IoTeX

Uma Rede Descentralizada para a Internet das Coisas com Base em uma Blockchain Voltada à Privacidade

A Equipe IoTeX (support@iotex.io)

Última Atualização: 12 de Julho de 2018
Versão 1.5

Aviso Legal: Este documento tem a intenção de ser uma visão geral técnica. Não tem a intenção de ser compreensivo ou de já ser o documento final; assim, aspectos não essenciais não são abordados, como API's, ligações e linguagens de programação.

Resumo

A maioria dos dispositivos de Internet das Coisas (IoT) são implantados atualmente de uma maneira centralizada, embora sejam de natureza descentralizada. Muitos problemas apareceram: escalabilidade, alto custo de operação, preocupações quanto à privacidade, riscos de segurança, e falta de valores funcionais. A Blockchain, a descentralização em seu puro projeto, pode ser uma boa solução para estas questões. Primeiramente, a blockchain é elástica o suficiente para solucionar o desafio de escalabilidade da IoT de maneira econômica. Segundo: reter informações em blockchains bem definidas elimina a preocupação desses dados referentes à IoT serem armazenados em nuvem e potencialmente serem vazados ou usados de forma incorreta. Terceiro: a blockchain com contratos inteligentes e tokens tem um enorme potencial para habilitar uma coordenação autônoma de dispositivos para criar valores funcionais. Entretanto, as blockchains atuais possuem suas limitações encarando tais problemas devido às características especiais da IoT, como por exemplo grande quantidade e diversidade de aparelhos, restrições de computação, armazenamento e energia, etc.

Este document apresenta a IoTeX, uma rede descentralizada para a IoT com base em uma blockchain voltada à privacidade com quatro principais inovações:

- '*Blockchains in blockchain*' para uma rede distribuída bem balanceada que maximiza escalabilidade e privacidade de maneira econômica;
- Verdadeira privacidade na blockchain baseada em um confiável Código de pagamentos, assinaturas em anel de tamanho constante sem *trusted setup* além da primeira implementação de *bulletproof*;
- Consenso rápido com finalização instantânea melhorando muito a taxa de transferência da rede e reduzindo custos de transação;
- Arquiteturas de Sistema flexíveis e leves baseadas na IoTeX e criadas especificamente para aplicativos-chave da IoT em diversos segmentos industriais.

Assuntos

Lista de Figuras

Lista de Tabelas

1 Internet das Coisas

A Internet das Coisas (IoT) está surgindo rapidamente como manifestação da visão de sociedade interligada – tudo que se beneficia de suma conexão está conectado. No entanto, essa transformação de longo alcance é apenas o começo. Espera-se que o número de dispositivos de IoT cresça por volta de 21% ao ano, indo para 18 bilhões em 2022 [10] e espera-se também que o mercado global da IoT vá de 170 bilhões de dólares em 2017 para 560 bilhões de dólares até 2022 [15], em uma taxa composta de crescimento anual de 26.9%. Embora muitos especialistas e clientes tenham definido a IoT como a próxima revolução industrial ou a próxima internet, existem três problemas principais que estão atrasando os massivos desenvolvimento e adoção da IoT.

1.1 O Problema da Escalabilidade

A maioria dos dispositivos de IoT são conectados e controlados de uma maneira centralizada atualmente. Estes aparelhos são conectados a infraestruturas *back-end* em serviços públicos de nuvem ou em fazendas de servidores para transmitir informações e receber comandos de controle. Hoje, a capacidade da IoT encara um gargalo por causa da escalabilidade e elasticidade dessas infraestruturas *back-end*, servidores e *data centers*. Não é provável que o substancialmente alto custo de operações para rodar a IoT seja coberto pelo lucro da venda de dispositivos. Como consequência, muitos vendedores não conseguem oferecer aparelhos com preço acessível e aplicações que realmente sejam escaláveis e confiáveis o bastante para situações no mundo real[35].

1.2 A Falta de Privacidade

Espera-se que a IoT abra as portas para a participação em massa de usuários finais em serviços críticos como energia, mobilidade, assuntos legais e estabilidade democrática. Os desafios da privacidade se originam do fato de que a IoT interage com o mundo físico de forma direta e automática e que a quantidade de dados coletados irá aumentar substancialmente assim que ela for mais escalável. Algumas das ameaças comuns à privacidade, como enumeradas em [37], são:

1. Identificação: Associar um identificador (persistente), *ex:* um nome e endereço ou um pseudônimo de qualquer tipo, como um indivíduo;
2. Localização e Rastreamento: Obter a localização de um indivíduo através de diferentes meios;
3. Perfilação: Compilar conjuntos de informações sobre indivíduos para deduzir interesses por associação com outros perfis e fontes de informação;

4. Violações de Privacidade: Transportar informações privadas por meio de um meio público e revela-las a alguém não desejado no meio do processo;

5. Transições no Ciclo de Vida: Os aparelhos geralmente guardam grandes quantidades de dados sobre sua própria história durante todo seu ciclo de vida, o que poderia vazar durante mudanças de controle no ciclo de vida de de um dispositivo;

6. Ataque de Inventário: A obtenção não autorizada de informações sobre a existência e características de coisas pessoais. *Ex:* Criminosos podem usar dados de inventário para verificar uma propriedade com o fim de encontrar a hora mais segura de invadi-la;

7. Ligações: Unir sistemas separados e anteriormente diferentes de forma que a combinação de fontes de informações revele informações (verdadeiras ou falsas) que o sujeito não revelou para as fontes previamente isoladas e, mais importante, também não queria que fossem reveladas.

Todas essas ameaças comuns à privacidade ocorrem devido a vazamento de dados a nível de dispositivos ou durante a comunicação; ou ainda, de maneira mais frequente, vazamentos de dados por partes centralizadas.

1.3 A Falta de Valores Funcionais

A maioria das soluções existentes da IoT sofrem com a falta da criação de valores significativos. "Estar conectado" é a proposição de valor mais usada. Todavia, somente habilitar a conectividade não torna um aparelho inteligente ou útil. Uma porção maior do valor que a IoT produz vem da interação, cooperação e, eventualmente, coordenação autônoma de entidades heterogêneas. Como boas analogias temos as células individuais que ajudam a construir organismos multicelulares, sociedade de insetos, humanos construindo cidades e estados. Por cooperarem, todos estes indivíduos se unem para construírem algo que tem um valor maior do que o deles próprios. Infelizmente, de acordo com [29], 85% dos 'aparelhos legado' não tem a habilidade de interagir ou cooperar uns com os outros devido a questões de compatibilidade. O compartilhamento de dados para aplicações comerciais e operacionais é praticamente impossível.

2 A Blockchain

A tecnologia da Blockchain foi introduzida em2008 e sua primeira implementação, o Bitcoin, foi apresentado um ano depois, em 2009, publicado no documento *Bitcoin: A Peer-to- Peer Electronic Cash System* [21] por Satoshi Nakamoto (pseudônimo). Basicamente, a blockchain é um banco de dados transacional distribuído que é compartilhado por todos os nós que participam na rede. Essa é a inovação mais técnica do Bitcoin e age como um livro contábil público para as transações. Cada nó presente no Sistema tem uma cópia completa do status atual da cadeia, o qual contém

todas as transações já feitas. Cada bloco contém um hash do bloco anterior, conectando-os. Os blocos interligados formam a blockchain.

2.1 Componentes

Uma blockchain pode ser vista como um conjunto quadridimensional com três camadas horizontais, incluindo transações e blocos, consenso e interface de computação, além de uma camada vertical para a governança.

Transação e Blocos

Como a camada horizontal mais abaixo, as transações assinadas estão entre todos os nós e os blocos são gerados por *full nodes*. Essa é a base da blockchain onde a transferência de ativos digitais (bem como os valores inerentes) e segurança das contas são alcançadas através de meios como assinatura de curva elíptica, funções *hash* e Árvore de Merkle.

Consenso

A camada horizontal do meio manifesta a natureza *peer-to-peer* da blockchain, onde todos os nós na rede alcançam consenso em todos os estados internos através de técnicas como o *Proof of Work* (PoW), o *Proof of Stake* (PoS) e suas variantes, *Byzantine- fault tolerance* (BFT) e suas variantes, *etc*. A camada de consenso afeta demais a escalabilidade. O PoW geralmente é considerado menos escalável em comparação ao PoS. Além disso, essa camada impacta profundamente na segurança em termos de *double spending* e outros ataques focados em modificar o status da blockchain de uma forma imprevista.

Interface de Computação

As primeiras duas camadas horizontais ditam a forma da blockchain enquanto a camada da Interface de Computação é crítica para tornar uma blockchain útil, o que engloba extensibilidade e usabilidade. Por exemplo, os contratos inteligentes foram implementados pelo Ethereum para habilitar a programabilidade, onde se pode contar com o "computador mundial" distribuído para executar os termos de um contrato. A Sidechain, junto com a mineração mesclada, também tem sido intensamente desenvolvida para suportar a programabilidade. Protocolos de segunda camada como o *Raiden network* [25], atestam que os canais tem sido desenvolvidos para estender a escalabilidade de uma blockchain nesta camada. Além disso, ferramentas, *SDKs*, *frameworks* e *GUIs* também são muito importantes para a usabilidade. A camada de Interface de Computação dá aos desenvolvedores a capacidade de projetar apps descentralizados (DApps), uma parte essencial em fazer a blockchain útil e gerar valor para a mesma.

Governança

Assim como com os organismos, as blockchains de maior sucesso serão aquelas que melhor se adaptarem aos seus ambientes. Assumindo que esses sistemas precisam evoluir para sobrevirem, o projeto inicial é importante, mas no decorrer de muito tempo os mecanismos para mudança são mais importantes. Essa é camada vertical de governança. Há dois componentes críticos para a governança:

- Incentivo: Cada grupo no sistema possui seus próprios incentivos. Estes incentivos não são 100% alinhados com todos os outros grupos no sistema. Os grupos vão propor mudanças com o tempo que sejam vantajosas para eles. Os organismos são tendenciosos em relação à sua própria sobrevivência. Isso geralmente se manifesta em mudanças na estrutura de recompensas, na política monetária, ou nos equilíbrios de poder.

- Coordenação: Visto que é improvável que todos os grupos tenham 100% de alinhamento nos incentivos por todas as vezes, a habilidade para grupo lidar com seus incentivos em comum é crítica para eles afetarem as mudanças. Se um grupo consegue coordenar melhor do que outro, isso cria desequilíbrio de poder a seu favor. Na prática, um ftor decisive é quanta coordenação pode ser feita *on-chain* (*ex:* votos para as regras do sistema conforme a Tezos [34], ou ainda retroceder na ledger se a maioria dos *stakeholders* não aprova a mudança) *vs*. *off-chain* (como as Propostas de Melhorias para o Bitcoin (BIPs) [3]).

2.2 Modelos Operacionais

As Blockchains podem ser classificadas como 'livres' e 'fechadas', dependendo de como são operadas. Por exemplo, o Bitcoin é uma blockchain 'livre', o que significa que qualquer um pode criar um endereço e começar a interagir com a rede, a qual "constrói confiança a partir de confiança". Em contraste, a blockchain fechada é um ecossistema monitorado onde o acesso de cada participante é definido e diferenciado com base em uma função, que é contrária à anteriormente citada.

Existem pros e contras para cada abordagem. Independente disso, todas essas considerações se resumem a questões de projeto como escalabilidade, computação e complexidade. Por exemplo, o Bitcoin e o Ethereum são blockchains construídas em cima de nós "*trustless*" porque a escalabilidade é muito desejada. Consequentemente, ou muito poder computacional é exigido (no caso do PoW) ou precisa-se de um mecanismo de consenso mais sofisticado. Em contraste, Fabric [14] é uma blockchain 'fechada' onde todos os nós são considerados como confiáveis e possuem identidades criptográficas, ou seja, são emitidos por serviços como *Public Key Infrastructure* (PKI), o que os torna altamente escaláveis com pouco poder computacional e um mecanismo de consenso relativamente direto.

