技术白皮书 **v1.9.9**

2020 年 11 月

Crust

Ⅰ 概述

A） 去中心化系统和云计算服务的新机遇

去中心化账本技术指的是交易记账由分布在不同地方的多个节点共同完成，这些节点都可以参与监督交易合法性，同时也可以共同为其作证。区块链是分布式账本技术的一种形式，区块链分布在点对点网络上并由其管理。由于它是一个分布式账本，因此可以在没有中央服务器管理的情况下运行，并且可以通过数据库复制和信任计算来维护其数据质量。但是，区块链的结构使它有别于其他类型的分布式账本。区块链上的数据被分组并以块的形式组织起来，这些块按照时间顺序依次连接形成一条链，并使用密码学技术对其进行安全保护。基于区块链这种分布式数据结构，可以为零信任的去中心化网络带来共识。

能否解决实际社会问题，以及能否真正提升社会生产活动的效率，是技术能否持续发展和普及的关键衡量标准。信息技术对于社会生产关系与生产效率的影响是已被验证的，且正处于持续深化的进程中，而存储和计算恰是信息技术生产力革命的两大核心基础要素。因此，区块链技术不仅需要提供凝结共识、去信任化的机制（即“价值去中心化”），也应当为存储、计算两大核心生产要素提供去中心化的基础设施（即“存储去中心化”与“计算去中心化”）。纵观现有的主流去中心化共识机制，区块链技术被广泛用于信任凝聚（生产关系）基础设施，并且往往伴随着大量算力以及存储资源的消耗，而对存储和计算去中心化的支撑则相对空缺。

现有的存储和计算场景广泛地承载于中心化的云计算平台，而存储服务作为云计算市场最重要组成部分之一，同时也成为了大多数云服务的基石。中心化云存储旨在将储存资源放在中心化磁盘阵列上，使用者可以在任何时间、任何地方，透过任何可联网的装置连接到云上方便地存取数据。但是，在这种中心化服务的方式下，存在服务稳定性不足、网络带宽成本高、数据传输能力有限等问题。本项目旨在以区块链的信任凝聚出发，通过技术的创新优化，从去中心化存储场景切入，进而实现可信、可靠、高效、泛在的去中心化云服务生态。

B） Crust 概述

Crust 是基于有意义工作量证明机制（Meaningful Proof of Work, 简称 MPoW）和担保权益证明共识（Guaranteed Proof of Stake，简称 GPoS）构建的数字加密应用层，同时也是一种支持去中心化存储与计算的新一代区块链技术。

Crust 网络是一个高安全性、低能耗且公平开放的网络。从去中心化存储场景切入，通过 GPoS，Crust 将凝聚的共识基础反哺于去中心化存储，使任何人都可以简单、公平地利用闲置存储设备参与去中心化文件系统的构建，支持对有意义的数据进行高效、安全以及低成本地存取和处理。

MPoW 的灵活性，决定了 Crust 的生态设计除了可以将共识凝聚和去中心化存储结合，还可以将共识凝聚和去中心化计算相结合。从去中心化存储出发， Crust 从技术上为激励层（共识）+网络层+ 持久层（存储）+应用层（计算）的全栈生态无缝过渡提供了可能性。

II 一种基于可信执行环境技术的有意义工作量证明机制 MPoW

A） 可信执行环境 TEE

可信计算（Trusted Computing）是在计算和通信系统中应用基于硬件安全模块支持的可信计算平台，以提高系统的安全性。随着可信计算研究的不断深入，大众视线逐渐由传统硬件芯片安全模式转向了可信执行环境（TEE，Trusted Execution Environment）。TEE 是由 Global Platform 提出的概念，目前 TEE 有着多样化的实现方案，其中基于 Intel 芯片的 SGX 以及基于 ARM 开源框架的 TrustZone 是可信执行环境技术实现中最被广泛认知且应用的。

可信执行环境是一种由多种计算机相关技术组合而成的安全技术， 以下 5 个技术概念是可信执行环境的核心规范：

1. Endorsement key 签注密钥

签注密钥必须随机生成并且不能被改变。其中私有密钥必须被安全保存，除了指定接口可以调用，无法通过任何方式获得。而公共密钥用来认证及加密待发送的敏感数据。

2. Secure input and output 安全输入输出

输入输出是指用户与系统之间的交互，其途径包括键盘、外设、网络接口等。安全输入输出是指，从系统用户到访问的进程间存在一条受保护的路径。

3. Memory curtaining 储存器屏蔽

储存器屏蔽拓展了一般的储存保护技术，提供了完全独立的储存区域。即便是操作系统自身也没有屏蔽区的完全访问权限，因此入侵者即便控制了操作系统，运行时（Run Time）的数据也是安全的。

4. Sealed storage 密封存储

密封存储通过把私有信息和用户使用的平台环境配置信息捆绑在一起来保护私有信息。意味着被密封存储的数据只能在相同的安全环境下读取。

5. Remote attestation 远程认证

远程认证是指，由签注秘钥生成当前系统的软件证明书，系统上的任何改变可以通过证明书被远程授权方感知和校验，从而使得系统的执行逻辑安全可信。

以上 5 个关键技术是一个完备的 TEE 技术方案所应该拥有的。目前主流的 TEE 技术是基于硬件芯片的 Intel SGX 和 ARM 开源框架 TrustZone，Crust 目前对两种主流解决方案以及基于 TPM（Trusted Platform Module）的 TEE 软件实现都有支持。由于 SGX

