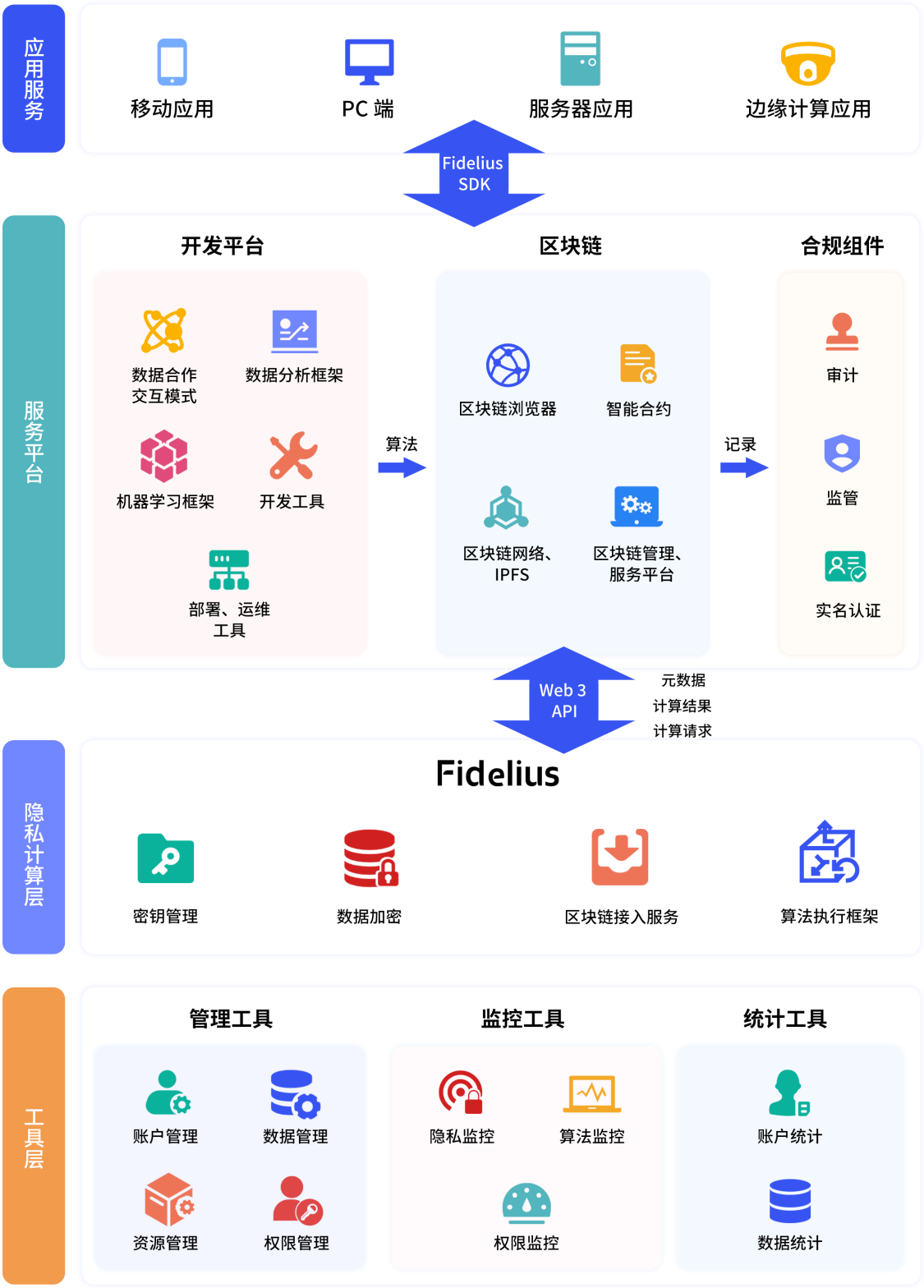
应用服务
移动应用
PC 端
服务器应用
边缘计算应用
Fidelius SDK
服务平台
开发平台
区块链
合规组件
数据合作交互模式
数据分析框架
机器学习框架
开发工具
部署、运维工具
算法
区块链浏览器
智能合约
区块链网络、IPFS
区块链管理、服务平台
记录
审计
监管
实名认证
Web 3 API
元数据
计算结果
计算请求
隐私计算层
Fidelius
密钥管理
数据加密
区块链接入服务
算法执行框架
工具层
管理工具
监控工具
统计工具
账户管理
数据管理
资源管理
权限管理
隐私监控
算法监控
权限监控
账户统计
数据统计