Tabela 1: Benefícios da IoT a partir das Propriedades da Blockchain

Propriedade da Blockchain	Benefícios da IoT
Descentralização	Escalabilidade, Privacidade
Tolerância a Falhas Bizantinas	Disponibilidade, Segurança
Transparência & Imutabilidade	Confiança sólida
Programabilidade	Extensibilidade

3 Benefícios e Desafios da Blockchain e da IoT

Senso/percepção, transformação/transmissão e processamento são a essência das coisas mais inteligentes neste planeta. Para a IoT, enquanto a camada de senso e percepção está espontaneamente distribuída, as duas últimas não estão por enquanto, o que é a raiz da maioria dos problemas de escalabilidade, privacidade e extensibilidade. Nós vislumbramos a tecnologia blockchain, se ela de fato serve como espinha dorsal e sistema nervoso da IoT, como a melhor candidata para solucionar específicos da IoT, que foram acima citados.

3.1 Benefícios

Por abraçar a tecnologia blockchain, a IoT se beneficia imediatamente dos aspectos abaixo graças às propriedades da blockchain, incluindo descentralização, Tolerância a falhas Bizantinas, transparência e imutabilidade. A Tabela 1 resume como a IoT se beneficia das propriedades da Blockchain.

Descentralização

A Descentralização liberta usuários e dispositivos de controle centralizado e monitoramento consistente, lidando assim com parte das questões de privacidade impostas pelas partes centralizadas que monopolizam o mercado e tentam compreender cada aspecto do usuário/dispositivo para seu próprio benefício, *ex:* a propaganda. A Descentralização, sob o contexto da criptoeconomia, também indica ”elasticidade” que geralmente é definida como “o grau em que um sistema é capaz de se adaptar às mudanças na carga de trabalho, provisionando e desprovisionando recursos de maneira autônoma, de modo que, a cada momento, os recursos disponíveis correspondam à demanda atual com a maior precisão possível”. Uma blockchain e uma criptoeconomia subjacente podem ser projetadas de forma que sejam elásticas e econômicas o suficiente para cenários e aplicações na IoT. Por exemplo, um maior número de nós pode ser processado se a rede tiver tarefas de computação e incentivos suficientes para realizar isso.

Tolerância a Falhas Bizantinas (BFT)

O objetivo da tolerância a falhas Bizantinas é evitar falhas nas quais componentes de um Sistema falham de formas arbitrárias, *ex:* não apenas por pararem ou encerrarem mas por processarem solicitações incorretamente, corromperem o seu status local e/ou produzirem saídas inconsistentes ou incorretas. A falas Bizantinas modelam os ambientes no mundo real, nos quais computadores e redes podem se comportar de maneiras inesperadas devido a falhas de hardware, congestionamentos na rede e desconexões, bem como ataques maliciosos. As propriedades da BFT podem ser alavancadas para alcançar muitas propriedades de segurança desejadas no contexto da IoT, *ex:* eliminar ataques de intermediários (MITM) visto que não existe um úncio segmento de comunicação que possa ser interceptado e adulterado, fazendo com que ataques do tipo '*Denial of Service*' (Dos) sejam quase impossíveis.

Transparência e Imutabilidade

A Blockchain fornece garantias criptográficas de que as informações ancoradas na cadeia são sempre transparentes e imutáveis, o que pode ser útil em muitos cenários, *ex:* estados ancorados do mundo IoT na blockchain para propósitos de audição, reconhecimentos de firma, análises forenses, gerenciamento de identidades, autenticações e autorizações.

Programabilidade

O Bitcoin apareceu com uma programabilidade básica para permitir que uma transação fosse bem sucedida somente se o pequeno script *underlying* executasse corretamente. O Ethereum aprimora este recurso para alcançar o contrat inteligente '*Turing-complete*', que é escrito em uma programação de alto nível e executado em uma pequena máquina virtual conhecida como EVM. Essa programabilidade pode e deve ser estendida aos dispositivos da IoT, alguns dos quais possuem atualmente somente uma lógica simples e de difícil programação que não pode ser modificada posteriormente, uma vez que já tenha sido usada.

3.2 Desafios

Se beneficiar de propriedades comuns providas por blockchains não quer dizer que toda blockchain é adequada para uso na IoT. De fato, não parece que nenhuma blockchain pública atual possa ser aplicada à IoT já que existem alguns problemas desafiadores.

Garantia de Privacidade Nativa não é Suficiente

Guarantias de privacidade nativa da blockchain só podem ajudar a abordar o problemático ponto da privacidade na IoT, na medida que elas retém dados sobre a cadeia ao invés de dados dos servidores centralizados, utilizando pseudonimato.

Contudo, se o pseudônimo de um dispositivo estiver vinculado a sua identidade, tudo que ele já fez sob esse pseudônimo estará agora vinculado a ele.

Não Existe Blockchain Que Seja Uma Solução Mágica

Como mencionado acima, a IoT é um universe de sistemas e aparelhos heterogêneos com diferentes propósitos e habilidades. É impossível encontrar uma blockchain que seja a 'solução mágica' que se encaixe na maioria dos cenários. Para ilustrar, uma blockchain para coordenar milhões de nós industriais deve focar em alta escalabilidade e boa taxa de transferência nas transações, enquanto uma blockchain para coordenar dispositivos inteligentes domésticos deve focar em privacidade e extensibilidade. Em um nível maior, os dispositivos IoT como espécie estão definitivamente evoluindo de forma rápida, *ex:* novas tecnologias sendo integradas, novos padrões sendo desenvolvidos, novos aparelhos sendo fabricados com novas funções. Em comtraste, a um nível menor, a capacidade individual, propósitos e ambiente operacional dos dispositivos IoT também ficam mudando com o passar do tempo.

Operações em Cadeia São Pesadas

No mundo da IoT, muitos dispositivos são considerados como nós fracos porque são:

- Incapazes de realizer mineração PoW devido a restrições de poder computacional;
- Incapazes de armazenar grandes quantidades de dados (*ex* nível de *gigabytes*, sem mencionar níveis *terabyte* e *petabyte*) devido a restrições de armazenamento;
- Incapazes de verificar todas as transações por processor toda a Blockchain;
- Incapazes de se conectar aos *peers* portodas as vezes, dependendo do seu tempo de atividade e qualidade da conectividade.

Assim sendo, a maioria das blockchains atuais são muito pesadas/robustas para a IoT.

3.3 Trabalhos Relacionados

A IOTA, lançada recentemente, foi construída ao redor de uma tecnologia não convencional chamada *tangle* [24]. A IOTA tenta dissociar o mecanismo de transição de estado com o mecanismo e canonização de consenso por desprezar conceitos como o de blocos e o de cadeia. Em vez disso, os emissores das transações são também quem as aprovam e a verificação das transações é feita usando um grafo acíclico dirigido (DAG) para tornar as transações rápidas e deixa-las sem custo. A eficiência é obtida por perder status globalmente definidos, o que faz com que recursos desejados como *Simple Payment Verification* (SPV) se tornem bastante desafiadores para clientes e contratos inteligentes.

A IoT Chain (ITC) [16], outro projeto de Blockchain para IoT sediado na China, herda a mesma estrutura *tangle* da IOTA, e tem, portanto, os mesmos prós e contras. A HDAC [13] é também uma outra blockchain para IoT recentemente proposta. O projeto coreano, o qual tem parceria com o grupo Hyundai, irá se concentrar em campos mais específicos da IoT como autenticação de dispositivos e transações *Machine-to-Machine* (M2M).

4 IoTeX: Visão Geral do Projeto e Arquitetura

4.1 Princípios do Projeto

A IoTeX visa se tornar, para a IoT, a espinha dorsal e o sistema nervoso escaláveis e voltados à privacidade. Para alcançar isso e lidar com os desafios previamente mencionados, nosso projeto de arquitetura conta com os seguintes princípios:

Separação de Tarefas

Conectar diretamente todos os nós de IoT em uma única blockchain é um sonho que não pode ser real. Além do fato de que diferentes aplicações de IoT exigem um conjunto de recursos de blockchain fundamentalmente diferente, alocar cada nó de IoT em uma blockchain acelera seu tamanho e dificuldade de processamento, eventualmente tornando-a robusta demais para muitos dispositivos de IoT. Como alternativa, uma separação de tarefas assegura que cada blockchain interajas com um grupo e específico de nós de IoT e, simultaneamente, interaja com outras blockchains quando necessário. Isso é análogo à internet – dispositivos heterogêneos primeiramente formam um grupo intra-conectado, a intranet. Intranets menores podem posteriormente dar origem a uma maior, o que criaria uma conexão à espinha dorsal da internet e as faria comunicar-se entre si. A "Separação de Tarefas" geralmente cria um Sistema bem balanceado para maximizar eficiência e privacidade.

Navalha de Occam

Cada blockchain possui diferentes usos e aplicações e devem ser projetadas e otimizadas em diferentes direções. Por exemplo, uma blockchain que seja dedicada a transmitir transações entre suas *subchains* não precisa ter um contrato *Turing-complete* rodando nela. Uma outra blockchain que conecte dispositivos na mesma zona de confiança não deve se preocupar muito com privacidade nas transações.

Fácil Adequabilidade à IoT

Como já discutido, o mundo da IoT é repleto de sistemas e nós heterogêneos, mais fortes ou mais fracos no que tange a seus recursos de computação, armazenamento e força.

Visto que operações que podem ser realizadas por nós fracos podem ser facilmente feitas pelos fortes, as operações na cadeia devem ser projetadas e otimizadas para nós fracos, ou seja, elas devem ser leves o bastante para conservar recursos como computação, armazenamento e energia.

4.2 Arquitetura: *'Blockchains in Blockchain'*

A IoTeX é uma rede de muitas blockchains que são organizadas hierarquicamente, onde várias blockchains podem rodar ao mesmo tempo de outras, mantendo a interoperabilidade. No universo IoTeX, como mostrado na Figura 1, a *root blockchain* gerencia muitas blockchains independentes, ou *subchains*. Uma *subchain* conecta-se e interage com dispositivos IoT que compartilhem algo em comum, por exemplo, eles têm um propósito funcional semelhante, operam em ambientes parecidos, ou dividem o mesmo nível de confiança. Se uma *subchain* não funciona bem, seja por ser atacada ou por vivenciar erros de software, a *rootchain* não é atingida de forma alguma. Além disso, transações entre blockchains são suportadas para transferir valores e dados de *subchains* para a *rootchain* ou de uma *subchain* para outra através da *rootchain*.

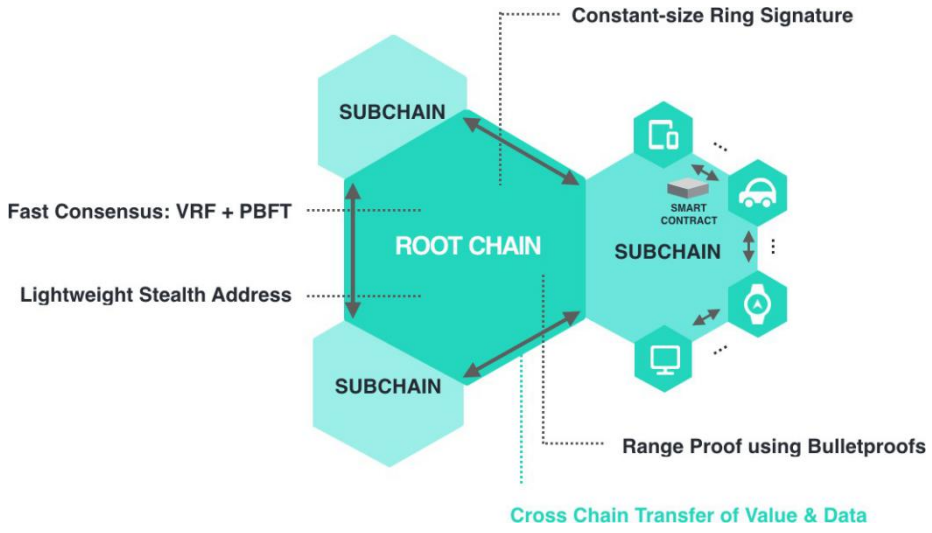

Figura 1: IoTeX: Blockchains in blockchain, arquitetura de uma *rootchain* e *subchains*.