被更广泛地应用于 PC 端，并且拥有相对更高的安全性，后文的 TEE 技术阐述以 SGX 为主。

相比于复杂的算法层面解决方案，TEE 在实现逻辑上更加简单而有效。在技术发展方面，TEE 拥有快速发展的技术生态，并且有着持续发展的强劲动力。在功能方面，TEE 支持复杂计算逻辑的可信执行，这更加契合了 Crust 的技术愿景，即在实现去中心化存储的基础上进一步支撑去中心化计算，进而形成完备的去中心化云服务生态。

B） MPoW 机制

在区块链系统中没有像银行一样的中心化机构，所以在进行传输信息、价值转移时，共识机制解决并保证每一笔交易在所有记帐节点上的一致性和正确性问题。区块链的共识机制使其在不依靠中心化组织的情况下，依然大规模协作完成运转。目前主流的区块链竞争共识机制如 PoW、PoC 等，往往需要基于特定计算或存储过程产生的工作量，这些计算或存储过程被普遍认为是无意义的。Crust 通过结合 TEE 技术，立足于去中心化云计算和校验的场景提出了独创的 MPoW 机制。MPoW 可以被用来安全、公平、高效地量化各种有意义的数据存储和计算工作。

MPoW 机制主要负责节点工作量的统计和环境验证。我们将从存储场景出发说明这两个功能和相关流程：

工作量的统计：

节点接收到分发的数据，存储到硬盘。当用户数据被存储后，在本地 TEE 内执行定期抽查程序，校验 Merkle Hash 来确定节点申明的存储空间被用来正确保存用户文件

环境验证：

节点 TEE 内运行检查程序，对网络内其它节点的 TEE 环境信息以及可信执行代码版本信息的远程认证逻辑。

可以看出，数据的完整性检查、存储的验证和统计、节点环境的检查以及节点身份验证均受到 TEE 的保护。

MPoW 具有以下优点：

透明性：存储机制公开透明。

公平性：对工作节点工作量以及奖励的计算均受到 TEE 的保护，节点无需担心工作量得不到应有回报，同时也无法通过作弊获得额外奖励。

高效性：存储量的证明无需进行大量冗余的挑战，也无需存储任何无意义的数据。无论计算资源还是存储资源都能被高效利用。

发展性：TEE 支持完备的计算，并且有着不断发展的势能。这就意味着 Crust 区块链生态可以基于 MPoW 实现更加强大的功能，保障了从存储共识到计算共识演进的可行性和发展性。

III Crust 网络

Crust 网络是一个无限横向扩展，节点可以自由进出的的 P2P 对等网络。本章将从 Crust 节点和网络构建，以及 Crust 技术架构两方面对 Crust 网络进行介绍。

A） 节点的功能

有意义的工作

有意义的工作意味着，节点能够提供有效的存储和计算资源用以满足真实的存储和计算需求。在网络构建初期，Crust 致力于构建去中心化存储网络。因此，下文将以去中心化存储网络为目标场景进行叙述。在去中心化存储网络中，节点主要负责存储用户数据。节点工作量奖励一方面来源于用户的存储空间租用，另一方面来源于贡献存储空间而获得的区块链奖励。

工作量证明

节点在硬件上需要支持 TEE，提供存储空间并运行符合 MPoW 机制的软件或程序。为了保证用户数据被完整存储，节点在每个区块周期需要对已存储的文件对象进行 Merkle Hash 片段自抽查，并在 TEE 内生成 TEE 存储声明报告。由于抽查机制被写入 TEE 内，抽查自检的流程无法被操作系统层面中断且不可被篡改，因此，每个节点的工作量统计置信程度相当于 TEE 技术的安全程度。

区块中记录的存储工作量来源于 TEE 存储声明报告。TEE 存储声明报告是 Crust 区块链上存储量记录的基本单位。其中包含节点的存储量信息以及一个来自本地 TEE 的签名。

节点环境的验证

节点需要负责其它新节点身份验证、TEE 的验证以及工作量的验证。节点运行支持 MPoW 的镜像，执行如下步骤：

1. 根据收到的 TEE 存储声明报告进行工作量统计
2. 验证网络中其它节点的 TEE 信息
3. 接收、验证并打包 TEE 存储报告到链上
4. 接收、验证并打包存储租赁合约到链上
5. 接收、验证并打包其它交易信息到链上

在TEE的保护下，某个节点对其它节点的验证是可信的，携带恶意、作弊代码的TEE节点理论上均无法加入网络。

节点的入网

初始状态，全网存在若干个初始节点，节点的 TEE 包含了验证节点所需的逻辑。
由于链上需要维护 TEE 节点的公钥证书，节点入网流程如下：

1.拉取网络中现存节点的公钥及相关信息
2.通过 TEE 的远程证明技术跟网络中的某个节点相互验证
3.公布验证结果，其他节点对验证后打包上链生效
4.提供本节点 TEE 生成的公钥，并写入链上

节点的服务

节点可以提供存储和检索服务。通过 IPFS 协议，节点可以将用户的有意义文件复制至本地进行存储，也可以响应用户或网络中其它节点的检索请求，交换自己拥有的文件或文件块。

