White Paper Técnico

(Alguns detalhes podem ser adicionados)

Sistema financeiro descentralizado

CREDITS

Versão 1.5/12.09.2017

Conteúdo

Abstrato

A plataforma CREDITS é um sistema financeiro descentralizado para a interação direta entre participantes nos princípios de peertopeer (P2P). A plataforma expande o potencial de usar serviços financeiros com base num livro de contas distribuído, excluindo contratos inteligentes e CREDITS cryptocurrency. O sistema visa unir todos os participantes num site, fornecendo-lhes uma plataforma para criar e usar serviços financeiros, onde todos podem oferecer um serviço e usá-lo. Graças a um sistema tecnológico equilibrado, a plataforma CREDITS oferece uma nova solução técnica e um novo modelo conceitual de interação dos participantes em rede para o desenvolvimento de serviços financeiros descentralizados modernos.

Introdução

Um acordo totalmente compatível com o sistemas de entrega de serviços que permite a formação de serviços financeiros: transferências de dinheiro, troca de moeda e valor, crédito, financiamento e outros serviços diretamente entre os participantes. Tudo é fornecido sem intermediários adicionais, de acordo com um princípio - um dos participantes iguais - para outros participantes do sistema. Como resultado, todos recebem serviços mais baratos, mais rápidos e melhores.

O mundo está a mover se em direção à interação direta entre as pessoas nos princípios do peertopeer - igual a igual. Uma revolução já aconteceu! Isso é claramente visto pela reviravolta nos meios de comunicação de massa: até a década de 1990, jornais, revistas e TV eram os principais provedores de informações. Hoje, os líderes de opinião são blogueiros, encontrados em canais do Youtube e redes sociais, o dinheiro é investido em crowdfunding e ICO, e as informações são armazenadas em sistemas de nuvem descentralizados.

O setor financeiro, talvez, é uma das poucas indústrias que ficam para trás, que resiste à introdução da descentralização e à interação direta entre os participantes. Embora, tecnicamente, seja muito mais fácil criar serviços financeiros descentralizados do que criar veículos não tripulados.

Um ambiente tecnológico correspondente é necessário para criar um sistema de produtos e serviços financeiros descentralizados com base em lshger distribuído:

1. Alta velocidade de execução (em segundos), juntamente com a capacidade de lidar com uma grande quantidade de transações simultaneamente (centenas de milhares por segundo) a baixo custo de cada transação (para micro pagamentos e transações não corretas).

2. Desenvolvimento de um sistema onde todos os participantes e itens necessários para os serviços descentralizados financeiros qualitativos são combinados: personalização de usuários, KYC, agência de histórico de créditos, centros de liquidação de dinheiro fiduciário, retirada e encaminhamento de criptografia e assim por diante

Estas são duas tarefas básicas e básicas que atualmente dificultam o desenvolvimento de produtos financeiros peekspeer.

Apresentamos uma solução para essas tarefas, implementamos com a ajuda do sistema financeiro CREDITS.

A plataforma descentralizada tecnológica única de CREDITS pode combinar todos os participantes de serviços financeiros, executar de forma segura e rápida todas as transações usando os princípios de um livro contábil distribuído. Selfexecuting contratos inteligentes e os princípios de um sistema de votação federativo oferecem oportunidades ilimitadas a todos os participantes para criar interações únicas de vários produtos financeiros. A plataforma abre um novo mercado enorme e um novo potencial para o uso de projetos e serviços de blockchain em setores financeiros e outros que não poderiam ser usados anteriormente devido às limitações de custo e velocidade de transação.

1. Roteiro de rede

Definições

1. Um sistema é um conjunto de nodes de rede descentralizados que executam o processamento, salvando transações, executando e confirmando os termos de contratos inteligentes, processando pedidos de sistemas de terceiros, fornecendo dados de informações quando solicitado.
2. Um node de rede é um computador em que um cliente de rede completo está instalado, conectado a um sistema comum, verificando transações e executando-os no ledger.
3. Um ledger é a lista de transações confirmadas pelo sistema e armazenadas em todos os nodes da rede.
4. Uma transação é o item do sistema, denotando uma solicitação para executar um método de contrato inteligente ou qualquer ação na rede e registrando os resultados num sistema de blockchain.
5. Um contrato inteligente é o item do sistema, protocolos de computador que facilitam, verificam ou asseguram a conformidade com os termos de interação. Eles geralmente têm uma interface de usuário e muitas vezes imitam a lógica das relações contratuais. A principal propriedade de um contrato inteligente é a descentralização e a independência de uma fonte central.
6. Um método de contrato inteligente é o código do programa responsável por calcular o resultado do trabalho dos termos do contrato inteligente e gravá-lo na rede.
7. Uma parte contratante é o participante da rede final e o usuário do sistema.