A root blockchain é uma cadeia pública acessível a todos, a qual tem três principais objetivos:

1. **Transmitir** valores e dados pelas *subchains* de uma forma que preserve a privacidade para tornar possível a interoperabilidade entre *subchains*;

Tabela 2: Comparação Entre Rootchain e Subchain

Propriedades	**Rootchain**	**Subchain**
Pública vs Privada	Pública	Pública ou Privada
Escalável	Exigida	Depende
Robusta	Muito Exigida	Exigida
Voltada à Privacidade	Exigida	Depende
Extensibilidade	Sem *Turing Complete*	*Com Turing Complete*
Finalidade Instantânea do Bloco	Exigida	Exigida

2. **Supervisão** das *subchains*, isto é, penalizar os operadores coligados da *subchain* pela interrupção de ligações

3. **Estabelecimento e ancoragem** de pagamentos e confiança para *subchains*.

Com esses objetivos definidos, a *rootchain* foca especificamente na escalabilidade, robustez, funções de preservação da privacidade e habilidade de coordenar *subchains*.

Uma *subchain*, por outro lado, poderia ser potencialmente uma blockchain privada e depender da *rootchain* para interagir com outras *subchains*. Uma *subchain* deseja flexibilidade e extensibilidade para se adaptar a diversas exigências de diferentes aplicações de IoT. É bem provável que uma *subchain* seja rodada por operadores cujos papeis dependam de uma ligação suficientemente alta sendo depositada na *rootchain*. Como opção, o Sistema permite que os operadores nominem um ou mais operadores para agirem por ele, com ou sem ligações extras. O operador age como um cliente leve na *rootchain* e como um *full node* na *subchain*, para selar novos blocos.

Ao todo, as propriedades da *rootchain* e *subchains* estão resumidas na Tabela 2.

4.3 Root Blockchain

A *root blockchain* usa um modelo baseado em UXTO como o Bitcoin [21] e a Monero [20] pelos seguintes motivos:

- A ordenação de transações se torna trivial sem a necessidade de números *nonce* ou sequenciais, o que coloca demandas mínimas em esquemas de consenso e permite que as transações sejam processadas em paralelo;

- Aplicar técnicas existentes de preservação de privacidade tais como a assinatura em anel e ZK- SNARKs para omitir remetente, recebedor e quantidade da transação se torna possível.

A *root blockchain* é composta por blocos ligados por hash, um bloco que é composto por um cabeçalho que se liga ao bloco anterior, por hash, e uma lista de transações.

A *rootchain* permite, a princípio, dois tipos de transações: (1) transações básicas incluindo P2PKH, P2SH, Multisig *etc*., e transações avançadas que habilitam operações entre blockchains como *BondedRegistration*, *Lock*, *ReLock*, *Reorg etc*. As transações validadas são adicionadas a um bloco com tamanho dinâmico, com limite superior à 8MB. Um bloco é produzido a cada três segundos pelo nosso esquema de consenso, como detalhado na próxima seção. A *rootchain* é projetada para não usar *Turing-complete* com o suporte de um script baseado em pilha e um rico conjunto de *opcodes*.

4.4 Subchains

A IoTeX vem com uma estrutura para desenvolver e suprir uma *subchain* personalizada para aplicações descentralizadas de IoT por encapsular primitivas de camada baixa como *gossiping* e mecanismos de consenso. O modelo de permissão, especificação, parâmetros e tipos de transação da *subchain* pode ser customizado para se adequar à aplicação.

As *subchains* da IoTeX usam um modelo baseado em conta, que é melhor para acompanhar mudanças de estado. Há dois tipos de contas, semelhantes ao Ethereum, sendo contas regulares e contratos. Transações válidas são acrescentadas ao bloco, que é produzido pelo mesmo esquema de consenso da *rootchain* no intuito de obter o mesmo grau de finalidade para tornar a comunicação entre blockchains mais eficiente. As *subchains* podem usar o token da *rootchain*, IoTeX, ou definir seu próprio token. O token definido pelos desenvolvedores nas *subchains* podem ser distribuídos publicamente através de *token sales* ou por meio de operações públicas de trade em exchanges.

O contrato inteligente é suportado pelas *subchains* e roda em uma máquina virtual leve e eficiente. Estamos avaliando o *Web Assembly* (WASM) [36], um padrão de web emergente para construção de aplicações web de alta performance. A WASM é rápida, eficiente e pode se tornar determinística e apta ao *sandbox* com algumas modificações, assim como foi tentado pelo projeto EOS [9]. Outras opções também estão sendo exploradas. Com contratos inteligentes, os dispositivos de IoT conectados à mesma *subchain* utilizam o estado compartilhado de duas formas:

- Primeiro, os dispositivos podem interagir com o ambiente físico baseado nos status das suas *subchains*, isto é, as lâmpadas ligam e desligam sozinhas com base em um "estado de relógio" na *subchain*;
- Por outro lado, os dispositivos podem mudar de estado nas *subchains* quando houver alterações no ambiente físico, por exemplo, o termostato atualiza a temperatura por meio de contratos inteligentes baseado nos dados do seu sensor.

4.5 Comunicação Entre Blockchains

Espera-se que a comunicação entre blockchains seja muito usada nas aplicações IoT. Sempre há a necessidade de um dispositivo IoT em uma *subchain* coordenar outro dispositivo em uma *subchain* diferente. Novamente, limitados pelo baixo poder computacional e capacidade de armazenamento dos dispositivos IoT, estamos motivados a projetar uma comunicação entre blockchains de uma forma rápida e econômica.

Pegging e Finalidade de Bloco

Pegging é um mecanismo para escalar a rede do Bitcoin network por meio de sidechains, proposto originalmente em [1]. Ele se baseia muito no método *Simplified Payment Verification* (SPV) [21], e permite que Bitcoins efetivamente "se movam" da blockchain do Bitcoin para a *sidechain* e vice-versa. A ideia por trás disso é simples: Os tokens são enviados a um endereço especial para ficarem trancados na blockchain do Bitcoin; uma vez que essa transação de 'tranca' tenha sido confirmada, é enviada uma transação *Reorg* para a *sidechain*, incluindo a transação de tranca e uma prova de inclusão na forma de um galho Merkle. A *sidechain* usa SPV para verificar a transação *Reorg* e, caso seja validada, ela cria a mesma quantidade de tokens e os envia para o endereço desejado na sidechain. Atualmente, o *pegging* serve como uma primitiva para quase todos os protocolos de comunicação entre blockchains, por exemplo, Cosmos, Lisk e Rootstock. Dois fluxos separados de *pegging* podem ser facilmente alocados juntos para formar a chamada *Two-Way Pegging* (2WP) para fazer transferências de tokens de entrada e saída.

Finalidade de bloco é a garantia que um novo bloco gerado é final e não pode ser modificado. A finalidade de bloco impacta substancialmente a implementação concreta do *pegging* já que é preciso esperar até que a finalidade de bloco seja atingida (pelo menos com alta flexibilidade) na blockchain remetente antes da requisição à *Reorg*. A maioria das blockchains públicas, como a do Bitcoin, não possuem finalidade instantânea. A blockchain destinatária só pode obter uma garantia probabilística, isto é, à medida que mais mineradores PoW confirmarm uma transação, é mais provável que a transação tenha sido aceita. Utilizar um consenso de finalização lida com esse problema porque a cadeia recebedora tem garantia com uma confirmação de bloco na blockchain remetente. Para aplicações IoT, espera-se que as transferências de valor e dados entre blockchains seja rápida e econômica, o que exige um mecanismo de consenso de finalização tanto na *rootchain* como nas *subchains*. O consenso da IoTeX alcança a finalidade instantânea de bloco, detalhada na próxima seção.

Protocolo de Comunicação Entre Blockchains

Vamos analisar em detalhes o protocolo por assumir que um endereço chamado Charlie na *subchain* 1 quer despachar uma transação para um endereço chamado David na *subchain* 2 e que as blockchains usam o mesmo tipo de token sem taxas de transação, para efeito de simplicidade. Note que por aplicar-se ingenuamente o método *pegging*, seriam necessárias quatro transações para

fazer uma "ligação remota" da *subchain* 1 para *subchain* 2 via *rootchain*, ou seja, (1) uma transação de 'tranca' na *subchain* 1; (2) uma transação *Reorg* contra a *rootchain*; (3) outra transação de 'tranca' na *rootchain* e (4) mais uma transação *Reorg* contra a *subchain*.
2. Esse processo indica que David tem que aguardar a aceitação dessa "ligação remota" por pelo menos quatro blocos e os dados que essa "ligação" carrega precisam ser armazenados em todas as três blockchains, o que torna tudo caro e lento. Pretendemos otimizar esse processo por combinar (2) e (3) em uma transação *ReLock*, o que não apenas acelera todo o procedimento como também deixar de alocar dados na subchain 1 e na rootchain. Nosso protocolo está especificado na Figura 2.

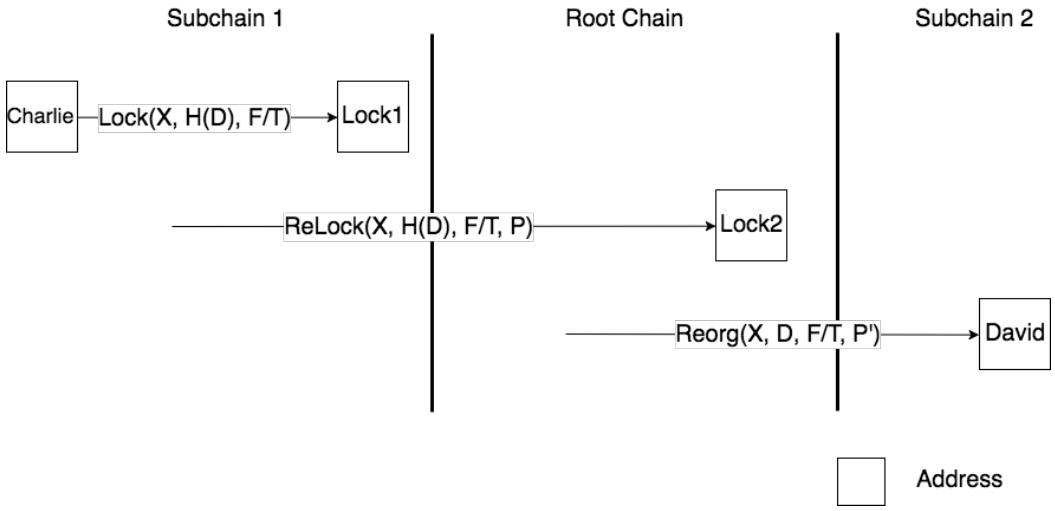

Figura 2: Transações Entre Blockchains

O protocolo *cross-blockchain* da IoTeX tem os seguintes critérios:

1. Cada subchain é registrada na rootchain por submeter uma transação chamada *BondedRegistration* para a *rootchain*, incluindo o nome de sua cadeia, o ID da cadeia, configurações, bloco gênesis e denominação dos operadores. Esse é um processo de execução única;

2. Quando Charlie quiser despachar uma transação para David, ele inicia uma transação de 'tranca'(X, $H(D)$, F/T) onde X é a quantidade de tokens, $H(D)$ é o hash da informação D a ser inserida, F/T indica os endereços de partida e destino, incluindo IDs para ambas as cadeias;

3. Uma vez que a transação de 'tranca' tenha sido incluída na *subchain* 1, Charlie inicia uma transação *ReLock*(X, $H(D)$, F/T , S, P) para a *rootchain* por incluir X, $H(D)$, F/T , alguns status atuais da *subchain* 1, descrita como S, bem como uma prova de inclusão P

que inclui galhos de Merkle de cabeçalhos recentes e outros galhos provando que a transação de 'tranca' foi incluída;

4. A *rootchain* valida a transação *ReLock* e a aceita por incluí-la no bloco mais recente, além de criar X tokens e trancá-los em um endereço especial;

5. Uma vez que a transação *ReLock* tenha sido incluída na *rootchain*, Charlie transmite uma transação $Reorg(X, D, F/T, P^j)$ na rede da *rootchain* com X, D, F/T e mais outra prova de inclusão P^j que prova a inclusão da transação *ReLock*;

6. Os operadores da *subchain* 2 ficam cientes da transação *Reorg* e validam/criam a mesma quantidade de tokens na *subchain* 2. Após isso, os tokens são enviados para o endereço que David associou usando D.