组网逻辑

由于 Crust 网络的节点存在区块链共识和有意义的工作两种不同功能，Crust 网络被天然地包含了数据存储和验证出块双重网络。Crust 的存储层适配多种分布式存储协议如 IPFS 以及 DAT 等 P2P 网络架构和 DHT 技术，用于快速、稳健地存储和分发数据块。验证网络则主要负责验证节点信息以及维护区块链数据。

每个节点在申请入网时，已入网的节点需要对该节点所启动的 TEE 实例进行验证，验证结果会被记录到链上。TEE 实例一旦重启或者销毁，节点需要重新验证入网。上链信息主要由一个四元组组成：

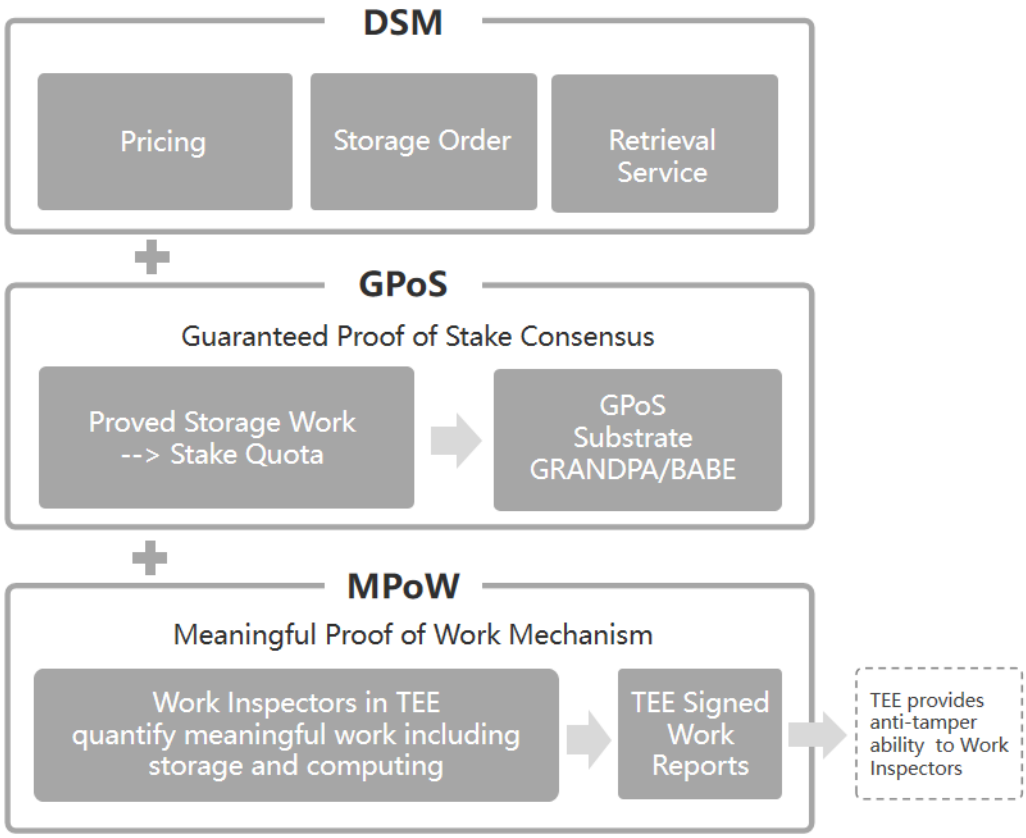
$$\{R, Sig_M(R), Sig_V(R), Sig_M\big(Sig_V(R)\big)\}$$

其中 R 是被验证节点报告，其中包括节点可信执行环境信息、节点声明的存储量以及空盘证明。Sig 代表签名运算，V 和 M 分别代表背书节点和入网节点。这个四元组确保了每个节点存在唯一的背书节点。