Rede de Nodes

Utilizamos vários tipos de nodes, dependendo da sua finalidade de construir uma rede descentralizada baseada em acesso livre e conexão de node:

1. Um node comum (ОУ) é o node que participa na verificação de transações para validade, mas tem um fator mínimo de confiança. Também é um candidato para o papel de um node confiável e o node do processamento atual no próximo ciclo de seleção de função de node no trabalho.

2. Um node confiável (ДУ) é o node que participa na verificação de transações e possui o fator de confiança máximo (1), é um candidato para a função do node do processamento atual e do node comum. Este node não pode ser confiado durante um número calculado matematicamente de ciclos de seleção e votação entre nodes. O cálculo matemático depende do número de nodes e da complexidade da rede.

3. O node principal (ГУ) da rede é o node participante na verificação e responsável pela adição de transações ao bloco do razão maior da transação. Este node não pode tornar-se confiável ou o node do processamento atual durante um número de ciclos de votação calculado matematicamente, cujo cálculo matemático depende do número de nodes e da complexidade da rede.

O sistema usa um fator de confiança - um valor numérico fracionário absoluto de 0 a 1, expresso em termos matemáticos do número de nodes confiáveis +1 para o número total de nodes na rede. O número máximo de nodes confiáveis não pode exceder 50% dos nodes de trabalho.

O Ultimo Bloco Guardado

O livro comum de blocos (CRB) é o estado sincronizado de todo o livro comum de blocos em todos os nodes do sistema.

Pelo conteúdo do bloco do razão, significamos uma unidade de informações armazenadas que contém um código de hash do bloco anterior e uma lista de dados relacionados a esse livro com o número associado do bloco anterior. Após o recebimento do bloco de outro node, ele ocupa o lugar no maior livro comum de blocos de acordo com o número. Isso economiza a largura de banda da rede.

Durante a sincronização, apenas o número de bloco é verificado primeiro. Se o bloco estiver ausente neste node, ele é baixado e salvo.

Como resultado, o sistema, a qualquer momento, contém a última cópia atualizada do livro. Nomeamos o último livro maior (LR). Ele é criado automaticamente pelo node responsável pela formação do razão em chegar a um consenso. Este bloco é enviado a todos os nodes do sistema para manter a uniformidade atualizada do estado do registro em todos os nodes do sistema.

Cada node está associado a todos os outros nodes na rede e troca constantemente novos blocos com transações com eles, de modo a manter sempre a informação relevante. Todos os blocos formam um conjunto de transações que esperam ser adicionadas ao razão. Ao mesmo tempo, cada servidor gera conjuntos assumidos de candidatos para outros servidores e o conjunto proposto de transações. Uma decisão é feita após a verificação, seja para adicioná-los ao livro-razão.

Como resultado, é possível armazenar vários dados do razão em vários servidores - os nodes do sistema e todas as informações estão protegidas. Quanto mais nodos no sistema, mais confiável e independente é.

Sincronização de nodes

Cada novo node é iniciado e sincronizado após a definição da determinação e verificação de confiança completa. Para melhorar a taxa de processamento de informações, todos os processos são tratados simultaneamente, independentemente uns dos outros. Se não houver variáveis de entrada, uma loja de livros de contas vazia é criada - um espaço é reservado na RAM para acesso simplificado. No caso de o ledger necessário não estar disponível, uma solicitação é enviada para nodes confiáveis para receber todas as transações feitas para a conta sincronizada.

Se o parâmetro de entrada for um objeto que caracterize a transação, a busca em todos os segmentos de sincronização em execução será iniciada. A operação resulta num código numérico - o número da posição no ledger do node confiável para o segmento atual ou o número do erro se o valor for menor que zero. Se o método thread terminar com um erro de conexão, o t hread termina completamente.

2. Consenso de rede

O consenso em CREDITS é um método de decisão de grupo. Com o objetivo de desenvolver soluções finais aceitáveis para todos os nodes de rede.