Compartilhando a "Banda" da *Root Blockchain*

Uma possível preocupação com respeito à comunicação entre blockchains é de que uma *subchain* maldosa lance spams na *rootchain* ou em outra *subchain* por enviar uma quantidade enorme de transações entre blockchains, o que deixaria esgotada a capacidade das outras blockchains. Uma forma de mitigar isso é deixar que cada *subchain* ofereça uma cota e limitar "por valor" as transações de uma *subchain* se sua cota estiver esgotada.

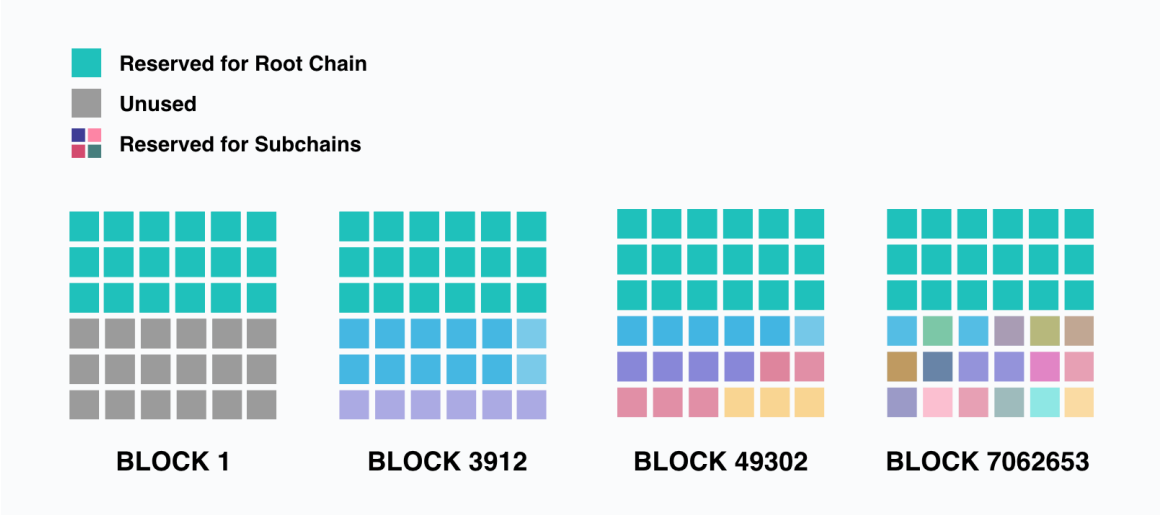

Figura 3: Modelo de Banda para Compartilhamento de Capacidade da *Rootchain*

A cota pode ser definida baseada no espaço de armazenamento dentro de um bloco. Assumindo que o tamanho do bloco seja de no máximo 8MB, que 4MB é destinado para transações normais acontecendo na *rootchain* e que mais 4MB são

Tabela 3: Técnicas de Preservação de Privacidade para Blockchains

Técnica	Omitir Quem Envia	Omitir Quem Recebe	Omitir Quantia
Endereços *Stealth*	N	S	N
Compromisso de Pedersen	N	N	S
Assinaturas em Anel	S	N	N
zk-SNARKs	S	N	S

reservados para todas as transações entre blockchains, o que depois ainda é dividido em, digamos, 4096 pedaços de cota com cada um deles sendo de 1KB. Uma *subchain* oferece *n* pedaços de cota (com um certo limite adicional) de acordo com a utilização pretendida por executar um depósito proporcional a *n*. A cada rodada, *n*KB é o limite que pode ser usado dentro de um novo bloco para transações a partir desta *subchain* e cada uma dessas transações recebe uma cobrança de taxa de "banda" do depósito (para recompensar os mineradores que ajudam a reforçar esta regra); as transações restantes são colocadas em fila e eventualmente caem quando chega a hora. A alocação de cotas pode ser dinâmica no sentido de que ela se altera quando a *rootchain* cresce, conforme mostrado na Figura 3. Se uma *subchain* lança spams em outras, ela desperdiça rapidamente seu depósito e eventualmente até perde a cota. Por outro lado, se uma *subchain* faz um grande depósito para meramente reservar uma grande fatia de banda sem de fato usá-la, a *rootchain* terá um mecanismo para reembolsar parte do depósito baseado no quociente entre o número médio de transações por bloco e a banda reservada, o que ajuda a estabilizar a banda reservada em relação ao seu uso atual.

5 Transação Embutida de Preservação à Privacidade

A privacidade oferecida nativamente por Bitcoin e Ethereum é limitada ao pseudonimato. Os detalhes das transações não são confidenciais. A quantidade da transação e os ativos sendo transferidos, seus metadados e suas relações com outras transações podem ser facilmente descobertas por qualquer um. De fato, há três aspectos na privacidade: privacidade de quem envia, a de quem recebe e a privacidade dos detalhes da transação neste contexto. Diversos esquemas criptográficos podem ser aplicados para lidar com eles, como mostrado na Tabela 3.

A IoTeX integra endreços *stealth* para a privacidade de quem recebe, assinaturas em anel para a privacidade de quem envia e Compromissos de Pedersen para omitir quantidades nas transações com as seguintes inovações e melhorias:

- Uma estrutura leve de endereços *stealth* é projetada para isentar os recebedores de escanear toda a blockchain para saber de transações que estejam chegando;

- A assinatura em anel é otimizada para obter um tamanho mais compacto com um *setup* confiável distribuído.

5.1 Omitindo Quem Recebe Usando Um Código de Pagamento Confiável

Endereço *Stealth*

A técnica do Endereço *Stealth* se originou do protocolo *Cryptonote* [28], o qual soluciona o problema de quem recebe usando o conceito do protocolo de troca de chaves de Diffie–Hellman. Assumindo que Bob quer omitir o fato de que ele recebe tokens de Alice, funciona da seguinte maneira:

1. Bob cria dois pares de chaves privadas e públicas, nomeando como (a, A) e (b, B), onde $A = a \cdot G$ e $B = b \cdot G$, e onde G é o ponto-base em uma curva elíptica.
2. Bob publica as chaves públicas (A, B), as quais são conhecidas como seu endreço *stealth*;
3. Alice faz contas e envia tokens para $P=H(rA)G+B$ usando uma função hash H, um número grande aleatório r e o endereço *stealth* B de Bob. Essa transação é transmitida junto com $R = r \cdot G$;
4. Bob acompanha todas as transações, calcula $P^j = (H(aR) + b)\ G$ (já que ele conhece a, b, R e G) com a esperança que P^j seja igual a P. Se $P^j = P$, Bob poderia gastar os tokens enviados para P^j com a chave privada $H(aR) + b$.

Uma desvantagem óbvia do endereço *stealth* é que o recebedor ou tem de escanear todas as transações na rede (o que não é ideal para o universo IoT) ou depender da assistência de um *trusted full node* (o que compromete a privacidade até um certo ponto).

Código de Pagamento

O Código de Pagamento foi pensado para lidar com a desvantagem dos endereços *stealth* acima citada, com um certo sacrifício na privacidade. A ideia é que Alice notifique Bob de uma forma confidencial sobre um Código de pagamento e Bob observe somente transações contra endereços provenientes daquele código. Assim sendo, esta proposta tem dois fluxos – o de notificação, que é um *setup* único entre duas partes específicas, e o de envio, que pode acontecer por várias vezes entre essas duas partes.

Assumindo que Alice tenha um par master de chaves público-privadas ($mpub_{Alice}$, $mpri_{Alice}$) onde $mpub_{Alice} = mpri_{Alice} \cdot G$ e o par de chaves público-privadas da carteira ($wpub_{Alice}$, $wpri_{Alice}$) onde $wpub_{Alice} = wpri_{Alice} \cdot G$; Bob tanha um par master de chaves público-privadas ($mpub_{Bob}$, $mpri_{Bob}$) onde $mpub_{Bob} = mpri_{Bob} \cdot G$, o fluxo de motificação único funciona como abaixo:

1. Bob deriva $B_0 = b_0 \cdot G = (mpri_{Bob} + Hash(0, seed, metadata)) \cdot G$, a converte para um endereço de notificação addr(B_0), a publica e a escuta.

2. Alice escolhe aleatoriamente um código *cc*; ($mpub_{Alice}$ || *cc*) é o código de pagamento para Alice;

3. Alice calcula um segredo compartilhado $S = wpri_{Alice} \cdot B_0$ e envia o código de pagamento disfarçado $P^j = (mpub_{Alice}||cc) \oplus HMAC512(xofS)$ para addr(B_0);

4. Ao receber, Bob verifica $wpub_{Alice}$, recupera $S = wpub_{Alice} \cdot b_0$ e tira P^j do disfarce para obter ($mpub_{Alice}||cc$).

Uma vez que o fluxo de notificação esteja concluído, Alice e Bob estabelecem um canal privado unidirecional para o envio dos tokens. O primeiro fluxo de envoi funciona da seguinte forma:

1. Alice deriva um novo endereço a partir do seu código de pagamento (que já está compartilhado com Bob) por $A_0 = a_0 \cdot G = mpub_{Alice} + Hash(0, seed, metadata) \cdot G$;

2. Alice escolhe a próxima chave pública não usada e derivada de B_0. Observe que B_0 é a chave pública não usada para a primeira rodada.

3. Alice calcula tanto o novo segredo compartilhado $S_0 = a_0 \cdot B_0$, como a chave pública passageira usada para enviar a transação, a qual é $B_0^j = B_0 + \text{SHA256}(S_0) \cdot G$

4. Bob poderia derivar A_0 de forma não interativa já que ele conhece o código de pagamento de Alice e apenas presta atenção à endereços provenientes de $B_0^j = B_0 + \text{SHA256}(S_0) \cdot G$ and $S_0 = A_0 \cdot b_0$.

5. Ao receber, Bob poderia usar os tokens com a chave privada $b_0 + \text{SHA256}(S_0)$.

O fluxo de envio a seguir funciona de maneira semelhante.

Bob não precisa escanear ou depender de um *full node* para escanear todas as transações. A transação de notificação deixa clara a intenção que Alice tem de enviar algo à Bob, porém o atual "envio de algo" está omitido de todos os outros.

Código de Pagamento Confiável

Para minimizar ainda mais os riscos de privacidade, projetamos o código de pagamento confiável em cima da proposta original de código de pagamento. Enquanto o fluxo de envio permanece o mesmo, nós melhoramos o fluxo de notificação para tornar possível que a Alice compartilhe secretamente seu código de pagamento com Charlie sem usar a transação de notificação, assumindo que Alice e Bob possuam um canal privado unidirecional e que Bob e Charlie tenham outro canal privado unidirecional. Para alcançar isso, nós aprimoramos *Hashed Timelock Contracts* (HTLCs), o que exige que o recebedor de um pagamento ou reconheça o recebimento antes de um prazo

por gerar uma prova de pagamento criptográfica ou perca a chance de reivindicar o pagamento, retornando-o a quem o enviou.