当节点的存储状态发生变化，比如有用户数据存储或者存储量发生改变时，需要在 TEE 内校验外部存储状态变化并更新存储声明报告，同时将新的存储状态上链。

B） 技术架构

Crust 包含了工作量证明层 MPoW、区块链共识层 GPoS 以及分布式云存储/计算层。



MPoW

工作量证明机制 MPoW 建立在 TEE 基础上，为代码的可信执行提供技术保证，TEE 技术为 MPoW 机制提供以下支持：

1.存储隔离区的安全计算

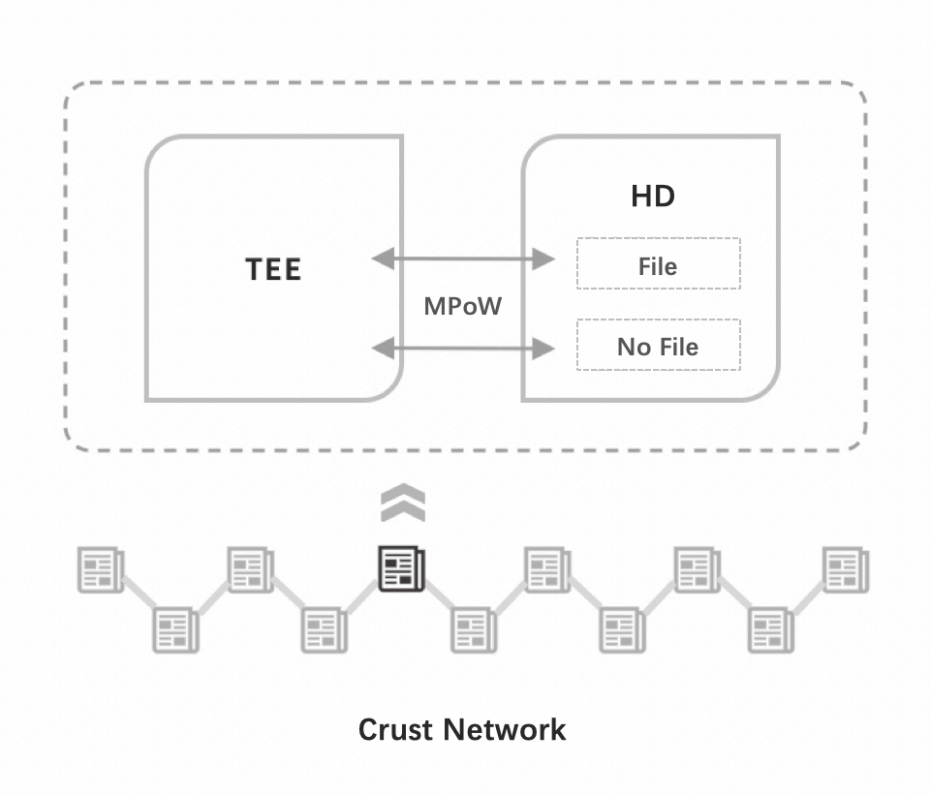
存储隔离区内的数据无法被外部进程获取

2.TEE 与节点身份（公钥）的绑定

TEE 生成的公钥通过远程证明，能唯一关联节点的合法性

3.隐私和敏感数据的加密封存

对特殊的隐私和敏感数据，可在 TEE 存储隔离区进行处理，但在传输和存储的过程中均处于加密状态，节点无法进行观察获取。



MPoW 机制包含两层证明：环境证明和工作量证明。

1.环境共识：

节点入网时需要基于 MPoW 机制，对节点的 TEE 进行共识。Crust 网络中节点对待入网节点的环境进行验证，通过验证的节点身份以及其 TEE 公钥将会被记录在链上。

2.工作量共识：

a) 每隔一个随机周期，Crust 节点的存储量和存储状态都会被节点本地的 TEE 随机抽查。MPoW 的封装和验证逻辑也是由本地 TEE 处理的。Crust 存储节点收到用户文件后，在 TEE 内执行重加密封装并保存，这样，外存中的文件只有 TEE 能还原，节点无法进行女巫攻击。每个周期 TEE 在快速本地存储验证后，签署一个工作量报告上链，链上其它节点只需要验证工作量报告的签名即可，极大简化了存储量共识流程。因此，相比与基于复杂远程挑战算法的验证，基于 TEE 的验证降低了对网络和计算资源的占用。

b) 基于 MPoW，还可以对工作节点的计算工作量进行统计、验证和共识。Crust 提出了一种工作量共识算法 PoRT（Proof of Running Tracking），通过将可信执行环境与 LXC（Linux Container）结合，可以实现对工作节点计算量进行统计并达成共识。

GPoS

在整个 Crust 系统里有多个参与方，它们各自有不同的需求，按照每个角色参与的方式，我们将它们分为：验证人、候选人、担保人、用户，在此文中提到的用户，主要指存储和计算资源用户。

1. 验证人

验证人是 Crust 网络中打包并生成区块的节点，维护着整个区块链网络。同时根据 Crust 网络的 GPoS（Guranteed Proof of Stake）共识，验证人节点需要有存储资源作

为担保，并可 Stake 相应额度的 CRU 通证（Crust 网络中的原生通证，在下一章节详细介绍），且需要保持在线。所以验证人节点也是一个提供存储资源的节点。参与到网络中的验证人节点可以获得单独给于打包区块的奖励和区块链每个周期的奖励分成，且要承担被罚没资产风险。验证人也可以通过存储交易市场出让存储资源获得收益。

2. 候选人

候选人是 Crust 网络中参与竞争成为验证人，但没有获得验证资格的节点。和验证人节点一样，候选人节点也需要有存储资源作为担保，并可 Stake 相应额度的 CRU 通证，且需要保持在线。和验证人节点的区别是，候选人节点不参与生成区块，不能获得单独给于生成区块节点的奖励。候选人节点可以获得区块链每个周期的奖励分成，同时也可以通过存储交易市场出让存储资源获得收益。候选人和验证人并不是固定的，每一个周期它们的身份可能产生变化，主要依据每个周期末节点 stake 的通证数量决定。