Comparação de Consenso

A definição dos princípios do livro CREDITS descentralizado para comparar diferentes tipos de consenso:

- disponibilidade da rede (os nodes podem gravar dados na rede e lê-los a qualquer momento);
- Modificabilidade por todos os nodes de rede participantes;
- Consistência de todos os nodes do sistema (todos os nodes vêem uma versão absolutamente idêntica da rede, que é atualizada após as mudanças);
- Resistência à separação (se um node se tornar inoperável, isso não afeta a operação de todo o livro).

Parâmetro comparado	Créditos específicos PoW e PoC	PoW	PoS
O princípio de identificar o node que gerou o bloco.	Cálculo da função matemática. Confirmação do armazenamento da última cópia do razão.	Execução de um cálculo iterativo da função matemática, com complexidade variável.	Procure a pilha máxima entre os participantes (nodes concorrentes).
Ataque 51%.	Improvável, uma vez que é necessário ter um livro-razão completo em recursos e um poder computacional para calcular, e os nodes confiáveis são selecionados dinamicamente.	Provavelmente, mas será muito caro em termos de uso de recursos.	Provavelmente, mas caro, devido à necessidade de aumentar a própria pilha.
Compensação pelo trabalho realizado no site para adicionar ao ledger / blockchain.	Calculado automaticamente, depende da comissão por operação.	Fornecido correção para o bloqueio de mineração.	Fornecido correção para o bloqueio de mineração.

O Conceito do node da Rede Principal

Todos os nodes de rede são descentralizados e nenhum deles tem prioridade. É necessário definir um node de rede que processe a fila de transações armazenadas em diferentes nodes de rede. Depois disso, ele deve inserir um bloco de transação recém-gerado no lgerador.

A plataforma CREDITS usa seu próprio protocolo combinado para aumentar a velocidade do processamento de transações, para fornecer segurança total de armazenamento, processamento e transferência de transações de dados. O protocolo baseia-se no cálculo da função matemática de todas as transações contábeis, aplicando os princípios da Prova de Trabalho. Ele determina com precisão o armazenamento da última cópia atualizada do razão e do software neste node (Prova de Capacidade), calculando a soma de verificação dos valores de todo o conteúdo - o código de hash. O tamanho dos arquivos também está determinado, como a prova de que esta é a última cópia atualizada e um código hash da última transação registrada no sistema.

Para se tornar o node da rede principal, o node procura o valor da função hash que calcula com base no último livro contábil armazenado. Organizamos um ambiente competitivo saudável entre os nodes de rede para a oportunidade de se tornar o node principal, gerar e armazenar um novo razão.

Depois de calcular a função e obter o resultado, é enviado para todos os nodes de rede para verificação. O resultado contém um carimbo de data / hora do cálculo e um valor baseado no cálculo da função dos arquivos e software do razão. Todos os nodes recebem o valor calculado, compare o tempo de cálculo alocado para a busca do servidor de rede principal, verifique-o e confirme o fator de confiança do node e também confirme sua oportunidade de participar da competição - para se tornar o principal node da rede.

Depois de receber a aprovação de todos os nodes de rede, uma lista é formada por nodes que calcularam corretamente o valor da função e contém um timestamp. O node que recebeu o resultado correto e o aprovou no tempo mais rápido, torna-se o node da rede m ain do momento.

O conceito de algoritmo SHA2 é usado para calcular a soma hash do arquivo.

As funções Hash da família SHA2 são construídas com base na estrutura MerkleDamgard.

A mensagem inicial após a adição é dividida em blocos, cada bloco é dividido em 16 palavras. O algoritmo passa cada bloco de mensagem através de um ciclo com 64 ou 80 iterações (rodadas). Em cada iteração, 2 palavras são convertidas e o resto das palavras definem a função de conversão. Os resultados de cada processo de bloco são resumidos. A soma é o valor da função hash. No entanto, o estado interno é inicializado com base nos resultados do processamento de bloco anterior. Portanto, é impossível processar blocos independentemente e resumir os resultados.

Equipamento de nodes de rede

Estamos nos esforçando para construir uma plataforma com as características de processamento de transações mais rápidas, portanto, propomos usar um incentivo material para manter os nodes de rede nas melhores condições: equipamentos de servidor de alto desempenho e uma alta largura de banda alta.