Assumindo que Charlie tenha um par master de chaves público-privadas ($mpub_{Charlie}$, $mpri_{Charlie}$) onde $mpub_{Charlie} = mpri_{Charlie} \cdot G$: O fluxo de notificação otimizado funciona conforme abaixo.

1. Charlie deriva $C_0 = c_0 \cdot G = (mpri_{Charlie} + Hash(0, seed, metadata)) \cdot G$, o converte para um endereço de notificação addr(C_0), e o publica. Observe que C_0 é publicado para que Alice calculi o segredo compartilhado, poré não para receber quaisquer transações;

2. Alice gera seu Código de pagamento ($mpub_{Alice}||cc$) da mesma forma;

3. Alice calcula um segredo compartilhado $S = wpri_{Alice}\ C_0$ e envia um Código de pagamento disfarçado $||\ P^j = (mpub_{Alice}\ cc)\ HM\ AC512(xof\ S)$ com X tokens como incentivo e $HT\ LC(Hash^2(cc))$ para Bob, usando o canal privado unidirecional deles, onde $HT\ LC$, como parte do script de tranca ou recuperação, afirmam que os tokens podem ser gastos se a pré-imagem de $Hash^2(v)$ for dada, ou seja, $Hash(cc)$;

4. Bob, incentivado pelos tokens enviados por Alice, manda P^j, Y, $Y < X$ tokens e $HTLC(Hash^2(v))$ para Charlie, usando o canal privado unidirecional deles;

5. Charlie, ao receber a transação de Bob, calcula $S = wpub_{Alice}\ c_0$ para tirar do disfarce o Código de pagamento de Alice e a transação por revelar o $Hash(cc)$, o que faz com que a transação "Alice-Bob" se torne apta para ser gasta, com o intuito de recompensar Bob.

Uma vez que este fluxo estiver concluído, Alice e Charlie estabelecem um canal privado unidirecional para enviar tokens. É digno de nota que o roteamento da transação de Alice pode ter várias etapas.

Nossos códigos de pagamento confiáveis oferecem a melhor privacidade em termos de omissão na intenção de "enviar algo" na cadeia por meio da otimização de canais privados existentes sem a adição de poder de computação e sobrecarga no armazenamento aos nós, o que, enquanto projetados para cenários da IoT, são utilizáveis na maioria das blockchains como o Bitcoin.

5.2 Habilitando Transações Confidenciais

5.2.1 Descrição do Problema

Uma transação típica na blockchain do Bitcoin está retratada na Figura 4. Essencialmente, uma transação na blockchain é só uma tupla ($\{pk_{in,i}\}$, $\{pk_{out,j}\}$, $\{v_{i,j}\}$), onde $\{pk_{in,i}\}$ são endereços de entrada, $pk_{out,j}$ são endereços de saída e $v_{i,j}$ são quantidades de transação entre endereços de entrada e saída. Devido às transações de Bitcoin serem armazenadas em texto limpo na *ledger* pública, elas levantaram muitas preocupações referentes à segurança e privacidade.

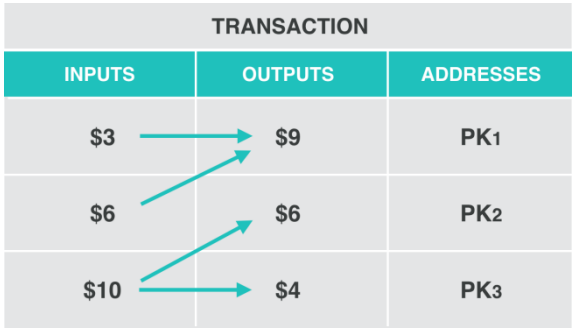

Figura 4: Uma Transação na Blockchain do Bitcoin

O alvo das transações confidenciais (Figura 5) é deixar que *apenas* remetentes e recebedores das transações revelem os $\{v_{i,j}$ (valores) e esconda-os de todo o mundo. Além disso, as transações confidenciais também permitem que outras entidades da rede verifiquem a validade dessas transações em questão sem verem as verdadeiras quantidades. A realização de transações confidenciais na blockchain exige um número de técnicas criptográficas avançadas.

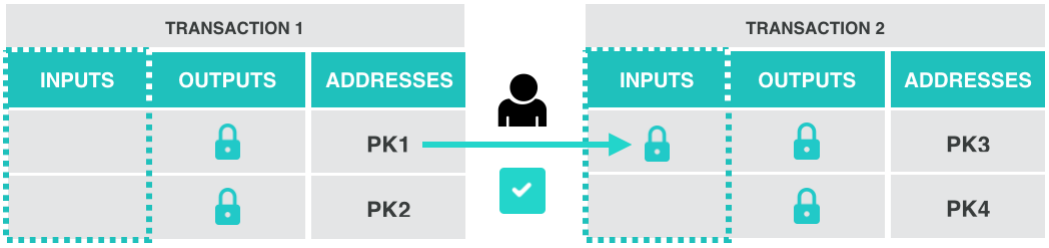

Figura 5: Uma Transação Confidencial com Verificabilidade Pública

5.2.2 Blocos de Construção Criptográficos

Prova de Conhecimento

Uma prova de conhecimento, descrita por (P, V), é uma prova interativa entre um provador P e um verificador V, na qual o provador quer demonstrar que ele sabe de alguma informação. Mais especificamente, P tem (x, w) pertencentes a uma relação R, onde x é o problema e w é a solução (também chamada de *testemunha*). V conhece x e ele aceita somente se P puder convencer V que ele conhece w.

Prova de Conhecimento-Zero

Em um Protocolo de Conhecimento-Zero, o provador prova uma declaração ao verificador sem revelar nada sobre essa declaração a não ser que ela é verdadeira, o que protege o provador de um verificador malicioso, que poderia tentar obter mais conhecimento do que o necessário. O protocolo pode ser *interativo* ou *não interativo*. A diferença chave das provas não interativas é que todas as interações consistem de uma única mensagem enviada pelo provador ao verificador. Usamos a notação $\mathsf{NIZKPoK}(\alpha, \beta) : a = g^{\alpha}\ b = g^{\beta}$ para demonstrar uma prova não interativa e de conhecimento-zero dos valores α e β tais que $a = g^{\alpha}$ e $b = g^{\beta}$. Todos os alores não colocados entre parênteses são presumidos de serem conhecidos ao verificador. Quando usamos uma prova de conhecimento-zero não interativa para autenticar dados auxiliares, o esquema resultante é descrito como uma *assinatura de conhecimento* [8]. Basicamente, um esquema de assinatura de conhecimento significa que alguém em posse de uma solução w para o problema x assinou a mensagem m. Para a expressão $\mathsf{NIZKPoK}$, usamos a notação $\mathsf{SoK}[m](\alpha, \beta) : a = g^{\alpha} \wedge b = g^{\beta}$ para denotar uma assinatura de conhecimento na mensagem m.

Assinatura em Anel

O conceito de assinatura em anel foi introduzido inicialmente por Rivest *et al*.[27] em 2001 como um tipo especial de assinatura em grupo. Em uma assinatura em anel, o signatário da mensagem escolhe um conjunto de membros, incluindo eles mesmos, como os possíveis assinantes da mensagem. O verificador pode ser convencido de que a assinatura foi de fato gerada por um dos membros. Entretanto, o verificador não é capaz de dizer qual membro de fato gerou a assinatura. Ao contrário de uma assinatura em grupo genérica, uma estrutura de assinatura em anel não envolve designar um gerente de grupo para administrar o conjunto de membros, eliminando assim a possibilidade de revelar a identidade do verdadeiro signatário da mensagem, através do gerente de grupo. Com o intuito de oferecer anonimato em transações de tokens via contratos inteligentes, um tipo especial de assinatura em anel, chamado de assinatura em anel linkável, tem sido empregada na Monero, uma criptomoeda voltada à privacidade [20]. Assinaturas em anel linkáveis possuem a seguinte propriedade adicional: quaisquer assinaturas geradas pelo mesmo signatário, seja assinando a mesma mensagem ou mensagens discrepantes, possuem um identificador (como uma etiqueta) linkando as assinaturas. Essa propriedade permite que terceiros verifiquem eficientemente que as assinaturas foram geradas pelo mesmo signatário, sem o vazamento da identidade do verdadeiro signatário. A assinatura em anel linkável usada na Monero é a chamada Assinatura em Grupo Anônima, Espontânea e Linkável Multi-Camadas (MLSAG) [22], a qual é uma assinatura em anel em um conjunto de vetores-chave e tem uma complexidade de comunicação de $O(m(n + 1))$, onde m é o número de pares de chave público-privadas que estão sob posse do signatário e n é o tamanho do anel.

Acumuladores

Os acumuladores, que foram originalmente propostos por Benaloh e de Mare em [2], são definidos como funções hash com a propriedade de serem *quase-comutativas*. Uma função quase-comutativa $f : X \times Y \to X$ satisfaz isso, para todo $x \in X$ e em $y_1, y_2 \in Y$, temos $f(f(x, y_1), y_2) = f(f(x, y_2), y_1)$. Um acumulador permite que combinemos um conjunto de valores em uma compilação segura e essa compilação não depende da ordem na qual os valores são acumulados. Também podem ser usados para gerar uma testemunha que habilite alguém a atestar que um dado valor é verdadeiramente parte do acumulador.

Estrutura de Compromisso

Uma estrutura de compromisso é um protocolo que permite que um usuário se comprometa com um valor de sua preferência sem revela-lo ao recebedor do processo. Em uma fase posterior, quando o usuário é questionado a revelar o valor em questão, o recebedor terá os meios para verificar que seu valor revelado é de fato e incondicionalmente linkado ao seu compromisso. Uma estrutura de compromisso deve atender dois requisitos. Enquanto a solicitação de *omissão* evita que o recebedor saiba o conteúdo do compromisso, a solicitação de *espera* evita que o usuário trapaceie ao abrir seu compromisso. Na estrutura de compromisso de Pedersen [23], os parâmetros do domínio são um grupo cíclico $\mathbb{G}$ de primeira ordem q e geradores $(g_0, \ldots, g_m)$. Para se comprometer aos valores $(v_1, \ldots, v_m) \in \mathbb{Z}_q^m$, é preciso escolher um número aleatório $r \in \mathbb{Z}_q$ e o compromisso

$$C = \mathrm{PedCom}(v_1, \ldots, v_m; r) = g_0^r \prod_{i=1}^{m} g_i^{v_i}$$

5.2.3 Nossas Melhorias

Em [31], Sun *et al*. apresentaram o RingCT 2.0, que empregou um acumulador criptográfico para reduzir ainda mais a complexidade de comunicação para $O(n)$ às custas de computações adicionais. Observamos que, embora o RingCT 2.0 reduziu significativamente a complexidade de comunicação quando comparado à MLSAG, a geração de parâmetros de domínio do acumulador exige um processo único de "*trusted setup*" assim como a Zcash. Portanto, é preciso confiar que quem gerou os parâmetros secretos irá destruí-los quando eles terminarem, o que levantou preocupações de segurança e privacidade para o sistema. Par lidar com essa questão, nossa solução é empregar um protocolo de computação segura de múltiplas partes (SMPC) em meio a um conjunto de nós *bootstrapping* da blockchain para gerar parâmetros de domínio secretos de uma forma segura e distribuída. Além disso, as direções a seguir estão sendo atualmente investigadas para otimizar os protocolos semelhantes ao RingCT em termos de sobrecargas computacionais e de comunicação:

- Uma nova estrutura de assinatura em anel linkável com complexidade de comunicação menor que $O(n)$
- Uma nova abordagem para agregar múltiplas assinaturas em anel linkáveis
- Um protocolo sigma para setups não confiáveis de parâmetros de domínio secretos

Pretendemos propor uma nova solução para transações confidenciais que seja capaz de alcançar um bom equilíbrio entre comunicação e custos computacionais.