图 3: 系统架构

为上层应用提供接入方式。根据需要，服务平台的表现形式分为两部分，可供用户直接访问的网页和可用应用访问的 API。在服务平台内部，核心是隐私计算层及区块链，并且提供了其他组件，这些组件可以根据需求进行组合、定制。

隐私计算层 负责完成数据使用的交互过程，包括数据元文件的生成、更新，数据分析任务的生成、发布、执行，以及计算结果的交付。隐私计算层的核心是 Fidelius 隐私计算中间件，基于 TEE 可信执行环境，提供了数据的通用计算分析支持。

区块链 区块链提供了信息传输通道，保证了关键信息可溯源以及不可篡改。这些信息包括算法提供方发布的算法信息，数据提供方发布的数据信息，任务请求信息以及交付的结果信息。通过部署在区块链之上的智能合约，在提交任务的计算结果时，系统会对数据使用结果进行验证，保证数据、算法、参数以及结果四者的一致性，从根本上保证数据合作的可信性。

Fidelius 并不依赖于特定的区块链实现，因此能够部署在不同区块链系统上。当然，Fidelius 的性能，如吞吐量、响应时间等，受到区块链系统的影响，我们推荐使用高性能的 YeeZChain 作为底层区块链平台[12]

开发平台 为开发者提供了定制数据分析算法的工具，包括 1. 不同的数据分析框架，用于实现不同的业务场景；2. 开发工具，包括可信编译器、静态分析工具等，用于帮助开发者更便捷的完成开发任务；3. 部署、运维工具，用于帮助开发者将开发完成的算法提交到区块链上、监控算法的使用情况、以及对算法的使用进行授权。注意，受限于区块链的存储性能，算法本身并不需要存储在区块链上，而是将算法的关键信息存储在区块链上；算法本身可以使用额外的文件存储服务。

在某些场景下，该解决方案用于完成企业之间确定的数据合作任务，这些任务所需的算法是确定的，或者变动不频繁的，在这种情况下，可以一次性的部署所需的算法，并不需要完整的开发平台。

合规组件 得益于区块链不可篡改的特性，合规组件可以通过区块链中的的数据进行审计，包括对于发布的算法的审计、对于发布的数据的审计、对于数据使用方的审计等。额外的，还可以禁止不符合合规要求的算法、数据、使用请求等。

合规组件对于大多数场景而言是必要的，但是由于各个场景的需求不同，因此对于合规的要求也千差万别，因此我们也提供了对于合规组件的定制服务。

[12]关于 YeeZChain 详细介绍参见《YeeZchain v1.1 介绍文档》，https://gitlab.com/Yeez/yeez_introduction

工具层 在大多数场景下，隐私计算层是 “私有的”，即隐私计算层的相关信息是不愿与服务平台共享的，为此，我们为提供隐私计算的用户提供了相关的管理工具，这些工具包括对于隐私计算节点的用户管理、数据管理、资源管理及权限管理，还包括对于隐私计算节点中运行的任务的监控及统计。

3 Fidelius 隐私计算中间件

Fidelius 隐私计算中间件运行在隐私计算节点内，是整个解决方案的核心，本节我们对其进行详细介绍。

3.1 计算模型

Fidelius 隐私计算中间件（简称中间件）基于硬件 TEE 技术，不同的 TEE 实现对于运行在 TEE 中实例有不同的名称，为了方便描述，我们统一称之为 enclave（飞地），如同 ‘program（程序）’ 一词一样，根据上下文的不同，enclave 一词既可以表示一个二进制文件，也可以表示一个运行在 TEE 的实例。

中间件依赖于 TEE 的三个特性：

1. enclave 运行过程中内存不可见、不可修改；
2. enclave 代码不可修改；
3. 本地 enclave 之间的认证（attestation）。

图 图4所示为中间件的计算模型。每个隐私计算任务包括两个 enclave，分别为 EKeyMgr 和算法，其中 EKeyMgr 内置于中间件中，用于管理典钥及典钥相关的操作，算法由开发者提供，用于完成对数据的分析或使用。两个 enclave 通过 ECDH 建立加密信道，保证 enclave 之间的通信是安全的。