Como uma compensação material, o proprietário do node da rede principal receberá uma remuneração na moeda CREDITS a partir de várias comissões por transações deste cronograma processado. O restante (½) destina-se ao orçamento geral de desenvolvimento do projeto para suporte ao usuário, recursos atuais e desenvolvimento de novos produtos. A porcentagem pode ser alterada, bem como separada para o sistema de formação de taxa através da votação federativa pelos nodes de rede, após a oferta inicial de moedas por pelo menos três anos.

Como resultado, incentivamos os proprietários do servidor a manter este servidor no hardware com o maior desempenho e a manter um canal de comunicação de alta qualidade e alta velocidade.

Construção de consenso

Como resultado, temos o node de rede principal selecionado por todos os nodes. As principais tarefas do node principal são: obter transações no status do candidato para adicionar ao ledger de todos os nodes, processá-los, construir o último livro relevante e enviar um ledger recém-construído a todos os nodes da rede. O processo de manipulação de transações e construção do último livro relevante é precisamente a busca de uma solução de consenso. O resultado da construção do último livro de contas relevante é a solução consensual.

Todo o processo pode ser dividido nas seguintes etapas:

1. Procure o node da rede principal;

2. Construção de nodes confiáveis;

3. Receber a lista de transações e elaborar uma lista de candidatos para adição ao razão;

4. Processando a lista de candidatos, votação de nodes (nodes confiáveis e comuns têm diferentes fatores de peso (fator de confiança);

5. Remoção da lista de candidatos de transações não confirmadas que não foram verificadas ou que tenham uma confirmação negativa;

6. Construindo uma lista de transações confirmadas a serem adicionadas ao razão;

7. Adicionando transações ao razão maior com o timestamp e o código hash do bloco que continha a transação;

8. Enviando o bloco com transações para todos os nodes da rede. Quando recebido, é adicionado aos registros de todos os nodes.

Criando e iniciando a rede

Todo o processo pode ser descrito na seguinte sequência:

1. O usuário final da rede no sistema gera uma transação.

2. Quando todas as condições do contrato inteligente especificado nela são atendidas, o usuário inicia a ação (transação) ao chamar o método necessário usando o software da plataforma.

3. Para seguir os princípios fundamentais da blockchain, o kernel de validadores acompanha a sincronização e a invariância da versão mais recente.

4. No momento da construção do consenso, todas as transações recebidas durante o ciclo são coletadas no bloco.

5. Um número é atribuído ao bloco, consistindo de um timestamp e um identificador de node convertido num código de hash e, em seguida, o bloco é colocado no módulo de consenso.

6. Após a compilação da lista branca de candidatos, não apenas o hash da transação é gravado no razão, mas também o hash do bloco, para certificar sempre a fonte com base nela.

7. Este hash é um tipo de assinatura do bloco e aquele que criou esse bloco com transações.

8. Após a construção de consenso usando um algoritmo de pesquisa federativa, as transações adicionadas ao bloco são passadas para o kernel do validador para serem gravadas no razão.

Transações não incluídas no registro

As transações não incluídas na lista de transações prontas são marcadas como rejeitadas. Informações sobre isso são exibidas imediatamente no remetente (iniciador) da transação. As transações não incluídas no livro maior permanecem no conjunto de candidatos e são armazenadas nos nodes da rede. Todas as novas transações recebidas pelo servidor no momento do consenso também chegam lá, e então o processo de busca começa de novo. Tal operação cíclica contínua da rede permite realizar transações por um período de tempo bastante curto, mantendo um alto grau de confiabilidade e relevância da informação.

3. Processamento de transações

Transações

Uma transação é a unidade mínima do sistema informando a plataforma da execução de métodos de contrato ou transferências diretas entre contas sem criar um contrato inteligente, seguindo a colocação do resultado na rede peertopeer.

Construção de consenso

O sistema usa um modelo federativo para construir um consenso - a votação de nodes de validadores confiáveis, e também o algoritmo de construção de consenso - um algoritmo para a passagem de um autômato de fintastate. O consenso funciona por ciclos (etapas de tempo), por etapa de tempo, as transações são extraídas e colocadas num pool (matriz onedimensional). Depois de serem colocados no pool, todas as transações são enviadas para nodes confiáveis para receber uma resposta. Se a resposta for recebida, a transação para a qual o pedido foi adicionado pode ser adicionada ao razão desse validador. Depois disso, é enviado para o próximo validador na rede. Quando o consenso é construído - no final da cadeia onde a legalidade da transferência é totalmente confirmada, a transação é enviada para validação com uma marca para gravação e gravação no razão.