5.3 Provando a Faixa de Quantidade da Transação com *Bulletproofs*

Como uma substituição inesperada aos compromissos de Pedersen, os *bulletproofs* [5], um novo protocolo de prova de conhecimento-zero não interativo com bem poucas provas e sem um *setup* confiável, foi proposto recentemente e reduz o tamanho de uma faixa de prova de linear para sublinear e diminui ainda mais o tamanho da transação sem sobrecargas de computação adicionais. Uma vez que os *Bulletproofs* se encaixam bem no projeto da IoTeX, nós vamos integra-los à IoTeX.

6 Consenso Rápido com Finalidade Instantânea

6.1 Background

6.1.1 Proof of Work

O *Proof of Work* (PoW) é a espinha dorsal para alcançar consenso global na maioria das blockchains, incluindo as do Bitcoin e Ethereum. O PoW faz com que a construção de um bloco válido e a anexação dele à blockchain seja computacionalmente difícil. Quanto maior a blockchain se torna, mais difícil fica para reverter quaisquer transações anteriormente gravadas pela blockchain. Para manipular a blockchain, um invasor precisa obter 51% de todo o poder de computação de uma rede de blockchain baseada em PoW.

Embora o PoW ofereça uma solução elegante para o consenso global de grandes blockchains distribuídas, ele tem várias desvantagens inerentes. Os custos de computação gerais para manter o consenso global são os mesmos de um ataque de 51%. Isso significa que mesmo que a maioria dos participantes na blockchain sejam honestos, eles ainda têm de usar muita eletricidade para manter a blockchain, o que não é adequado para ambientes de redes IoT que geralmente prezam por eficiência energética. Além disso, a nível de dispositivos individuais, a computação PoW normalmente exige muitos ciclos de CPU e utilização de memória, o que impõe solicitações complicadas à fabricação do hardware e custos de dispositivos de IoT incorporados. Por último mas não menos importante, o PoW não provê finalidade instantânea, o que é uma propriedade crítica exigida na construção de uma comunicação entre cadeias eficiente.

6.1.2 *Proof of Stake*

O *Proof of Stake* (PoS) foi proposto como uma alternativa eficiente ao PoW para as blockchains alcançarem consenso. Ele visa evitar os problemas do PoW que foram mencionados acima. A ideia básica do PoS é que um conjunto de nós aleatoriamente escolhido vote no próximo bloco e seus votos recebem peso com base no tamanho de seus depósitos (ou *stake*). Se certos nós não se comportarem apropriadamente, podem perder seus depósitos. Dessa forma, sem o computacionalmente intenso PoW, a blockchain pode rodar de maneira muito mais eficiente e pode atingir uma estabilidade econômica: Quanto mais *stake* um participante tiver, mais incentivos o nó tem para manter o consenso global, e menos provável que o nó se comporte de forma errada. Existem alguns projetos e implementações públicas do PoS, como o Tendermint [32] que tem sido adotado por muitas aplicações [33].

6.1.3 *Delegated Proof of Stake* (DPoS)

O *Delegated Proof of Stake* (DPoS) melhora a ideia do PoS no sentido que o DPoS permite que os participantes escolham alguns delegados para representar suas porções de *stake* na rede. Por exemplo, Alice pode mandar uma mensagem à rede para dar a Bob a capacidade de representar o *stake* dela e votar no nome dela. O DPoS fornece vários benefícios para nossas aplicações IoT:

- Pequenos *players* podem colocar seus *stakes* em pools para terem uma maior chance juntos de participarem na proposta e votação de blocos e dividirem a recompensa depois.
- Nós restritos por recursos podem escolher seus delegados, portanto nem todos os nós precisam ficar *online* para contribuir com o consenso.
- Os delegados podem ser os nós com grande poder energético e condições de rede, além de podem ser escolhidos dinâmica e aleatoriamente, então teremos uma maior disponibilidade geral para a rede alcançar consenso.

Dentre as típicas criptomoedas usando o DPoS citamos a EOS [9] e a Lisk [18].

6.1.4 Tolerância Prática a Falhas Bizantinas

A Tolerância Prática a Falhas Bizantinas (PBFT) foi proposta por Castro e Liskov [7] em 1999 como um algoritmo eficiente e resistente a ataques para chegar a acordos em uma rede assíncrona distribuída. Planejamos usar a PBFT para o algoritmo de votação de nosso mecanismo de consenso DPoS porque é um algoritmo conciso e bem

estudado que oferece finalidade rápida, o que é muito importante na construção de uma blockchain eficiente e escalável. Como demonstrado no documento original de Castro e Liskov, a PBFT oferece disponibilidade e segurança se no máximo um terço dos nós na rede forem defeituosos ou mal-intencionados, além do que o custo de rede da PBFT é irrisório, cerca de 3% em comparação com o sistema de rede não replicado.

Algumas criptomoedas baseadas na PBFT são a Stellar [30] e a Zilliqa [38].

6.2 *Randomized Delegated Proof of Stake* (Roll-DPOS)

Para ter um mecanismo de consenso rápido e eficiente com finalidade de bloco instantânea no contexto da IoT, combinamos os conceitos do DPoS, PBFT e *Funções Aleatórias Verificáveis* (VRFs). A VRF por introduzida oficialmente por Micali *et al.* em [19] e é uma família de funções que pode produzir publicamente provas verificáveis para a exatidão de suas saídas aleatórias. Em alto nível, nossa proposta de Roll-DPOS tem quatro fases: *eleição de candidatos*, *formação de comitê*, *proposta de bloco* e *finalização do bloco*.

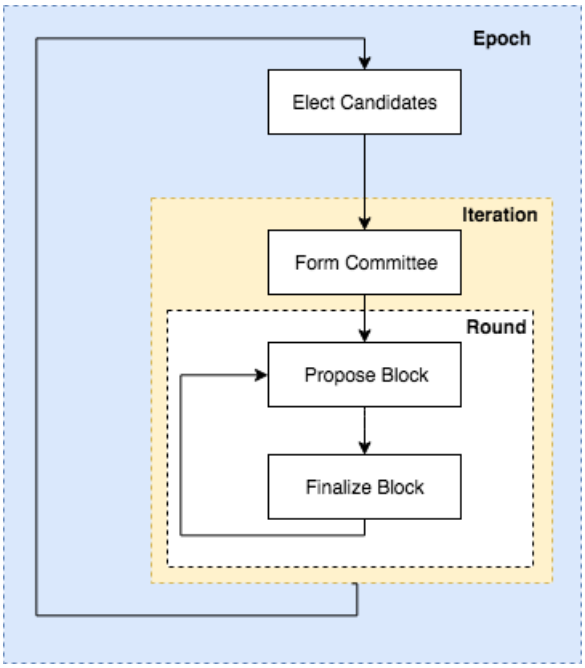

Figura 6: *Randomized Delegated Proof of Stake* (Roll-DPOS)

6.2.1 Eleição de Candidatos

Todos os nós da rede IoTeX poderiam participar nessa fase em termos de votação para os candidatos ao comitê. Para incentivar os nós a votarem, o sistema se certifica de

que os delegados dividam recompensas com seus eleitores. Os candidatos formam um grupo de pelo menos 97 delegados; esse número irá crescer no futuro para evitar ao máximo a centralização do poder de mineração. Uma vez que os candidatos são selecionados, eles serão fixados em uma época, a qual consiste em 47 iterações.

6.2.2 Formação de Comitê

Em cada iteração, um comitê aleatório de tamanho 11 é escolhido da *pool* de candidatos usando VRF para criar blocos nas próximas 11 rodadas. A ideia é utilizar a *hash* do bloco na última iteração e a chave privada do nó como entradas para a VRF produzir uma saída booleana indicando se um deles for selecionado para membro do comitê, uma prioridade indicando um pedido para proposta de bloco e uma prova indicando certa qualificação para propor o bloco em uma rodada específica. O uso da VRF é importante visto que ela fornece uma maneira não-interativa de classificar todos os delegados por proporem blocos de uma maneira justa e segura. Para este fim, nós usamos a eficiente VRF como é usada no Algorand[12].

6.2.3 Proposta de Bloco

A cada rodada (aproximadamente a cada 3 segundos), cada nó de comitê propõe um novo bloco e o transmite para a rede, junto com a prioridade e a prova. Apenas o bloco proposto por um nó de comitê com a mais alta prioridade e que não tenha sido proposto na mesma iteração é considerado pelos outros nós, o qual é chamado de bloco candidato.

6.2.4 Finalização do Bloco

Na mesma rodada, todos os outros nós usam PBFT para votar a favor/contra o bloco candidato. Se mais de dois terços dos nós de comitê concordarem sobre a validade do bloco candidato, ele é finalizado e anexado à blockchain por todos na rede. Depois disso, *proposta de bloco* e *finalização de bloco* são executadas na próxima rodada; se a iteração atual terminar, outro comitê aleatório será formado antes que as funções *proposta de bloco* e *finalização de bloco* sejam executadas.

6.3 Criando Pontos de Verificação Periódicos para Clientes Leves

Nas redes IoT, esperamos que muitos dispositivos sejam *clientes leves*, que são os participantes na blockchain que não registram localmente o histórico completo de transações. Considerando a sobrecarga de armazenamento da blockchain inteira, ex, mais de 100GB no caso do Bitcoin [4], muitos dispositivos de IoT embutidos e de baixo custo podem não ter capacidade para baixar a blockchain inteira. Todavia, esses clientes leves ainda têm como verificar rapidamente a exatidão da blockchain

E interagir com ela – o projeto está incluído no *whitepaper* original do Bitcoin, de Satoshi [21].

Entretanto, usar o PoS ao invés do PoW tem uma desvantagem para clientes leves. Ao verificar a exatidão de blockchains baseadas no PoS, os clientes precisam baixar uma lista de chaves públicas e assinaturas para proponentes de bloco e eleitores e os grupos de proponentes de bloco e eleitores podem mudar para cada bloco. Assim, quando clientes leves voltam a ficar *online* depois de um tempo *offline*, eles têm que baixar um grande número de chaves públicas e assinaturas, além de verificar todas elas. Para mitigar essa questão de desempenho, Vitalik, o inventor do Ethereum, propôs a criação de pontos de verificação periódicos na blockchain, chamados *épocas* [6], a cada, por exemplo, 50 blocos. Cada *checkpoint* pode ser verificado conforme o *checkpoint* anterior, de forma que os clientes leves possam acompanhar muito mais rapidamente a blockchain toda.

7 O Token na Rede IoTeX

O token digital native e criptograficamente seguro da rede IoTeX (IOTX) é um grande componente do ecossistema na rede IoTeX e ele é projetado para ser usado apenas na rede. Antes do lançamento da mainnet da IoTeX, o token existirá como um token compatível ERC20 na blockchain do Ethereum, o qual será migrado para um token na mainnet da IoTeX quando esta for lançada.

A IOTX é exigida como um 'cripto-combustível' virtual para o uso de certas funções designadas na rede IoTeX (como executar transações e rodar as aplicações distribuídas na rede IoTeX), provendo os incentivos econômicos que serão consumidos para encorajar participantes a contribuírem e manterem o ecossistema na rede IoTeX. Recursos computacionais são requeridos para rodar diversas aplicações e executar transações na rede IoTeX, bem como para validar e verificar informações/blocos adicionais na blockchain, portanto os provedores desses serviços/recursos exigiriam incentivos econômicos para a provisão desses recursos (isto é "mineração" na rede IoTeX) para manter a integridade da rede e a IOTX será usada como unidade de troca para quantificar e arcar com os custos dos recursos computacionais consumidos. A IOTX será minerável por 50 anos, com as recompensas de mineração reduzindo ao passar do tempo, com base em um modelo de redução por gradiente linear.