EKeyMgr 在安装部署之后，需要经过某种认证方式保证部署的 EKeyMgr 是正确的，这些认证方式包括但不限于：

- 使用 TEE 提供商提供的远程认证服务；
- 独立部署的远程认证服务；
- 经过培训、认证的操作员的人工核实。

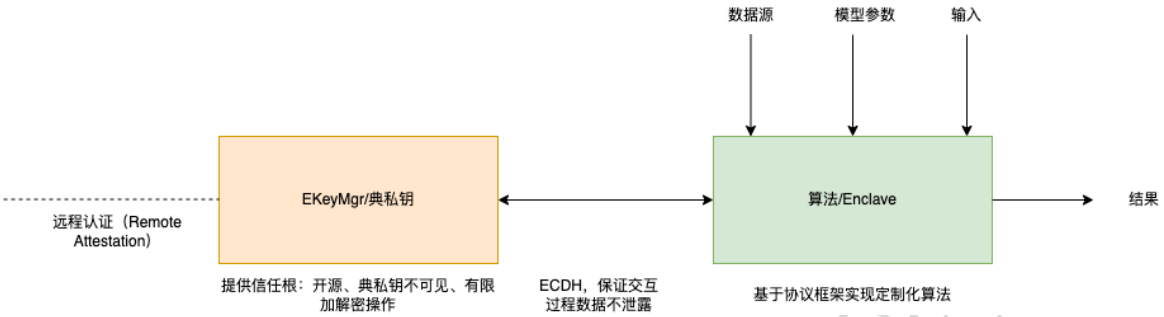

图 4: Fidelius 隐私计算中间件计算模型

具体认证方式视具体情况确定。

算法 enclave 包含了分析数据的业务逻辑，由不可信环境提供数据源、模型参数及输入参数，在 TEE 环境中处理得到最终的计算结果。

3.2 典钥与枢钥

在 Fidelius 中，EKeyMgr 通过典钥提供了信任根，典钥是一对椭圆曲线下的公私钥对，在 EkeyMgr 中生成，并在生成之后使用 TEE 内的密钥加密典私钥后存储在不可信环境中，因此典私钥只有 EKeyMgr 可见，可以认为典私钥绑定了对应的 EKeyMgr 或隐私计算节点。一个隐私计算节点存在至少一个典私钥。典公钥的原文公开、验证后，上传至区块链。尽管公开的典公钥可以用于任意的加密操作，EKeyMgr 在典私钥上提供的操作是有限的，EKeyMgr 不提供任意的解密操作，即使用典公钥加密的数据是不能解密并公开个不可信环境的。

枢钥是一对椭圆曲线下的公私钥对，与典钥不同的是，枢钥不在 TEE 环境中生成，而是在一般的环境下生成，例如浏览器、客户端、移动端等。枢钥与数据、算法模型、请求等业务相关，用户需要妥善保存自己的枢私钥，以免造成隐私泄露。

枢私钥转发 我们定义枢私钥转发的过程，枢私钥转发用于将枢私钥转移到目标算法 enclave 内，并且避免算法 enclave 所在的不可信计算环境获取该枢私钥。如图5所示为枢私钥转发流程：

1. 在本地生成枢钥，并妥善保管枢私钥；
2. 从智能合约（区块链）中获取经过认证的典公钥，注意，使用未经认证的典公钥会造成隐私泄露，因此不推荐；
3. 使用典公钥加密枢私钥，并对目标算法 enclave 进行签名；

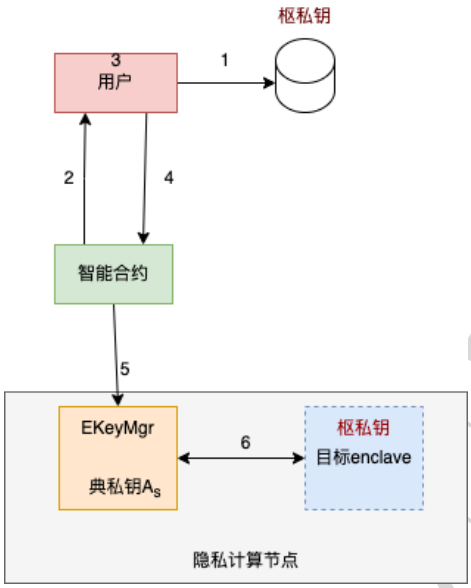

图 5: 枢私钥转发流程

4. 将加密后的枢私钥及签名提交到区块链上;