Processamento de transações

Para alcançar a natureza descentralizada do sistema, cada servidor deve possuir o armazenamento do razão e também ser um manipulador completo de todas as transações.

O sistema usa o conceito de kernels do sistema. Por kernels, nos referimos a um manipulador de dados que executa uma tarefa de produção específica, independentemente da disponibilidade e operabilidade dos demais componentes do sistema. Cada kernel, na entrada, no momento em que a tarefa é executada, recebe uma lista de variáveis para processamento. E sempre obtém um resultado na saída - positivo, qualquer outro ou um erro. Como resultado, o kernel do sistema sempre contém o código de resposta, além do conjunto de dados principal. Essa estrutura é necessária para a maior velocidade possível de cada processo, que deve funcionar independentemente um do outro.

Estrutura de Entrada de Rede

Para alcançar um desempenho importante do livro maior, mas, ao mesmo tempo, sem comprometer a segurança, propomos a utilização de uma base de dados do livro de contas, sem construir a árvore Merkle a partir do código hash do bloco anterior e do resultado da transação.

Merkle tree (TTH - Tiger Tree Hashing) é um tipo de função hash usada para verificar a integridade dos dados, obter um identificador exclusivo da cadeia e restaurar a seqüência. Os dados são divididos em pequenas partes - blocos que são individualmente esboçados usando Leaf Tiger Hash, então o Tiger Tiger interno é calculado a partir de cada par de hashes onebyone. Se o hash não tiver um par, então ele será transferido para a nova cadeia inalterada. Em seguida, o Tiger Hash interno é calculado novamente na cadeia para cada par. Este procedimento é repetido até que haja um hash para a esquerda.

Quando o razão é operado usando árvores Merkle, a velocidade de processamento da transação é muito baixa e a carga em recursos de computação é muito alta. Em nossa opinião, este não é um uso racional do armazenamento de dados.

CREDITS estrutura de rede

Oferecemos abandonar as árvores de Merkle e usar o livro de transações no sistema CREDITS, com cada entrada consistindo de um código de hash do bloco de transação para adicionar à lista de candidatos, além do livro-razão. Além disso, a entrada tem o identificador do node e o carimbo de data / hora quando ele foi gerado. A entrada do livro contabilístico contém a direção da transação, suas contas iniciais e finais, o tipo de writeoff, o número de unidades de writeoff, o tipo de depósito e o número de unidades de depósito. Este princípio aumenta a velocidade do processamento de transações, aumenta a complexidade da mudança do livro-razão ilegítimo e exclui as possíveis mudanças na entrada do livro-geral com retrospectiva.

Tamanho de Bloco

A unidade de tempo é o ciclo de busca dos nodes principais e confiáveis, e o tempo do ciclo é calculado dependendo da complexidade da rede. Por unidade de tempo, a rede contém N transações geradas e transferidas para processamento para a rede no final do ciclo anterior, até o início do ciclo seguinte, para obter o status de "Candidato a ser adicionado ao razão". As transações selecionadas da rede N são colocadas no bloco. O tamanho do bloco depende do número de transações nisso.

Pesquisar participantes da transação

CRÉDITOS A rede peertopeer pode ser representada como um gráfico, com contas de usuário sob a forma de vértices e uma infinidade de possíveis transações sob a forma de bordas direcionadas que conectam dois vértices (conta). Uma vez que todas as arestas têm um vértice inicial e um vértice terminal, você sempre pode construir um gráfico orientado (orgraph).

Se tomarmos as seguintes condições de identificação:

- Qualquer transação sempre possui um remetente e um receptor;
- Qualquer vértice (conta) pode sempre ser conectado a outro vértice com uma borda direcionada (transação);
- Qualquer vértice do gráfico (conta) tem um número finito de bordas direcionadas (transações de entrada e de saída).

Em conexão com o que precede, podemos dizer que o orgraph contém a rota necessária para cumprir as condições de transação necessárias e construir uma cadeia simples. Uma vez que é uma sequência finita de vértices, onde cada vértice (exceto o último) está conectado ao próximo vértice na seqüência por uma borda.

Canal de transmissão de dados

Cada canal de comunicação entre o node de rede principal eo node comum da rede CREDITS é um segmento separado (multithreading), dentro do qual os dados são enviados em forma criptografada quando a transação é executada.

