1 Introdução

A Plataforma de Tecnologia FunFair.io é um projeto ambicioso concebido para fornecer a próxima geração de jogos em qualquer lugar do mundo. Este documento discute as metas de projeto, execuções e os próximos passos. Para uma visão geral completa da plataforma FunFair.io, consulte nosso whitepaper em https://funfair.io/.

2 Termos de Utilização

Parcelas do desenvolvimento das tecnologias discutidas aqui podem ser protegidos por patentes pendentes nos EUA, a União Europeia, Ásia ou de outros mercados a nível mundial. Todas as informações aqui presentes estão orientadas para o futuro e são de natureza especulativa e podem mudar em melhor resposta a numerosas forças externas, incluindo tecnologia, regulamentações e mudanças de mercado.

3 Metas

A nossa missão principal é construir uma plataforma e protocolo para a próxima geração de entretenimento de jogos online. Todos os jogos na plataforma devem satisfazer os nossos critérios rigorosos:

- **Porcentualmente justo** que deveria contar com a jogabilidade blockchain para verificar a idoneidade.
- **Sem instituições financeiras envolvidas**, eles devem confiar na blockchain para armazenar transações;
- Tão **rápido** que o tempo de resposta entre o usuário pressionar qualquer botão e ver o resultado deve tender a zero;
- **Os jogos tem que ser flexíveis**. Permitindo qualquer tipo de decisão durante uma sessão de jogo, por parte do usuário, como: Mudar o tamanho das apostas de jogo para jogo, poder jogar imediatamente com os prémios e adicionar fundos durante a sessão

3.1 Diversão

Nós não queremos apenas jogos simples e chatos onde o usuário só pode "escolher um número". Estes são endêmicos para jogos blockchain e eles simplesmente não são divertidos. Mas nós somos divertidos. A indústria do jogo passou dezenas de anos refinando jogos divertidos para seus clientes - nós pensamos que é errado ignorar essas lições. Portanto, a plataforma deve:

- Apoiar jogos como o Blackjack, com a capacidade de dividir, dobrar e jogar em vários lugares;

- Suportar jogos onde o estado persiste entre as rodadas individuais.

4 Geração de Números Aleatórios

O núcleo de qualquer jogo de Casino é um gerador aleatório de número (RNG). Todos os softwares RNG são - efetivamente - pseudo aleatórios (PRNGs).

Os PRNGs na blockchain possuem muitos problemas:

Por transação: Alguns jogos de geração aleatória na blockchain geram uma única jogada por vez, o que é uma abordagem lenta e dispendiosa. Isto torna o jogo inviável para os jogadores e operadores. Além disso, estes normalmente exigem uma "confirmação e revelação" diária de informações secretas. O jogador pode verificar após o fato de que o operador não trapaceou, mas

deve confiar durante o jogo. Os jogos antigos que usam esta configuração como o SatoshiDice podem custar até US $ 5 para jogar antes de apostar qualquer aposta e levar até um dia por aposta.

Oráculos Outros jogos baseados em blockchain usam oráculos para gerar aleatoriedade. Estas também são lentas, limitadas às velocidades de confirmação da blockchain e adicionam questões de incentivo difíceis sobre os operadores, seus funcionários e servidores.

Block Hashes Os piores jogos usam Block Hashes para gerar um número aleatório. Lentos e extremamente vulneráveis ao ataque de mineros.

Nenhuma dessas tecnologias podem criar jogos convincentes em 2017.

5 Nossa solução: Canais de Sorte

Nós inventamos um sistema percentualmente justo usando “canais de sorte” contendo seeds RNG parciais pré-comprometidas fornecidas separadamente pelo jogador e pelo operador. Essas seeds parciais são combinadas com a blockchain no início da sessão de jogos.

Os “canais de sorte” funcionam permitindo que os participantes se envolvam em uma resposta rápida e livre de "reivindicações" atualizadas em um montante bloqueado de fundos. O protocolo Bitcoin Lightning foi o primeiro a popularizar a idéia. Em blockchains compatíveis com EthereumVM, os “canais de sorte” são muito simples de criar e implementar - um smart contract possui fundos bloqueados e, em seguida, libera quando os participantes do “canal de sorte” solicitarem.

Durante o jogo, nós criamos um "Canal de sorte"; Um "Canal de sorte" com a capacidade adicional de verificar um esquema de revelação progressiva por ambas as partes, avançando uma seqüência determinista mas imprevisível de números aleatórios.

Detalhes sobre nosso modelo de "Canal de sorte" serão disponibilizados durante nosso Token Sale. Por enquanto, sentimos que nos dá e nossos detentores de token uma vantagem competitiva para manter os detalhes de implementação privados.

6 Histórico de Desenvolvimento

Nós começamos com um caça-níquel relativamente complexo: 30 linhas de vitórias, wild cards, símbolos de dispersão e um modo de bônus com rotações grátis.