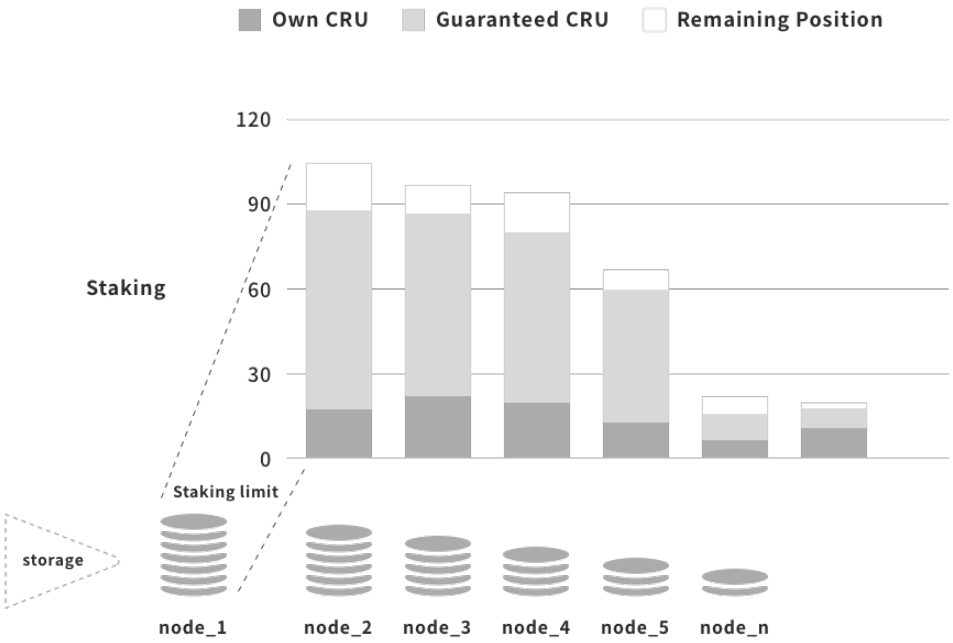
3. 担保人

担保人是 Crust 网络中为任意一个或者多个节点提供担保的账户。拥有 CRU 通证的账户都可以成为担保人，可将其 CRU 作为担保资产。担保人为节点提供担保可以获得担保收入。

4. 用户

用户是指使用 Crust 网络资源的消费方，主要指存储和计算资源的使用用户。会使用 CRU 通证或者 Crust 网络中支持的其它通证资产购买资源服务。

Crust 链使用的是 GPoS（Guaranteed Proof of Stake）共识机制，是一个以存储资源做担保额度的 PoS 共识。和现有的 PoS 项目类似，节点需要将 CRU 通证质押来竞争成为验证人，不一样的是节点还需要提供存储资源以获取相应的担保额度，有了担保额度才能 Stake 相应数量的 CRU。通过 MPoW 机制，节点的存储量的节点存储量监测机制，节点贡献的存储资源越多，能获得的抵押额便越高。



节点可以通过提供以下两类文件的存储证明来获得质押额度：

第一类是有意义的用户订单文件，存储有意义文件可以提高 Crust 网络的可用性和服务能力；（详情请参阅 DSM 描述）

第二类是无意义空盘证明文件；（详细请参阅第 IV 章）

GPoS 基于 Substrate 框架的 BABE/GRANDPA 算法进行最终出块。如果想从共识上攻击 Crust 网络，除了需要拥有大比例的 CRU 通证，还需要控制足够多的存储资源，这样的设计会让攻击难度变得相对更高。

DSM

Crust 的 DSM（Decentralized Storage Market，存储市场）旨在为基于 Crust 网络的应用和平台提供优质的存储服务。其中，Crust 的存储服务，主要包括存储订单机制和检索机制。

1，定价机制

Crust 网络中，用户不是针对某个节点，而是对整个网络签署存储订单。在这一模式下，用户存储订单生成时，网络会根据当前存储供需状态计算出一个价格。详情参考《Crust 经济白皮书》

2，存储订单机制

DSM 为用户提供了存储订单入口，用户将自己的文件以付费的形式在 Crust 网络中可靠的长期保存。

Crust 的存储订单机制是基于存储资源池（pool based）的订单模式。在这种模式下，用户对 Crust 网络发起订单，其中包括用户的存储需求和待存储文件的摘要，用户的

订单费用一部分支付给全网的奖励池，用于分发给全网提供 CRU 通证 staking 的节点；另一部分支付给用户所存文件的奖励池，用于分发给提供了此文件的存储证明的商户。

Crust 网络中节点可以通过 IPFS 文件协议获取到用户文件并保存到本地，经过本地 MPoW 机制的封装、校验和证明，节点可以在第一时间发起文件的声明。所有提供存储证明的节点将按顺序进入文件奖励序列，其中排名靠前的节点将会获得文件奖励池的奖励。

节点获取文件的速度以及 MPoW 证明的速度将会影响节点接收订单的能力。Crust 网络节点通过 BitSwap 的信用机制使得优质节点能更快获得用户订单文件（详细请参阅第 IV 章）。

Crust 网络的存储服务主要适配星际文件系统（Inter Planetary File System，简称 IPFS）和分布式哈希表（Distributed Hash Table，简称 DHT）等技术，实现了基本的数据完整性、内容寻址、防篡改和去重等功能。不同的地方在于，基于 MPoW 存储量统计和校验可以在本地 TEE 进行，增加了工作量统计的效率和可靠性。

除了基本的存储功能之外，Crust 网络在隐私保护上有进一步的探索。基于 TEE 的 Crust 存储网络可以支持在节点的可信空间（Enclave）之间建立加密通道以及数据的加密封存（Seal）。用户的隐私数据可以选择在加密通道间传输并被加密存储。被这种方式加密的用户数据将无法被除了用户以外的任何人（包括存储节点本身）获取。

3，检索服务

Crust 网络中检索分为两种：来自用户的检索需求，以及来自节点的检索需求。

用户的检索需求，代表了用户的数据使用需求，也是 Crust 网络之上存储应用的价值体现。

节点同样拥有文件的检索需求，一方面来自节点对新用户订单奖励的竞争；另一方面，存储有意义文件可以提升节点的 Stake Limit。

Crust 网络节点参考 BitSwap 的信用博弈机制，即向更多帮助过自己的节点提供更多的检索服务。博弈的结果是，提供检索的节点将更容易从其它节点处检索到数据（详细请参阅第 IV 章）。