Para garantir a segurança da rede, todos os dados entre os nodes do validador são transmitidos de forma criptografada e cada conexão entre nodes é de baixo nível com base na biblioteca de rede. Se a transferência de dados ocorrer com um erro, o segmento deve ser automaticamente interrompido, a entrada correspondente é colocada para gravação no sistema de log e, em seguida, para o arquivo de log. Os dados são transmitidos através de variáveis tipificadas. Os dados transmitidos são criptografados usando o algoritmo RC4 simétrico. Como este algoritmo funciona sob uma chave secreta comum, essa

chave é transferida quando uma conexão é criada entre nodes e é transmitida numa forma criptografada de acordo com o algoritmo DiffieHellman.

O algoritmo RC4, como qualquer criptografia de fluxo, é construído com base num gerador de bits pseudorandom. A chave é escrita na entrada do gerador, e os bits do pseudorandom são lidos na saída. O comprimento da chave pode ser de 40 a 2048 bits. Os bits gerados têm uma distribuição uniforme.

O algoritmo DiffieHellman permite que duas partes recebam uma chave secreta comum usando um canal desprotegido de ouvir, mas protegido contra a mudança de canal de comunicação. A chave recebida pode ser usada para trocar mensagens usando criptografia simétrica. O algoritmo é baseado na complexidade da computação de logaritmos discretos. Nela, como em muitos outros algoritmos com uma chave pública, os cálculos são realizados em módulo para um certo número P.

Primeiro, um certo número natural A, menor do que P, é selecionado de forma especial. Se quisermos criptografar o valor X, então nodes calculamos

$$Y = AX \bmod P.$$

E é fácil calcular Y tendo X. O problema inverso de calcular X de Y é bastante complicado. O Exponente X é exatamente chamado de logaritmo discreto Y. Assim, sabendo a complexidade do cálculo do logaritmo discreto, o número Y pode ser transmitido publicamente em qualquer canal de comunicação, pois com um módulo grande P o valor inicial X será quase impossível de escolher. o

O algoritmo DiffieHellman para gerar uma chave é obtido neste fato matemático.

Qualquer ação no sistema está vinculada ao timestamp, ao número do bloco anterior, ao login do usuário e ao ID do contrato inteligente. Isso permite encontrar duplicatas ao executar. Se uma duplicata for encontrada, então tomamos a primeira transação do grupo, o restante é considerado ilegítimo.

Ação no Sistema

Uma ação no sistema é uma transação que caracteriza a transferência mais simples do valor da conta para a conta ou a transferência do resultado do método do contrato para o validador, para a busca subseqüente de uma solução no subsistema de pesquisa de consenso.

Para evitar a duplicação da transação no mesmo bloco com o mesmo identificador, o sistema aceita o acordo de que a única transação verdadeira e correta é a que veio primeiro ao subsistema de validação para processamento. Uma vez que já está registrado no sistema de validação que uma transação já foi feita a partir da conta atual e não há valores na conta para realizar a transação, um consenso não pode ser encontrado. Assim, o problema do duplo desperdício é resolvido.

Quando a transação é executada, as informações são recebidas para o validador e confirmadas, as informações sobre a mudança de status do razão geradora são distribuídas automaticamente para todos os nodes da lista confiável, após o qual o razão é sincronizado.

Para sempre ter um ledger de transação uptodate entre todos os nodes confiáveis para o node de validador atual, é necessário sincronizar a transação recém-chegada no ledger de todos os nodes de cada vez. Para resolver este problema, uma porta separada para sincronização deve ser usada (se houver tal oportunidade). Esta oportunidade aumentará a velocidade de processamento das informações recebidas no kernel do validador devido à distribuição da carga na porta. O segmento de sincronização sempre é executado, é cíclico. A prioridade para a alocação de RAM e carga da CPU (usando ciclos de CPU) é menor do que a média. A memória armazena as últimas 1.000 operações e o estado das contas para elas (numa forma criptografada usando um algoritmo síncrono), isso aumenta a velocidade de resposta às solicitações de outros nodes de validador.

Adicionando uma transação para validação

A adição de transações ao ledger é chamada apenas do subsistema de validação imediatamente após a construção do consenso e compilação de uma lista branca com o resultado de uma economia de transações no ledger.

Ligar de sistemas de terceirização é impossível, eu posso melhorar a segurança.

Parâmetros de entrada - o objeto que caracteriza a transação. O valor resultante ResultValue <0 - a execução é interrompida com um erro, o valor resultante é um possível código de erro / 0 <ResultValue - a função foi executada sem erros, o r esult é o número da entrada no ledger.

