保护隐私的以太坊智能合约

Second State 和 Oasis Labs

摘要—随着去中心化金融（DeFi）应用飞速发展，剑指变革世界金融基础设施，当今的公共区块链在性能与隐私方面达不到金融服务的要求。这严重阻碍了传统金融机构涉足 DeFi 。 Oasis Ethereum ParaTime 提出了一个新的隐私第一的区块链网络，以执行以太坊智能合约，并交易以太坊数字资产。它让智能合约代码和逻辑做到完全透明。智能合约在一系列去中心化的验证人节点的复制虚拟机上执行，并且通过这些节点的共识，保证结果是正确的。同时，合约数据和状态对于验证人和节点运营者也是保护隐私的的。交易和结果仅对提交交易的用户显示。

1 简介

以太坊[1] 是一组协议，指定了如何在一系列去信任的计算机节点上复制状态（即存储的数据）。节点可以运行用户提交的程序（称为智能合约）来操纵状态。以太坊要求所有节点必须以完全相同的方式操纵状态，并因此维持所有节点之间的复制状态。该协议通过底层区块链保证这一点。这些区块记录了状态，并且每个新区块（即，对状态的更改）必须通过共识，被所有节点接受。

以太坊公共区块链的一个核心特性是其透明性。任何人都可以加入网络并成为节点。这种透明性使任何人都可以下载并验证完整的历史交易记录。但是，它还允许任何可能有恶意的人去分析状态数据和交易，以推断出交易方的身份及相互的关系。在去中心化金融（DeFi）应用中，恶意行为者可以推断出金融智能合约进行的交易，并采取预先行动。

最近发生的一次恶意预先交易攻击[2]案例表明，交易的隐私需要得到保护。当 DeFi 交易全部出于公开状态时，用机器人非法抢跑交易，比起苦苦研究交易策略和算法，更加有利可图。对于小投资者或技术上不成熟的投资者而言，如今的区块链 DeFi 环境险峻。

DeFi 攻击的另一个案例是 SushiSwap 的 Vampire Mining Scheme（吸血鬼挖掘阴谋）[3]。其威胁不止步于抄袭 IP 和软件，甚至还从 Uniswap 交易所“窃取”社区[4]，因为所有 Uniswap IP 地址在流动性池智能合约中都是公开可见的。对于 DeFi 交易所、借贷者和服务提供者而言，还需要保护其智能合约的内部数据以保护用户信息。

本文中，我们介绍了兼容以太坊的区块链系统的设计和实现，该系统可以保护智能合约的隐私，甚至不受节点运营者本身的影响。我们旨在保持智能合约代码和逻辑的透明度以及智能合约执行的完整性。同时，我们旨在隐藏交易内容，并有选择地对合约中的状态变量进行加密。

具体要求包括：

去中心化的公共区块链：我们必须创建一个公共且无许可的网络。 任何人都可以成为节点并处理所有交易。 节点运营者是匿名的，即使在自己的节点上也无法访问原始交易数据或指定的隐私数据。

向后兼容：所有现有的智能合约都必须像在以太坊上一样顺利运行。

细颗粒度保护隐私：对于新创建的 Solidity 合约，开发者可以将合约状态的一部分标记为隐私。 节点运营者执行智能合约，但是永远无法看到合约中的隐私部分的状态。

加密的交易：客户端将加密的以太坊交易发送到节点并接收加密的响应。 该节点以加密消息的形式将交易广播到网络上的其他节点。

我们的解决方案是基于对隐私计算的深入研究而提出的。

2 隐私优先的基础设施

Oasis 协议建立在扎实的理论基础之上，协议还提供了技术基础设施来支持公共区块链网络中的隐私交易。

Oasis 协议 [5] 规定了一个基于 Tendermint 的 PoS （权益证明）区块链[6]主网，该主网充当了网络上交易的唯一真实来源。每个 Oasis 验证人节点都可以选择运行其它 runtime （执行环境）软件，以使用链下（即隐私）数据执行任意计算任务，例如使用私有数据进行机器学习，并在 Oasis 区块链上记录共识结果的哈希值。

节点上的 runtime 软件彼此并行。也称为 ParaTime [5]。每个节点可以选择多个要支持的 ParaTime。对于要在 Oasis 主网上记录的交易，所有运行相同 ParaTime 的节点都必须产生相同的结果。 ParaTime 是一种分片，类似于 Polkadot 的 parachain [7]和以太坊 2.0 的分片（shard）链。

Oasis 协议要求保护隐私的 ParaTime 在可信执行环境 (TEE) 中运行[8]。验证后，每个 TEE 都会从网络中其他已经验证的 TEE 的那里接收秘密信息（比如从 TEE 的密钥管理服务接受系统产生的私钥）。然后，ParaTime 运用该隐私信息在 TEE 内部执行计算。