在这样的博弈机制下，有意义文件的存储和检索形成一个激励循环。节点的收益取决于节点能否更快的检索到文件，而节点在 Crust 网络中检索文件的速度又取决于节点在之前对其它检索请求的响应。Crust 网络的节点会尽可能多的存储用户数据，并增加数据下载速度，提高了用户的使用体验。

IV 技术实现

A) Crust 的远程验证

远程验证机制解决了软件执行的可靠性问题，是 TEE 抵御恶意行为的重要功能。在 Crust 中，远程验证同样是去中心化网络组建的核心。被节点通过在远程验证的过程中嵌入当前运行 TEE 的公钥，将节点的身份、执行逻辑以及平台参数与 TEE 公钥在区块链上关联起来。远程验证由 Crust 网络中任意节点发起，先后要求被校验者证明：

1. 它的身份

2. 它运行的逻辑未被篡改

3. 它在一个正版平台上运行，并且启用了英特尔 SGX

MPoW 远程证明的基本流程为：

1. 首先由挑战者向被验证平台提出证明请求，包括一个随机数用来防止重放攻击；

2. 被验证平台随之搜集在芯片制造时写入的背书证书（Endorsement Key，简称 EK）等信息用来标识可信平台的唯一身份，并用 EK 生成平台身份密钥（Application Identity Key，简称 AIK）来避免暴露隐私，然后将 EK 发送给隐私签证机构（Privacy Certification Agency，PCA）；

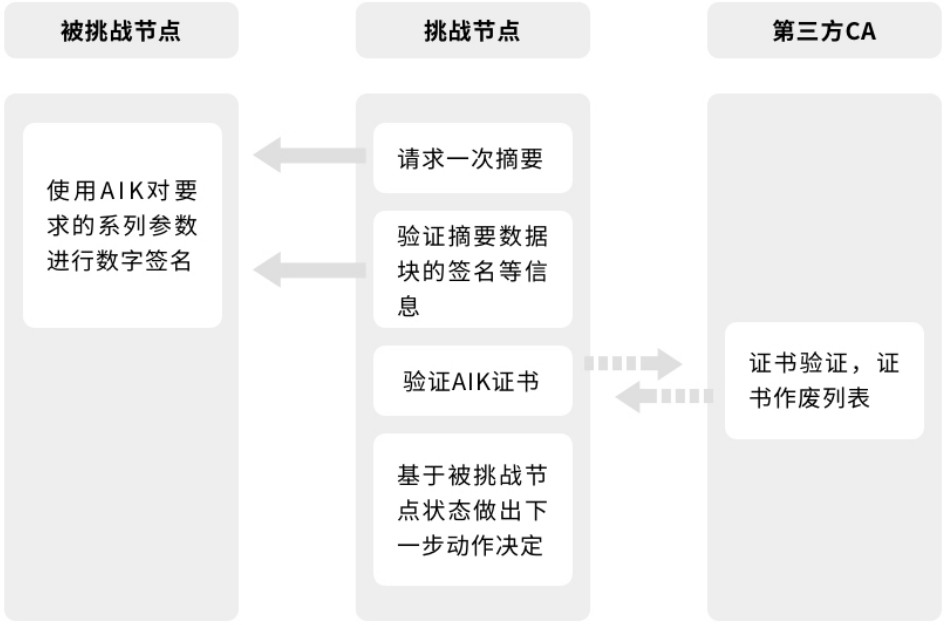
3. PCA 通过验证 EK，验证芯片的合法性，并对 AIK 颁发证书；

4. 被证实平台通过引证（Quote）操作，使用 AIK 对软件度量值进行签名，并随之将签名值与度量日志和 AIK 证书发送给挑战者；

5. 挑战者首先验证 AIK 证书的有效性，使用 AIK 公钥对数据进行解密获取软件度量值，通过软件度量值确保度量日志真实可信地返回给了挑战者，随后将度量日志的每一项与预期值进行比对，判断平台是否可信；

6. 挑战者将被挑战者的 AIK 公钥写入区块链

注：流程中的 1、2、3 又叫作初始化阶段，流程 4、5 叫作证明阶段



以上流程中生成的 AIK 私钥将会被节点保存在 TEE 的存储器屏蔽区，只能被 TEE 的可信执行程序安全访问。执行在节点 TEE 内的可信执行程序的运行结果，将会被 AIK 私钥签名，并且该结果可以被链上记录的 AIK 公钥验证。

B） 去中心化存储的支持

Crust 兼容了去中心化存储技术如 IPFS 的 P2P 基础网络架构和 DHT 技术，用于快速、稳健的存储和分发数据块。同时， Crust 在智能冗余、结构化数据支持、监管机制、文件加密和权限管理上做了一定的拓展和优化。

C） 数据的存储证明

一套完善可靠的数据存储证明机制需要包含数据完整性验证机制和数据时空验证机制。在 MPoW 中，数据完整性验证主要基于 MPoW 的 TEE 数据校验实现，数据时空验证十分类似经典的 PoSt 算法。

Filecoin 关于时空证明 PoSt 和复制证明 PoRep 给出了一系列定义，其本质逻辑是使得有效的证明人 P 能够说服一个验证者 V 相信 P 在一段时间内已经存储了一些数据 D。MPoW 通过 TEE 技术实现了本地存储的自验证，有效减小了复制证明的复杂性，在抵御女巫攻击、外包攻击以及生成攻击的同时，一定程度上简化了现有的 PoSt 流程，降低了网络成本和计算成本。