Parâmetro de entrada - o objeto que contém o rótulo exclusivo da transação, o remetente, o destinatário, o valor transferido, a correspondência do valor, o valor desejado, a quantidade do valor transferido, a quantidade do valor desejado e outras informações do sistema que podem ser alterado, se necessário.

Custo das Transações

O sistema usa a moeda CREDITS, que serve:

- Como um meio de pagamento interno para o uso do sistema;
- Para trocar moedas diferentes no sistema;
- Para trocar vários valores dentro do sistema;
- Para criar e processar operações em c ontracts inteligentes;
- Para comprar informações de fontes de terceiros para serviços dentro do sistema.

O custo de uma transação pode variar dependendo da carga da rede, num usuário particular do sistema, o que teoricamente pode direcionar um fluxo enorme de transações num determinado horário de pico. Sugerimos usar o método do material e o impacto nos usuários do sistema para controlar a carga da rede.

O custo de realizar transações nos primeiros três anos da operação do sistema será definido individualmente para diferentes tipos de transações e operações. No futuro, será desenvolvido um algoritmo para a geração automática do custo da transação.

4. Contratos Inteligentes

Introdução

Um contrato inteligente no sistema CREDITS é um algoritmo eletrônico que descreve um conjunto de condições pelas quais ações e eventos nos sistemas reais mundiais ou digitais podem ser associados.

Para implementar contratos inteligentes autocontrolados, é necessário um ambiente descentralizado que exclui completamente o fator humano, e para usar a transferência do custo de um contrato inteligente, é necessária uma cryptocurrency independente da autoridade central.

Entidades

Um contrato inteligente em CREDITS consiste nas seguintes entidades:

1. Propriedade (variáveis públicas) - a entidade do sistema que armazena dados públicos necessários ao trabalho do contrato no sistema CREDITS.
2. O método é a entidade do sistema CREDITS responsável por observar a lógica e a seqüência de ações ao realizar a transação (ações sob o contrato).

Os participantes no sistema CREDITS assinam os contratos inteligentes usando a chamada de método que modifica as propriedades do contrato, iniciando os processos para verificar o cumprimento das condições e coordenação.

Um contrato inteligente entra em vigor após a assinatura das partes. Para garantir o cumprimento automatizado das obrigações, é necessário um ambiente de existência que automatize totalmente a execução dos termos contratuais. Isso significa que os contratos inteligentes podem existir apenas dentro de um ambiente que tenha acesso livre ao código executável para os itens do contrato inteligente.

Todos os termos do contrato devem ter uma descrição matemática e uma lógica clara de execução. Assim, o principal princípio de um contrato inteligente é a automação completa e a confiabilidade das relações contratuais entre as partes.

Método de Contrato Inteligente

O método de contrato inteligente CREDITS é a entidade do sistema responsável pelo cumprimento da lógica e seqüência de ações durante a transação (ações sob o contrato).

A lógica e a seqüência de ações são descritas por um código de programa (módulo) contendo comandos, sua execução seqüencial permite obter o resultado desejado. Este código pode lidar com comandos do sistema (por exemplo, o comando de atribuição), comandos do usuário (funções escritas separadas), propriedades do contrato (variáveis estaticamente ou dinamicamente inicializadas disponíveis a partir de qualquer método do contrato) e métodos de qualquer outro contrato de terceirização disponível apenas para o proprietário do contrato conectado (terceira parte). Para mais divulgação, o desenvolvimento é fornecido em linguagens de script (por exemplo, JavaScript).

O método (código de programa) permite o uso de todos os operadores de linguagem de script (comandos) amplamente utilizados (atribuição, saltos condicionais e incondicionais), criação de funções e procedimentos (sub-rotinas), conexão de bibliotecas de terceiros.

Máquina Executável Virtual

O método do contrato do sistema CREDITS é executado no ambiente virtual do sistema (Máquina Virtual, a seguir denominada VM). Quando um método é chamado para um contrato particular, a VM aloca uma área de memória e carrega o bytecode contrato contida contendo os métodos e as variáveis inicializadas (ou redefinidas ao chamar outros métodos de contrato). A VM começa a processar o bytecode do método, no tempo de execução, as variáveis e o código são carregados na sua área de memória, e os comandos são executados sucessivamente, o resultado é transferido para a rede peertopeer para posterior colocação no ledger.