5. 隐私计算节点从区块链上获取加密的枢私钥及签名，并在 EKeyMgr 中解密枢私钥、验证签名;

6. 目标 enclave 通过加密信道，从 EKeyMgr 获得枢私钥。

注意，EKeyMgr 在获取到目标 enclave 的请求后，会验证目标 enclave 是否与签名提供的目标 enclave 一致，如果不一致，则会拒绝提供相应的枢私钥。

枢私钥转发提供了一种将“秘密”转移到目标 enclave 并且不会暴露的手段。与传统的通过远程认证建立可信信道的手段不同，Fidelius 优化了此流程，对算法 enclave 进行加速。

3.3 许可

许可是指在给定输入参数下，允许给定算法使用给定模型参数或给定数据源进行计算的密码学证明，按照用途分为模型参数许可以及数据源许可。

许可做为输入的一部分，由模型提供方或数据源提供方提供给算法。算法在开始执行实际的计算任务前，会对许可进行验证。当许可验证通过后，才会执行实际的计算任务。

一个许可包括三个部分:

- 数据源或模型参数的 Hash；
- 许可提供方的公钥，一般是枢公钥；
- 对于输入参数、数据源或模型参数的 Hash、计算节点的典公钥以及目前算法 Hash 的签名，签名私钥为枢私钥。

对于许可的检查是可选的。

3.4 Fidelius 隐私计算抽象

Fidelius 隐私计算中间件将一个计算任务表示为了一个 C++ 类，并通过模板参数提供了各个不同的配置以方便开发者定制不同的算法、满足不同的业务需求。

```
template<typename Crypto, typename DataSession, typename ParserT,
 typename Result, typename ModelT = void,
 template <typename> class DataAllowancePolicy = ignore_data_allowance,
 template <typename> class ModelAllowancePolicy = ignore_model_allowance>
class algo_warpper;
```

Listing 1: Fidelius 隐私计算中间件算法抽象

对于各个参数的说明如下：

- `Crypto`：密码算法簇，目前支持 `ypc::crypto::eth_sgx_crypto`，椭圆曲线为 SECP256k1，哈希函数为 SHA3-256，对称加密算法为 Rijndael128GCM，兼容以太坊。
- `DataSession`：数据源方式，支持无数据源（`noinput_data_stream`），单一未加密数据源（`raw_data_stream`），单一加密数据源（`sealed_data_stream`），多数据源（`multi_data_stream`）且其中每一个数据源都是加密数据源。
- `ParserT`：表示自定义的算法类，由开发者自行开发。
- `Result`：表示结果的类型，支持本地结果（`local_result`，结果不加密），上链结果（`onchain_result`，结果的数据大小能够上链），链下结果交付（`offchain_result`，结果的数据大小不能够上链），结果加密转发（`forward_result`）。
- `ModelT`：表示模型参数的类型，是 `ff::util::ntobject<...>`，由开发者指定，默认为 `void`，即没有模型参数。

- DataAllowancePolicy，表示数据源的许可验证策略，支持不检查数据源的许可（ignore_data_allowance）或检查数据源的许可（check_data_allowance）。
- ModelAllowancePolicy，表示模型的许可验证策略，支持不检查模型参数的许可（ignore_model_allowance）或检查模型参数的许可（check_model_allowance）。

通过选取不同的参数，中间件能够支持不同的场景。以常见的数据可用不可见为例，数据在数据提供方本地存储，且数据提供方提供相应的隐私计算节点，数据请求方的请求经过加密后提供给隐私计算节点，且计算的最终结果加密后提交到区块链上，此时的算法实现如下：

```
ypc::algo_warpper<ypc::crypto::eth_sgx_crypto,
    ypc::raw_data_stream,
    enclave_iris_means_parser,
    ypc::onchain_result<ypc::crypto::eth_sgx_crypto>
> pw;
YPC_PARSER_IMPL(pw);
```