D） 空盘证明

为了度量节点的存储供应量，我们定义了空盘证明机制，使得节点可以在 TEE 内有效地追踪节点声明的存储空间。

节点 TEE 内随机生成无意义数据块δ，并用δ写满可用空间。由δ构成的存储证明 r 将会被 TEE 和链追溯，TEE 会定期根据 r 对本地存储进行抽查校验，确保声明的存储量可用。

E） 数据封存

为了抵御生成攻击和女巫攻击，Crust 会在 TEE 内对用户文件进行封存（Seal）操作。节点无法主动通过源文件生成封存后的文件，而 TEE 对文件的完整性验证是基于封存后的文件，因此基于 TEE 的数据封存可以有效抵御女巫攻击和生成攻击。

F） 节点检索服务的激励

在 BitSwap 的信用机制内，节点的数据交换满足一定博弈机制。目前广泛实践的策略是，每个节点根据其他节点的收发数据情况计算信用分和负债率（debt ratio，r）：

$$\mathrm{r} = \frac{\mathrm{bytes\_sent}}{\mathrm{bytes\_recv} + 1}$$

并计算数据发送概率 P:

$$\mathrm{P}(\mathrm{send}|\mathrm{r}) = 1 - \left(\frac{1}{1 + \exp(6 - 3\mathrm{r})}\right)$$

Crust 网络中节点的信用博弈机制，旨在:

- 使得节点数据交换的整体性能和效率最高；
- 有效防止一些攻击行为和只下载不上传的现象；
- 使得节点能合理的分配带宽资源；

- 配合 Crust 网络的存储订单数据，使得博弈过程更短也更可靠；

配合 Crust 网络存储订单数据，节点可以甄别每一次检索否是针对 Crust 网络中有效文件的检索。基于这个信息，我们可以有效防止节点存储热门文件刷信用分的行为。

Crust 网络存储订单数据还能获得全局的存储信息，也就是“谁”存了“什么”。基于这些信息，Crust 网络可以计算出文件的重复率，对重复率低的文件给予一定的额外存储激励，这样保证了冷门文件存储的可靠性。

一个好的信用博弈机制，可以使网络中提供检索的优质节点可以更块获得存储订单文件，在长期会获得更多订单奖励，进而有效的激励全网节点提供检索并提高服务质量。

G） TEE 的升级

Substrate 框架提供了很强大的无分叉升级机制，但 Crust 的协议栈除了链上协议之外，还包含了部分 TEE 内的协议，因此 Crust 并不能直接的适用这个机制。

Crust 团队根据 Substrate 链和 TEE 的特性，实现了 TEE 的 AB-Upgrade 方案，可以无缝升级的方式安全的更新 TEE 内的代码，实现了链下的无分叉升级。

H） 攻击与威胁

1.SGX 的侧信道攻击

Intel SGX 技术是基于硬件的可信执行环境实现。即使是攻击者获得 OS，hypervisor，BIOS 和 SMM 等权限，也无法直接攻击 Enclave。因此，攻击者往往通过侧信道攻击，比如页表、Cache、DRAM 等攻击面。侧信道攻击主要手段是通过攻击面获取数据，推导获得控制流和数据流信息，最终获取 Enclave 的代码和数据信息，比如加密密钥，隐私数据等等。

在 Crust 的协议框架下，受到侧信道攻击威胁的是节点 TEE 中的核心敏感数据，也就是 TEE 的私钥。一种可行的抵御侧信道攻击方法是在程序的源码层面引入增强的密码学算法，比如使用增强的椭圆曲线和 AES 算法等。基于以上源码层面的增强实现数据流和控制流的隐藏，可以有效地保护节点 TEE 内敏感数据。

2.SGX-ROP 攻击

ROP 全称为 Return Oriented Programming（面向返回的编程）是一种新型的基于代码复用技术的攻击，攻击者从已有的库或可执行文件中提取指令片段，构建恶意代码。通过扫描已有的动态链接库和可执行文件，攻击者提取出可以利用的指令片段(gadget)，这些指令片段均以 ret 指令结尾，即用 ret 指令实现指令片段执行流的衔接。进行 SGX-ROP 攻击，需要将恶意程序加载进入 TEE 下执行，从而对主机造成破坏，并且恶意程序防护软件无法从 SGX Enclave 扫描到有用信息。

由于 Crust 是一个开源框架，社区内发布的任何程序代码和来源均可以被审查，因此从根本上杜绝了恶意代码破坏节点主机的可能性。与此同时，ROP 攻击主要针对本地系统，

如果节点 TEE 内嵌入恶意代码，依然会被网络中的其它节点发现从而无法入网，整个网络不会受到影响。

3.PlunderVolt、VoltPillager 攻击

PlunderVolt 和 VoltPillager 分别从软件和硬件对 SGX 的加密秘钥进行攻击，这种攻击本质上都是通过操控处理器电频电压，对 Intel 高级加密指令注入可控的硬件故障并导致其错误的输出，使得攻击者可以在 enclave 之外恢复加密秘钥。

Crust 的工作量上报机制、文件封装机制以及 MetaData 封装保存都依赖于 ECC、AES 等加密算法。为了避免节点通过 PlunderVolt 或 VoltPillager 攻击获取到私钥从而进行工作量报告伪造，我们将在关键步骤使用重写的加密算法，避免使用 Intel 高级加密指令，从而达到抵御 PlunderVolt、VoltPillager 攻击的效果。