O iniciador do método de execução é o usuário do sistema, em nome do qual esse método é lançado.

Termo de Valor

CREDITS cryptocurrency também é um indicador do termo de valor de uma unidade de contrato para comparar duas unidades completamente diferentes e construir um consenso ao executar ou aceitar o contrato pelas partes. Em vez de registrar cada combinação de valor / gateway separado, CREDITS cryptocurrency serve como um monte para efetuar transferências de valor. Isso é possível porque qualquer valor é líquido em relação à moeda CREDITS, o que significa que qualquer valor pode ser líquido em relação a qualquer outro valor.

Executando os Termos do Contrato Inteligente

O termo do contrato no sistema CREDITS é o valor dos campos de gatilho (verificado) necessários para fechar (concluir) o contrato.

Cumprimento dos termos do contrato inteligente é um procedimento quando os campos de gatilho (desejado) são verificados quanto ao valor desejado equivalente. Existem três maneiras possíveis de encontrar uma solução para cumprir os termos do contrato:

1. O contrato é celebrado entre duas ou mais partes para a transferência de valor. Nesse caso, o cumprimento do contrato é a provisão do custo equivalente ao valor para a parte transferente da parte receptora.
2. O contrato é celebrado entre as partes para a transferência de valor, mas o pagamento deve ser feito após cumprir um certo número de condições (por exemplo, entrega de valor para a parte receptora).
3. Um contrato de conversão de um valor para outro com um custo equivalente na forma de CREDITS é colocado no sistema. Nesse caso, a plataforma começa a procurar o caminho mais curto possível para trocar um valor por outro através da conversão em outros contratos. Qualquer cumprimento do contrato pode ser fornecido por uma transação, ou por várias transações, o que proporcionará uma oportunidade para coletar a quantidade necessária de unidades de valor para concluir o contrato.

Fontes de dados

Para o trabalho correto e totalmente executado, verificando e fornecendo informações adicionais, para fazer uma solução mais equilibrada e otimizada, o CREDITS usa provedores de dados de terceiros. A necessidade de introduzir fontes de dados adicionais no sistema deve-se à inadequação da informação pública sobre uma ou várias partes contratadas (por exemplo, obtendo o status de crédito do mutuário para tomar a decisão de emitir um crédito).

Para trabalhar com sistemas de informação de terceiros, a plataforma pode chamar um ônibus de integração, que por acesso remoto gera uma solicitação para um sistema de terceiros (site) num formato para apresentação de dados numa base paga para os participantes do sistema com pagamento i n CRÉDITOS.

O pedido é enviado de forma criptografada para portas e endereços fornecidos por sistemas de informação diferentes dos padrões. O resultado da solicitação pode ser qualquer resposta ao serviço que contém as informações necessárias para tomar uma decisão, ou um código de erro que caracteriza a impossibilidade de receber a resposta necessária e as possíveis etapas para eliminar erros.

5. Plano de implementação

Plano Técnico de Implementação do Projeto

	S1	S2	S3	S4	S5
	PreAlfa	Alfa	Beta	Lançamento do candidato	Lançamento
Armazenamento, Consenso Consenso de AMF	FA: chave desenhar Implementação	mFA: chave desenhar Implementação PoW (ProofofWor k) e PoC (cidade de ProofofCapa)	mFA Otimização	–	–
Banco de dados	Decentralização do Razão, implementação NoSQL Store	Histórico do MessagePack	–	Backup Blockchain	–
CVM (máquina virtual de créditos)	Design e implementação	Integração com o ecossistema	Otimização	verificar erros	–
Sistema de terceiros	–	Design e implementação	integrar-se a sistema	–	Otimização
Motor de inferência	Especificação Formal e Design Chave Elementos	Reasoner Integração com Blockchain	Otimização do Reasoner	–	–
Interface de usuário	Implementação	Design da Web UX	–	–	–
Wallet	Especificação Formal da Carteira		Design UX Aplicação Teste	–	Android, iOS, Desktop Wallets
RPC & REST API	Especificação formal	Blockchain Explorer	–	Explorador de terceiros	–

CREDITS cryptocurrency

Após o lançamento da versão de lançamento do sistema, será emitida uma quantidade fixa de 1.000.000.000 de CREDITS. Eles serão trocados por tokens padrão ERC20, emitidos na venda inicial

de token. Eles serão trocados a uma taxa de câmbio fixa: 1 token padrão ERC20 = 1 unidade monetária de CREDITS.