A IOTX é uma parte integral e indispensável da rede IoTeX, porque na ausência da IOTX, não haveria uma unidade de troca comum para pagar por esses custos, o que tornaria insustentável o ecossistema na rede IoTeX.

A IOTX é um token de utilidade functional não reembolsável que será usado como unidade de troca entre participantes na rede IoTeX. O objetivo de introduzir a IOTX é fornecer um método de pagamento seguro e conveniente entre os participantes que interajam dentro do ecossistema da rede IoTeX.

A IOTX não representa de forma alguma participação acionária, direitos, títulos ou interesse na *IoTeX Foundation Ltd*. (a **Fundação**), suas afiliadas, ou qualquer outra companhia, empresa ou empreendimento, nem dará aos *holders* quaisquer promessas de taxas, receitas, lucros ou retornos de investimento, e não se destina a constituir títulos em Cingapura ou em qualquer outra jurisdição relevante. A IOTX só pode ser utilizada na rede IoTeX e a posse de IOTX não possui outro direito, expresso ou implícito, além do direito de usar a IOTX como um meio para permitir o uso e a interação com a rede IoTeX.

Em particular, a IOTX:

(a) não é reembolsável e não pode ser trocada por dinheiro vivo (ou pelo seu valor equivalente em qualquer outra moeda digital) ou por qualquer obrigação de pagamento pela Fundação ou outra afiliada;

(b) não representa ou confere ao *holder* qualquer direito de nenhum tipo com respeito à Fundação (ou qualquer uma de suas afiliadas) ou suas receitas e ativos, incluindo, sem limitações, qualquer direito de receber receitas futuras, partes em ações, direitos de *stake*, compartilhamentos, quaisquer votações, distribuições, resgates, liquidações, direitos de propriedade (incluindo todas as formas de propriedade intelectual) ou outros direitos legais e financeiros, direitos equivalentes, ou qualquer outra forma de participação relacionada à Rede IoTeX, Fundação, Distribuidora e/ou seus provedores de serviço;

(c) Não se destina a ser uma representação de dinheiro (incluindo dinheiro eletrônico), *security*, *commodity*, título, instrumento de dívida ou qualquer outro tipo de instrumento financeiro ou investimento;

(d) Não é um empréstimo à Fundação ou qualquer uma de suas afiliadas e não tem intenção de representar uma dívida da Fundação ou de qualquer uma de suas afiliadas, não havendo expectativa de lucro; e

(e) não fornece ao *holder* qualquer propriedade ou outro interesse na Fundação ou em qualquer de suas afiliadas.

8 Ecossistemas Geridos pela IoTeX

A blockchain da IoTeX suporta uma variedade de exossistemas IoT: economias compartilhadas, casas inteligentes, veículos autônomos, cadeia de suprimentos, etc. Diferentes tipos de desenvolvedores alavancam a IoTeX de maneiras distintas. Dentre os desenvolvedores apoiados pela IoTeX estão fabricantes de hardware para IoT, desenvolvedores de sistemas de controle para dispositivos IoT, desenvolvedores de *apps* para casas inteligentes, fabricantes de dispositivos para economias compartilhadas, fornecedores de *crowdsourcing*, desenvolvedores de carros autônomos, etc. Essa seção descreve alguns ecossistemas geridos pela IoTeX.

8.1 Economias Compartilhadas

Nos últimos anos, muitas empresas têm focado nas economias compartilhadas, desde a área de transportes como Uber/Lyft/Didi, a área de hospedagem como a Airbnb, e de bicicletas também, como a Mobike/ofo, até compartilhamento a nível de itens pequenos como bancos de bateria/carga, guarda-chuvas, etc. Todas elas dão às pessoas um melhor modo de vida, embora algumas delas estão sofrendo com seus modelos de negócios. Esse é um outro tópico para discutir, seus modelos de negócios; nosso principal foco aqui são suas arquiteturas tecnológicas. Entre todas as economias compartilhadas, o segmento de transportes é um que não consegue se livrar da operação humana, vide os motoristas. Não é uma economia gerida pela IoT. Contudo, no futuro, quando a tecnologia de veículos autônomos amadurecer e se tornar popular, a área de transportes será gerida pela IoT.

Todas as economias compartilhadas geridas pela IoT dividem algumas similaridades: Todas elas exigem uma trava que pode ser aberta por um depósito e uma taxa de aluguel. É bem possível e também eficiente fortalecer todo o processo de compartilhamento e retorno usando um dispositivo IoT. No mundo centralizado, as economias são geridas por uma nuvem centralizada. Há várias desvantagens:

1. Um grande deposito é feito por uma empresa que pode não ser confiável. Recentemente, temos visto muitos casos como o da empresa que conduz um serviço de bicicletas compartilhadas na China que não é capaz de pagar depósitos de volta para seus usuários;

2. As economias compartilhadas não são totalmente guiadas pela comunidade. Muitas coisas compartilhadas são posse de empresas. Isso tem causado um desperdício de recursos na sociedade. Considere as bicicletas compartilhas como exemplo: Quando as empresas que fornecem o serviço estiverem fora de operação, as bicicletas serão descartadas.

3. Devido à natureza centralizada, os dados do usuário serão armazenados e controlados por uma empresa. Existem riscos onde tanto a nuvem como o cliente podem ser hackeados com o intuito de se obter esses dados.

A IoTeX, como uma infraestrutura, poderia ser utilizada para gerenciar essas aplicações sem os problemas acima e gerar economias compartilhadas descentralizadas e mais eficientes. De maneira concreta, uma economia compartilhada gerida pela IoTeX oferece os seguintes benefícios:

1. O depósito é determinado totalmente pelo contrato inteligente. Sem ninguém segurando o dinheiro, o retorno do depósito é sempre garantido. Os usuários não precisam confiar na empresa para utilizar o serviço.

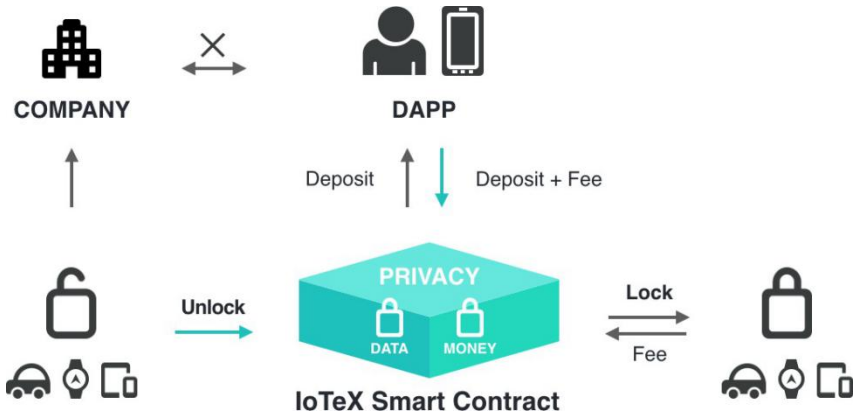

Figura 7: Economia Compartilhada Gerida pela IoTeX

2. Cada coisa compartilhada nota seu valor e missão de uma forma autônoma. No ecossistema, não importa quem possui as coisas compartilhadas nele. Todos podem ser donos de algo e contribuir para o ecossistema. A economia pode ser conduzida pela comunidade. Como resultado, empresas podem desempenhar a função de manter a trava na IoT e gerenciar a comunidade. É um modelo de negócios muito mais enxuto, onde as empresas podem se expandir rapidamente e atender mais pessoas.

3. Novamente, os usuários não precisam confiar na empresa para manter suas informações. Seus dados estão guardados na cadeia com proteção de privacidade.

A Figura 7 descreve como uma economia compartilhada funciona com base na blockchain da IoTeX.

8.2 Casas Inteligentes

No Mercado atual de casas inteligentes, muitos fabricantes de dispositivos IoT ainda estão usando tecnologias obsoletas para desenvolverem seus produtos. Eles precisam de uma grande quantidade em trabalho de desenvolvimento na sua nuvem. Os custos para desenvolvimento e manutenção são altos, e o desempenho é baixo por causa das idas e voltas necessárias até a nuvem. Implantar seus produtos na blockchain da IoTeX irá reduzir bastante os custos operacionais de engenharia e computação em nuvem, e, ao mesmo tempo, aumentar bastante o desempenho dos seus dispositivos. No exemplo de uma simples lâmpada inteligente, com tecnologia em nuvem, precisa-se

De duas viagens a partir da instrução do usuário par a alteração do estado de uma lâmpada. Os fabricantes não são especialistas em nuvem então geralmente o serviço deles não chega a ser brilhante. O processo e ida e volta pode levar de um a três segundos. Isso os força a usar o serviço de nuvem de grandes companhias de TI. Existem alguns pontos negativos em usar esses serviços de nuvem:

1. Os fabricantes não conseguem controlar integralmente a disponibilidade dos serviços em nuvem.

2. Eles precisam pagar continuamente pelo serviço de nuvem apesar de cobrarem uma única vez ao venderem seus dispositivos IoT.

3. Há riscos de suas nuvens, pelo lado do cliente, e/ou para a intranet serem hackeadas, fazendo com que dados de usuários sejam roubados, desencadeando problemas de segurança.

Em contraste, a blockchain da IoTeX gerencia os dispositivos localmente e, quando necessário, interage com uma cadeia pública na internet. A cadeia pública é mantida pela comnidade. Não há custos de manutenção para os fabricantes de IoT. A blockchain da IoTeX conta com proteção de privacidade, o que pode evitar que vazem informações ou que o controle seja hackeado mesmo se a intranet não for segura.

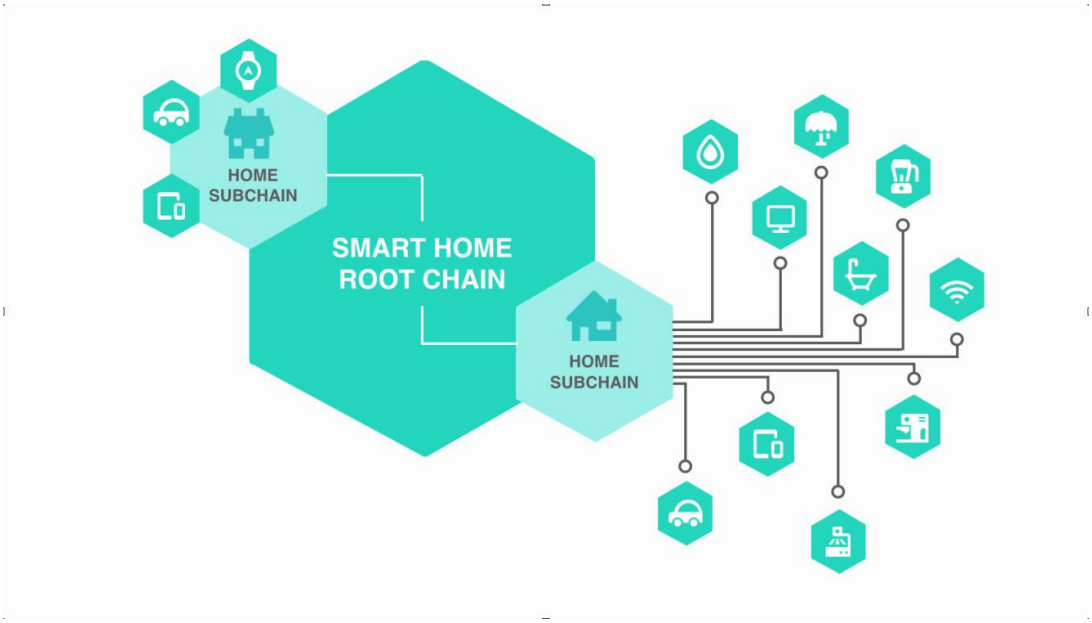

Figure 8: Casa Inteligente Gerida pela IoTeX

Além de permitir que fabricantes de IoT implementem seus dispositivos na blockchain da IoTeX, a IoTeX firmará parcerias com fabricantes de chips IoTpara desenvolver chips habilitados para a Blockchain da IoTeX, visando acelerar os ciclos de projeto e fabricação de dispositivos IoT. Os fabricantes de IoT irão simplesmente integrar o chip para obter suporte da blockchain da IoTeX para os seus dispositivos.