4.最坏情况

Crust 可以抵御目前已知的 SGX 安全漏洞，但对未来潜在的威胁也有一定的防治策略。我们假设最坏情况发生（虽然目前并没有发生的征兆 ），某个恶意节点攻破 SGX 并获取到了节点私钥，这就意味着节点可以随意伪造 TEE 软件和硬件环境证明。

工作量伪造：可能伪造文件完整性校验信息，从而声明一个虚假的存储量以骗取奖励。

假节点入网：可能进行虚假验证，从而导致包含恶意代码的节点加入网络。

5.应对方案

a) 设定一个合理的单点存储上限

   通过限制单节点存储量达到限制虚假算力，从而控制有效的恶意攻击对网络的影响

b) 双 TEE 架构

   此架构下，每个节点需运行两个不同厂商的 TEE。TEE 轮流提供工作量报告并相互校验。任何 TEE 的伪造工作量报告行为都会被另一个 TEE 发现并汇报。因此攻破任何一个单点 TEE 或某个 TEE 厂商的秘钥库泄露都无法对 Crust 网络产生威胁。

c) 基于 RICS-V 的 TEE 解决方案

   目前主流 TEE 解决方案因为不开源而遭受挑战。不开源，意味着可能的漏洞和后门，而 RICS-V 开源指令集架构则可以从根源解决这个问题。随着基于 RICS-V 的 TEE 解决方案的不断成熟，Crust 网络未来也将支持基于 RICS-V 的 TEE 解决方案

V 经济模型

本章将会简要介绍 Crust 中的角色和行为，关于更具体的经济模型细节，请参考《Crust 经济白皮书》。

参考第 III 章 B 节对区块链共识层的定义，Crust 经济生态的参与者包括：验证人、候选人、担保人以及用户。

CRU 是 Crust 网络中流通的功能性代币。在网络中主要有以下功能：

1. Staking 维护 Crust 网络的 GPoS 共识

2. 用于担保所选的节点

3. 作为提供资源服务的保障金和佣金

4. 作为使用网络的交易费

5. 可用于购买资源服务

6. 可用于链上治理机制的竞选和投票，并对提案进行表决

详情请参考《Crust 经济白皮书》

VI 技术演进

Crust 着力于协议的制定和持续完善，并对新技术和新参与者保持开放的态度。

除了前文提及的技术实现，还有一些工作可以为 Crust 带来成长，这些工作包括但不限于：

支持多种 TEE 解决方案。Crust 早期主要基于 Intel SGX 技术，未来 Crust 将会通过 TEE 抽象层接入各种解决方案，比如 ARM 芯片的 TrustZone、AMD 的 SEV、基于 TPM 模块的 Software TEE 以及未来的基于 RICS-V 的 TEE 方案。

支持计算的量化。基于 TEE 的支持，比如基于 TEE 的代码混淆算法，一些类似 FaaS 任务的去中心化执行可以被量化。

支持 Layer2 服务的完善。Crust 除了可以提供基础的去中心化存储激励，还将完善对 Layer2 的支持，使得 Crust 提供的云服务对用户更加友好和便捷。

支持完善的链上治理。为了更好的满足技术和生态的进步，Crust 将会开放公平高效的去中心化的链上治理。

接入 Web3 生态。Crust 可以解决 Web3 生态中的所有去中心化存储场景，同时也将获得 Web3 生态带来的生态加速。

VII 参考文献

[1] Satoshi Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[2] Sabt M, Achemlal M, Bouabdallah A. Trusted Execution Environment: What It is, and What It is Not[C]// IEEE Trustcom/bigdatase/ispa. 2015.

[3] Mckeen F , Alexandrovich I , Anati I , et al. [ACM Press the Hardware and Architectural Support for Security and Privacy 2016 - Seoul, Republic of Korea (2016.06.18-2016.06.18)] Proceedings of the Hardware and Architectural Support for Security and Privacy 2016 on - HASP 2016 - Intel Software Guard Extensions (Intel SGX) Support for Dynamic Memory Management Inside an Enclave[J]. 2016:1-9.

[4] Winter J. Trusted computing building blocks for embedded linux-based ARM trustzone platforms[C]// Acm Workshop on Scalable Trusted Computing. 2008.

[5] Bruschi D , Cavallaro L , Lanzi A , et al. Replay attack in TCG specification and solution[C]// Computer Security Applications Conference. IEEE, 2005.

[6] Douceur J R. The Sybil Attack[C]// International Workshop on Peer-to-peer Systems. 2002.

[7] Dias D, Benet J. Distributed Web Applications with IPFS, Tutorial[C]// International Conference on Web Engineering. 2016.

[8] Cai M, Chervenak A, Frank M. A Peer-to-Peer Replica Location Service Based on a Distributed Hash Table[C]// Supercomputing, Acm/ieee Sc Conference. 2004.

[9] "Filecoin: A Decentralized Storage Network", [online] Available https://filecoin.io/filecoin.pdf

[10] Lerman L, Bontempi G, Markowitch O. Side Channel Attack[J]. Cryptographic Attacks, 2013.

[11] Prandini M, Ramilli M. Return-Oriented Programming[J]. IEEE Security & Privacy, 2012, 10(6):84-87.

[12] Wood Gavin. Polkadot: Vision for a heterogeneous multi-chain framework. 2016.