例如，TEE 从密钥管理服务系统接收其私钥，并从中导出其状态加密密钥。节点使用公共密钥将交易数据直接加密到 ParaTime 中，从而实现了端到端的隐私保护。任何外部人员都可以使用公共密钥加密其数据，并将加密的数据发送到 TEE 软件进行处理。数据在 TEE 内部解密和处理。外部各方看不到数据，只能看到 TEE 软件选择输出的计算结果。

TEE 通常基于硬件。它确保即使节点运营者也无法访问私

2020 Second State，本项目得到了 Oasis 基金会的慷慨资助

有数据，包括即时计算结果。所有主要的 CPU 供应商都已推出自己的 TEE（例如 ARM TrustZone，Intel SGX 和 AMD SEV）。这些基于 CPU 的 TEE 通常称为 enclave。

Oasis 协议支持 TEE runtime 节点之间的机密信息交换。它为 TEE 节点提供了去中心化机制以证明自己并进行注册，并在区块链网络中广播加密交易。运行在所有 Oasis 节点上的 Oasis runtime 提供密钥管理功能，以促进 TEE 节点之间的安全密钥交换。

下一节中，我们将讨论 Oasis Ethereum ParaTime 如何为以太坊智能合约提供强大的机密性和隐私保护。

3 ETHEREUM ON OASIS

保护隐私的 Oasis Ethereum ParaTime [9] 在 Oasis blockchain 网络执行以太坊协议。以太坊交易和状态的隐私性是通过对两者进行加密并仅在 paratime 运行的受信任执行环境（TEE）的范围内进行处理来实现的。向后兼容性通过选择性加密状态变量和保留非隐私交易 header 来保持。

ParaTime 提供了以太坊公链的替代选择，兼具 Oasis 强大的隐私和高度保密性能。 Oasis Ethereum ParaTime 软件包含以下组件。

基于 OpenEthereum [11]的以太坊虚拟机（EVM）[10]。它与当前的以太坊区块链完全兼容。它可以执行所有现有的以太坊智能合约，因此能够管理基于智能合约的数字资产， 例如 ERC-20 token、 ERC-721 token、 DAO、 Uniswap 交易所等。
基于 Second State 虚拟机（SSVM）[13]的以太坊风格的 WebAssembly（Ewasm）虚拟机[12]。 Ewasm 是以太坊 2.0 协议的下一代执行引擎。与 EVM 1.0 相比，它提供了许多性能和功能强化。在 Oasis Ethereum paratime 中，我们在 Ewasm 中实验了新的隐私功能。

用于状态数据的以太坊 host 环境，例如基于以太坊账户的交易和状态语义。

尽管 Oasis Ethereum ParaTime 与当今的以太坊应用完全兼容，但由于其权益证明(PoS）和轻量级共识，它比以太坊主网要快得多。也是由于同样的原因，以太坊 2.0 正在变为采用权益证明。

兼容以太坊还意味着，以太坊数字资产（例如 ETH 和任何 ERC-20 代币）可以以去中心化和未经许可的方式在以太坊和 Oasis 以太坊之间移动。任何人都可以设置智能合约来进行 token 的 atomic swap，无需任何中介。

但最重要的是，Oasis Ethereum ParaTime 允许以太坊交易和智能合约状态对节点所有者和运营者保持机密。它是专为财务隐私和 DeFi 设计的区块链 runtime。

4 隐私交易

Oasis Ethereum ParaTime 在网络传播与执行中都保护以太坊交易。

以太坊用户通常通过 Web3 网关向网络提交交易，该网关暴露了一个 JSON RPC 协议。只能通过 HTTPS / TLS 访问该 RPC 网关，以确保从客户端到节点交易的隐私性。由于 ParaTime 软件（包括 RPC 网关）在 TEE 内部运行，因此可以保证端到端的隐私性。

在此设置中，客户端必须信任 RPC 网关，因为节点上的 HTTPS / TLS Web 服务器软件将能够把交易解密为纯文本。但是，由于任何人都可以在网络上创建 Oasis Ethereum ParaTime 节点，因此始终可以找到或创建受信任的 RPC 网关。

删除受信任的 RPC 的另一种方法是使用客户端加密直接与 Paratime 的 TEE 通信（例如，oasis.js [14]）。但是，由于该解决方案破坏了 Web3 的兼容性，因此不太需要此解决方案，因为这将需要将几乎所有基于以太坊的应用移植到新库中。

RPC 节点收到交易后，便通过 Oasis 协议广播该交易，以确保为每个节点的 TEE 正确加密交易。

每个 Oasis Ethereum ParaTime 节点接收交易，在 TEE 内部对其解密，并在 TEE 中执行。这样可以确保任何人，即便是节点运营者也无法看到交易中的内容。

同时，Oasis 协议可确保根据智能合约对交易进行解密并正确执行。所有 Oasis Ethereum ParaTime 节点必须在执行结果的哈希值上达成共识，以便最终完成交易，确认交易并将其包含在区块链中。