8.3 Gerenciamento de Identidade

O crescente mundo da IoT tem impactado a forma que o Gerenciamento de Identidade e Acesso (IAM) precisa funcionar. Em termos de identidade das coisas, o IAM deve ser capaz de gerenciar sistemas dos tipos usuário-dispositivo, dispositivo-dispositivo e/ou dispositivo-serviço. Uma saída direta para o gerenciamento de identidade é considerar a blockchain da IoTeX como um sistema PKI descentralizado (graças a sua imutabilidade) onde cada entidade recebe uma identidade criptográfica na forma de certificado TLS e o correspondente privado. Esse certificado, que tende a ter uma vida útil curta, é assinado pelo certificado embutido de vida longa do dispositivo e publicado na blockchain da IoTeX (tanto na *rootchain* como na *subchain*). *Peers* e outras entidades podem acessar e confiar no certificado de vida útil curta que está ancorado na blockchain e as coisas poderão ser autenticadas ao ficarem *online*, garantindo uma comunicação segura entre outros dispositivos, serviços e usuários, além de provar a integridade deles.

Além disso, os certificados embutidos de vida útil longa para dispositivos poderiam ser organizados hierarquicamente, como o PKI convencional, onde 'dispositivos-pai' assinam 'certificados filhos'. Com a hierarquia, a revogação e rotação de certificados se torna possível. Por exemplo, se um dispositivo é comprometido, seu dispositivo-pai ou até seu dispositivo-avô poderiam assinar um comando de revogação e envia-lo à blockchain, onde o último invalida o certificado do dispositivo.

9 Trabalhos Futuros de Pesquisa

A seguir estão listadas algumas direções de pesquisa futuras para otimização da IoTeX.

Computação de Preservação à Privacidade

Existem diversas áreas nesta direção, as quais estamos ativamente explorando:

- Como reter status confidenciais na blockchain que podem ser usados para computação por um grupo específico de nós;
- Contratos inteligentes de preservação à privacidade onde eles podem ser avaliados quando suas lógicas estiverem protegidas por criptografia. Enquanto a criptografia totalmente homofóbica [26] e esquemas de ofuscação indistinguíveis [11] são o Santo Graal na teoria, propostas práticas como *Hawk* [17] prometem para o futuro próximo;
- Reduzir ainda mais a necessidade de computação e armazenamento das técnicas de preservação à privacidade que a IoTeX está usando atualmente;
- Versões imunes à tecnologia quântica das técnicas de preservação à privacidade que a IoTeX está atualmente utilizando, como assinatura em anel resistente à tecnologia quântica.

Redução e Transferência dos Status

Estamos avaliando diferentes formas de diminuir seguramente os status armazenados nas *subchains* para reduzir a necessidade de armazenamento visto que muitos dispositivos IoT têm essa capacidade limitada. A compressão de blocos e transações são definitivamente de fácil obtenção. Sem contar que transferir status da *subchain* para a *rootchain* (uma vez que a última é mais forte em termos de armazenamento) de uma forma eficiente e que preserve a privacidade também é um assunto interessante para se aprofundar.

Governança e Auto-modificações

Enquanto a blockchain da IoTeX oferece incentivos para a manutenção do consenso em suas *ledgers*, ela não possui por enquanto um mecanismo *on-chain* que altere perfeitamente as regras que governam o desenvolvimento de seu protocolo e do protocolo de recompensas. Planejamos conduzir uma pesquisa sobre governança e auto-modificações par lidar com isso.

Blockchains Estruturadas em Árvore

A IoTeX hoje é uma blockchain de dupla camada e, naturalmente, deveria ser estendida a uma árvore de blockchains por alavancar técnicas como *Plasma* e *Cosmos*. O plano consiste em avaliar essas propostas e melhorar o projeto atual da IoTeX para eventualmente dar suporte a estruturas hierárquicas mais complexas.

10 Conclusão

Nesse documento, introduzimos a IoTeX, uma blockchain extensível, privada e escalável dedicada para a Internet das Coisas, com sua arquitetura e principais tecnologias englobando:
1. 'blockchains in blockchain' para maximizar escalabilidade e privacidade, 2. Verdadeira privacidade na blockchain com base em um Código de pagamento confiável, assinaturas e manel de tamanho constante sem um *trusted setup* e implementação pioneira de *bulletproofs*, 3. Consenso rápido com finalidade instantânea baseada em VRF e PoS para altas taxas de transferência e finalidade instantânea, além de 4. arquiteturas de sistema flexíveis e leves baseadas na IoTeX.

11 Agradecimentos

Gostaríamos de expressar nossa gratidão a nossos mentores e conselheiros e as várias pessoas nas comunidades de criptomoedas, criptografia e IoT, pelos seus feedbacks iniciais e sugestões construtivas.

Referências

[1] Adam Back et al. “Enabling blockchain innovations with pegged sidechains”. In: *URL: http://www. opensciencereview. com/papers/123/enablingblockchain-innovations-with-pegged-sidechains* (2014).

[2] Josh Benaloh and Michael de Mare. “One-Way Accumulators: A Decentralized Alternative to Digital Signatures”. In: *Advances in Cryptology — EUROCRYPT ’93: Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23–27, 1993 Proceedings*. Ed. by Tor Helleseth. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 274–285. isbn: 978-3-540-48285-7. doi: 10.1007/3-540-48285-7_24. url: https://doi.org/10.1007/3-540-48285-7_24.

[3] *Bitcoin Improvement Proposals*. https://github.com/bitcoin/bips.

[4] Blockchain Size. https://blockchain.info/charts/blocks-size. [5]Benedikt Bünz et al. *Bulletproofs: Efficient Range Proofs for Confidential Transactions*. Cryptology ePrint Archive, Report 2017/1066. https://eprint.iacr.org/2017/1066. 2017.

[6] Vitalik Buterin. *Light Clients and Proof of Stake*. https://blog.ethereum.org/2015/01/10/light-clients-proof-stake/.

[7] Miguel Castro, Barbara Liskov, et al. “Practical Byzantine fault tolerance”. In: *OSDI*. Vol. 99. 1999, pp. 173–186.

[8] Melissa Chase and Anna Lysyanskaya. “On Signatures of Knowledge”. In: *Advances in Cryptology - CRYPTO 2006: 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006. Proceedings*. Ed. by Cynthia Dwork. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 78–96. isbn: 978-3-540-37433-6. doi: 10.1007/11818175_5. url: https://doi.org/10.1007/11818175_5.

[9] *EOS*. https://eos.io/.

[10] AB Ericsson. “Ericsson mobility report: On the pulse of the Networked Society”. In: *Ericsson, Sweden, Tech. Rep. EAB-14* 61078 (2015).

[11] Sanjam Garg et al. “Candidate indistinguishability obfuscation and functional encryption for all circuits”. In: *SIAM Journal on Computing* 45.3 (2016), pp. 882–929.

[12] Yossi Gilad et al. “Algorand: Scaling byzantine agreements for cryptocurrencies”. In: *Proceedings of the 26th Symposium on Operating Systems Principles*. ACM. 2017, pp. 51–68.

[13] *HDAC Blockchain for IoT*. https://hdac.io/.

[14] *Hyperledger Fabric*. https://www.ibm.com/blockchain/hyperledger.html.

[15] Internet of Things (IoT) Market by Software Solution (Real-Time Streaming Analytics, Security Solution, Data Management, Remote Monitoring, and Network Bandwidth Management), Service, Platform, Application Area, and Region - Global Forecast to 2022. https://www.jasper.com/sites/default/files/cisco-jasper-hidden-costs-of-delivering-iiot-services-en_2.pdf. 2016.

[16] *ITC Blockchain for IoT*. https://iotchain.io/.

[17] Ahmed Kosba et al. "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts". In: *Security and Privacy (SP), 2016 IEEE Symposium on*. IEEE. 2016, pp. 839–858.

[18] *Lisk*. https://lisk.io/.

[19] Silvio Micali, Michael Rabin, and Salil Vadhan. "Verifiable random functions". In: *Foundations of Computer Science, 1999. 40th Annual Symposium on*. IEEE. 1999, pp. 120–130.

[20] *Monero – Private Digital Currency*. https://getmonero.org/.

[21] Satoshi Nakamoto. *Bitcoin: A peer-to-peer electronic cash system*. 2008.

[22] Shen Noether and Adam Mackenzie. "Ring Confidential Transactions". In: *Ledger* Vol. 1 (2016), pp. 1–18. doi: https://doi.org /10 .5195/ledger.2016.34.

[23] Torben Pryds Pedersen. "Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing". In: *Advances in Cryptology — CRYPTO '91: Proceedings*. Ed. by Joan Feigenbaum. Berlin, Heidelberg: Springer Berlin Heidelberg, 1992, pp. 129–140. isbn: 978-3-540-46766-3. doi: 10.1007/3-540-46766-1_9. url: https://doi.org/10.1007/3-540-46766-1_9.

[24] Serguei Popov. "The tangle". In: *IOTA* (2016).

[25] *Raiden Network*. https://raiden.network/.

[26] Ronald L Rivest, Len Adleman, and Michael L Dertouzos. "On data banks and privacy homomorphisms". In: *Foundations of secure computation* 4.11 (1978), pp. 169–180.

[27] Ronald Rivest, Adi Shamir, and Yael Tauman. "How to leak a secret". In: *Advances in Cryptology—ASIACRYPT 2001* (2001), pp. 552–565.

[28] Nicolas van Saberhagen. *Cryptonote v 2. 0*. 2013.

[29]Samsung. *Samsung ARTIK and Successful Strategies for Industrial IoT Deployment*. Samsung, 2016.

[30] *Stellar*. https://www.stellar.org/.

[31] Shi-Feng Sun et al. "RingCT 2.0: A Compact Accumulator-Based (Linkable Ring Signature) Protocol for Blockchain Cryptocurrency Monero". In: *Computer Security – ESORICS 2017: 22nd European Symposium on Research in Computer Security, Oslo, Norway, September 11-15, 2017, Proceedings, Part II*. Ed. by Simon N. Foley, Dieter Gollmann, and Einar Snekkenes. Cham: Springer International Publishing, 2017, pp. 456–474. isbn: 978-3-319-66399-9. doi: 10.1007/978-3-319-66399-9_25. url: https://doi.org/10.1007/978-3-319-66399-9_25.

[32] *Tendermint*. https://tendermint.com/.

[33] *Tendermint Ecosystem*. https://tendermint.readthedocs.io/en/master/ecosystem.html.

[34] *Tezos: A new digital commonwealth.* https://www.tezos.com/.

[35] *The hidden costs of delivering IIoT services*. https://www.jasper.com/sites/default/files/cisco-jasper-hidden-costs-of-delivering-iiot-services-en_2.pdf. 2017.

[36] *WebAssembly*. http://webassembly.org/.

[37]Jan Henrik Ziegeldorf, Oscar Garcia Morchon, and Klaus Wehrle. "Privacy in the Internet of Things: threats and challenges". In: *Security and Communication Networks* 7.12 (2014), pp. 2728–2742.

[38] *Zilliqa*. https://www.zilliqa.com/.