其中 YPC_PARSER_IMPL 是一个宏，用于隐藏相关的代码实现。

附录 A 相关技术、产品

整个隐私保护产业目前仍处于早期发展阶段。本章简单介绍下业界已有的相关技术和解决方案。

A.1 可信执行环境

可信执行环境（TEE，Trusted Execution Environment），它是电子设备（PC、智能手机、平板电脑等设备）CPU 上的一块区域。这块区域的作用是给数据和代码的执行提供一个更安全的空间，并保证它们的机密性和完整性。TEE 最早来自于移动终端开放组织（OMTP）在 2006 年提出一种解决方案：在同一个智能终端下，除了多媒体操作系统外再提供一个隔离的安全操作系统，这一运行在隔离的硬件之上的隔离安全操作系统，用来专门处理敏感信息以保证信息的安全。

目前提供 TEE 产品的主要来自英特尔、AMD、ARM 等芯片厂商。

- SGX：Intel SGX [13]是 Intel 架构新的扩展，在原有架构上增加了一组新的指令集和内存访问机制。这些扩展允许应用程序实现一个被称为 enclave 的容器，在应

[13]https://software.intel.com/content/www/us/en/develop/topics/software-guard-extensions.html

用程序的地址空间中划分出一块被保护的区域, 为容器内的代码和数据提供机密性和完整性的保护, 免受拥有特殊权限的恶意软件的破坏。

- TrustZone：TrustZone [14]是 ARM 公司提出的一种 TEE 实现方案，其将 SoC 的硬件和软件资源划分为安全和非安全两个区域。当设备执行隐私相关的操作时，比如指纹识别、密码处理、数据加解密、安全认证等，会在安全区域执行，其余操作在非安全区域执行，比如用户操作系统、各种应用程序等。

附录 B 为什么隐私保护需要区块链

B.1 什么是区块链

区块链技术最初是被应用在比特币 [11] 系统中，随后在密码货币领域得到快速发展，并迅速扩展到物联网、知识产权保护等领域，被认为是继移动互联网之后的第五代互联网颠覆性技术。

区块链并不是一种单一的技术，而是多种技术整合的结果，包括但不限于分布式存储、共识机制、智能合约、对称/不对称加密等等。这些技术以新的结构组合在一起，形成了一种新的数据记录、存储和表达的方式。

从一般结构出发，区块链满足的基本特点有：

- 自治（透明）：系统节点对等，可以自由选择加入或离开；去中心化，无管理机构或第三方仲裁。
- 分布式（共享）：只需要连接到最近的节点就可以获取所有账本信息，同样交易发布也仅需提交给临近节点，依次转发下去直至遍及整个网络。
- 不可篡改（永久）：基于密码学技术保证上链交易合法，每个节点都本地同步一份附有时间戳的账本副本，保证数据不可篡改。
- 按合约执行（公平）：所有节点都按照一个规则或合约行事并达到共识（例如智能合约或账本同步）。

B.2 区块链与 Fidelius 结合

在区块链中，所有数据 (包括交易类别、交易双方的地址、交易金额等) 都是公开的，这在一定程度上提高了参与者对数据真实可靠的信心。大多数区块链系统会将

[14]https://developer.arm.com/ip-products/security-ip/trustzone

所有交易数据记录在公共账本中，任何用户均可查询。一笔交易的有效性需要经过区块链中大部分节点的认可，而验证结果取决于交易数据 (例如交易金额，携带数据及其签名)。

早期区块链系统（以公链为代表），依赖于公开透明的交易，所有节点同步账本，这种方式导致了严重的隐私泄露。此后产生了许多具有匿名区块链系统，例如 ZCash[15]、Monero[16]等等，其主要采用了混淆服务和环签名等技术。上述匿名技术主要是用于保护用户身份隐私，这一方面是由于公链属于非许可区块链，用户对于匿名的需求更强烈；另一方面，大部分公链所承载的应用仍然为数字货币交易，即链上行为多数为普通转账交易，在此基础上，用户身份匿名化即可满足隐私保护需求。