5 隐私合约状态

交易的执行产生一个状态更改，这必须得到记录。例如，它可以更改智能合约中的帐户余额或变量值。以太坊主网将状态记录在每个节点上的未加密数据库中。对于 Oasis Ethereum paratime 我们必须保持状态隐私。

一种简单直接的方法是在将 TEE 内部的所有数据写入数据库之前对其进行加密。这样，只有节点的 TEE 才能从其状态存储读取并对其执行新的交易。

但是，这种方法的缺点是性能不佳。当合约函数是 pure view 时，Web3 RPC 网关应该能够通过简单地从其本地以太坊节点的当前状态 view 读取而在本地执行它。pure view 函数可以在没有支付 gas 的情况下执行，因为它不会改变区块链的状态，因此不需要共识。但是，如果所有状态数据都由 TEE 加密，则即使是只读 pure view 操作也必须通过跨 Oasis 协议的交易来完成，因为根据设计，Web3 网关运营者无法访问 ParaTime 的状态加密密钥。

6 细颗粒度的隐私保护智能合约

更好的设计是智能合约隐私性保持细颗粒度。智能合约开发者只需在其Solidity 合约源代码中添加隐私修饰符。所有标有隐私信息的字段都将保持加密状态，并且不会透露给任何人。在以下示例中，可以通过外部交易设置数据字段，但是永远无法查看其值。

```
1.    confidential int data;
```

```
2.    function f(int in) public {
3.      data = in;
4.    }
5.  }
```

加上交易的隐私性，只能合约的数据字段得到如下保护：

- 调用函数 f 的交易已加密；
- 每个节点在 TEE 内部解密和处理该交易，并将加密的数据字段保存到节点的本地状态存储中，节点运营者无法解密该状态字段；
- 用户没有可调用的合约函数来获取数据的当前值。

Oasis Ethereum ParaTime 将在其 Ewasm 编译器工具链和运行时均支持隐私语言关键词。

- Second State 虚拟机 (SSVM)支持两个附加的字节码指令用于内存操作。加载和存储指令类似于标准 WebAssembly 加载和存储指令[15]。SSVM 使用 TEE 中的专用加密密钥执行加载和存储指令，以确保在将标记为隐私的智能合约数据保存到数据存储或从数据存储加载之前已正确加密和解密。
- Second State EWASM 编译器（SOLL）[16]支持隐私关键字。它针对隐私变量发出用于内存操作的 sload 和 sstore 字节码指令。

有了 Oasis Ethereum ParaTime，我们可以确保智能合约中的隐私数据受到保护，同时可以以低成本且高性能地从节点自由访问公共数据。

7 结论

本文中我们介绍了用于隐私保护以太坊智能合约的 Oasis Ethereum paratime 。通过 token bridges 和 Atmic swap 合约，该新的以太坊网络可以提供一个安全的服务于 Defi 和其它需要保证用户隐私的真实世界以太坊 Dapp 的平台

鸣谢

本项工作得到 Oasis 基金会资助。

参考文献

[1] Buterin, V., “Ethereum Whitepaper” https://ethereum.org/en/whitepaper/ 2020
[2] Robinson, D. and Konstantopoulos, G., “Ethereum is a Dark Forest,” https://medium.com/@danrobinson/ethereum-is-a-dark-forest-ecc5f0505dff 2020
[3] Sushiswap, “The SushiSwap Project,” https://medium.com/sushiswap-org/the-sushiswap-project-8716c429cee1 2020
[4] Adams, H., Zinsmeister, N. and Robinson, D., “Uniswap v2 Core,” https://uniswap.org/whitepaper.pdf 2020
[5] The Oasis protocol project, “The Oasis Blockchain Platform,” https://docsend.com/view/aq86q2pckrut2yvq 2020
[6] Kwon, J., “Tendermint: Consensus without Mining,” https://tendermint.com/static/docs/tendermint.pdf 2014
[7] Wood, G., “Polkadot: Vision for a Heterogeneous Multi-chain Framework,” https://polkadot.network/PolkaDotPaper.pdf 2016
[8] Lee, D. et al., “Keystone: An Open Framework for Architecting Trusted Execution Environments,” https://dl.acm.org/doi/pdf/10.1145/3342195.3387532 2020
[9] Second State, “The Oasis Ethereum ParaTime,” http://www.oasiseth.org/ 2020
[10] Wood, G., “Ethereum: A Secure Decentralised Generalised Transaction Ledger,” https://ethereum.github.io/yellowpaper/paper.pdf 2020
[11] Second State, “The Second State VM,” https://github.com/second-state/SSVM 2020
[12] Oasis Labs, “A web client for the Oasis platform,” https://github.com/oasislabs/oasis.js/ 2020
[13] WebAssembly Community Group, “WebAssembly Instructions,” https://webassembly.github.io/spec/core/text/instructions.html 2017
[14] Second State, “SOLL: a new compiler to generate Ewasm from Solidity or YUL,” https://github.com/second-state/SOLL 2020