然而联盟链的应用场景更为广泛，其链上主要数据不再是代币（Token）转账信息，更多的是业务相关数据，此时公开所有数据在某些关键应用场景下是不可行的，例如金融、医疗等数据敏感性高的场合。因此，**区块链本身并不能直接满足交易数据的隐私保护需求**。

近年来也涌现出一些隐私保护技术，可以对区块链的更多交易信息（例如智能合约）实现隐藏，例如在附录 A中我们介绍的 zk-SNARK、WeDPR 等技术。然而在 Fidelius 的设计哲学中，我们并没有将 Fidelius 设计成与区块链紧耦合的关系，这是因为：

- Fidelius 的初衷是为了解决数据合作的隐私保护问题，而不仅仅局限于区块链上的数据隐匿；
- 我们发现当前大部分企业很难实现业务全部上链，这一方面是由于区块链作为底层基础设施仍然存在许多缺陷尚待解决[17]，另一方面企业往往已经有成熟的数据存储平台（数据中心或者云存储），没有必要再基于区块链进行数据存储。

尽管如此，Fidelius 中也使用了区块链技术，这里区块链主要起到了如下作用：

- 可信第三方：如图 2所示，数据提供方和数据使用方之间基于区块链实现了交互，由于共识机制本身的去中心化特性，实际上区块链扮演了可信第三方的作用。具体来说，区块链能够 1) 存储数据，且数据不会被篡改，2) 传输数据，且不依赖于中心化的证书机制（例如 HTTPS 所依赖的证书）[18]，3) 验证数据，即对数据使用情况及分析结果的验证。

[15]ZCash，https://z.cash/zh/

[16]Monero，https://www.getmonero.org/

[17]例如受共识机制、节点规模、存储等影响，区块链的吞吐量和扩展性仍然不能满足企业级业务的需求

[18]并且在参与主体规模增加的情况下区块链对传输开销的增加也非常低，若采用专线传输，开销则是指数级别的增长

- 抵抗作弊：由于区块链的特点，被区块链记录的数据合作行为是公开、且不可篡改的，这对于试图通过“刷单”等来影响数据合作的作弊行为有非常好的抵抗作用。虽然不同于在中心化的数据合作平台上处理作弊行为的“黑盒子”，公开的、不可篡改的数据合作记录可以被不断更新的自动化算法或人力检查，是一种有效的抵抗作弊的手段。

参考文献

[1] V. Ciriani, S. D. C. Di Vimercati, S. Foresti, and P. Samarati, "k-anonymous data mining: A survey," in Privacy-preserving data mining, pp. 105–136, Springer, 2008.

[2] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, no. 1, pp. 3–es, 2007.

[3] M.-L. Akkar and C. Giraud, "An implementation of DES and AES, secure against some attacks," in International Workshop on Cryptographic Hardware and Embedded Systems, pp. 309–318, Springer, 2001.

[4] U. Somani, K. Lakhani, and M. Mundra, "Implementing digital signature with rsa encryption algorithm to enhance the data security of cloud in cloud computing," in 2010 First International Conference On Parallel, Distributed and Grid Computing (PDGC 2010), pp. 211–216, IEEE, 2010.

[5] Y. Tsiounis and M. Yung, "On the security of elgamal based encryption," in International Workshop on Public Key Cryptography, pp. 117–134, Springer, 1998.

[6] C. Gentry and D. Boneh, A fully homomorphic encryption scheme, vol. 20. Stanford university Stanford, 2009.

[7] C. Dwork, "Differential privacy: A survey of results," in International conference on theory and applications of models of computation, pp. 1–19, Springer, 2008.

[8] Y. Lindell, "Secure multiparty computation for privacy preserving data mining," in Encyclopedia of Data Warehousing and Mining, pp. 1005–1009, IGI Global, 2005.

[9] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted execution environment: what it is, and what it is not," in 2015 IEEE Trustcom/BigDataSE/ISPA, vol. 1, pp. 57–64, IEEE, 2015.

[10] W. Dai, C. Dai, K.-K. R. Choo, C. Cui, D. Zou, and H. Jin, "SDTE: A secure blockchain-based data trading ecosystem," IEEE Transactions on Information Forensics and Security, vol. 15, pp. 725–737, 2019.

[11] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," tech. rep., Manubot, 2019.