Muitas implementações de caça-níqueis online determinam a posição dos rolos primeiro decidindo um resultado e - em seguida - animando os rolos para uma posição apropriada, mas essa abordagem

Uma solução de "escolher um número e depois animar" é sem saída tecnológica. Jogos mais complexos, como o Blackjack e a Roleta, não se traduzem facilmente nesse método, por isso implementamos o "caminho difícil" em preparação.

6.1 FunFair Slot v0.1

Nós codificamos as regras do caça-níquel na blockchain. Inicialmente, construímos um sistema no qual um smart contract escolhe um bloco futuro para ser usado como seed para um gerador de números aleatórios (RNG). Quando o bloco apropriado é minerado, o contrato usa o hash do bloco minerado anteriormente, experimentado apropriadamente para gerar entropia e, em seguida, caminha através de uma "Máquina de Destino", codificada no smart contract. Esta "Máquina de Destino" determina a posição de cada uma das bobinas da máquina caça-níquel.

Conforme discutido, este é um método ruim para obter números aleatórios. Este método tem riscos para os operadores se os mineros estiverem jogando - os mineros podem ter incentivo para descartar um bloco minado se contiverem uma perda, dependendo de uma série de fatores extrínsecos e intrínsecos.

6.2 Slot v0.1 Enhancements

Em seguida, estendemos o jogo básico de duas maneiras:

Multi-Spin O usuário pode inserir fundos para mais que um giro(spin). A entropia gerada pode ser usada para determinar uma seqüência de giros ao invés de apenas um de cada vez. Este é o núcleo da nossa primeira inovação que produz jogos instantâneos.

Free Spins Uma vez que conseguimos criar rotações múltiplas através da semente inicial, ampliamos as capacidades da máquina para adicionar rotações, ainda semeamos o mesmo número aleatório.

Este sistema foi construído e funciona como pretendido. Construímos uma plataforma elegante com base na web, incluindo animações de símbolos e efeitos para dar uma indicação da qualidade final do produto que podemos construir. Veja imagens e links para vídeos em https://funfair.io/whitepaper.

6.3 Problemas na v0.1

O projeto inicial foi um sucesso e provamos que era possível construir uma máquina caça-níquel de alta qualidade, diversão e instantaneamente receptiva na blockchain.

Mas este sistema falhou nos nossos principais requisitos, pois era vulnerável pelos mineiros, não permitindo decisões no meio do jogo e possuía um custo do gás proibitivo.

6.4 Metas de Design na v0.2

O esquema PRNG na versão 0.1 não suportou nosso objetivo de gerar jogos divertidos, rápidos, justos e flexíveis. Olhando uma solução alternativa, nós incorporamos três áreas de pesquisa nas comunidades de blockchain e jogos.

"Canais de Estado" Os "Canais de Estado" permitem que micro-transações sejam manipuladas diretamente, fora da blockchain(off-chain). Os fundos são comprometidos e bloqueados antecipadamente em um smart contract. Em seguida, as transações ocorrem entre as partes, progressivamente assinadas por todos. O status final é comprometido de volta à cadeia de uma

maneira que prova que todas as partes concordam com o resultado.

Percentualmente Justo Aqui o conceito básico é que alguma entropia está comprometida com a blockchain com antecedência, mas criptografada. Uma vez que a jogada ocorreu, esta entropia pode ser revelada como sendo a fonte do RNG usado para o jogo que, em combinação com a codificação das regras do jogo em um smart contract, pode ser usado para provar a equidade do jogo.

Cadeia de Hash Reverso Para criar uma cadeia de hash, escolhe-se uma seed segura inicial e o processo é repetido muitas vezes até que se revele o último hash. No momento da revelação, você criou uma série de números aleatórios secretos que podem ser revelados sequencialmente, um de cada vez e de trás para frente. O revelador não pode enganar ou mudar de idéia sem romper o algoritmo de hash - eles devem revelar o número anterior em seqüência, e este número pode ser verificado facilmente como o número secreto "anterior" correto.

6.5 Canais de Sorte v0.2

Para alimentar a geração de números aleatórios por trás da máquina Caça-Níquel da FunFair e, finalmente, por trás de todos os jogos em nossa plataforma nós inventamos um gerador de números aleatórios que opera fora da blockchain. Existe como um canal de estado durante a duração da sessão de jogos e oferece suporte a mensagens em tempo real entre o cliente da FunFair e o servidor. Chamamos esses canais de "Canais de Sorte".

6.5.1 A configuração do canal

Os canais de sorte suportam a comunicação em uma sessão de jogos entre o Player (cliente) e o Casino (operador). Eles fornecem um método rápido de baixo custo para a geração de números aleatórios, iniciando sessões de jogo, terminando e estabelecendo smart contracts na blockchain.

Para cada jogo, uma API do smart contract é chamada encapsulando as regras do jogo. O abrangente canal de sorte faz primeiramente duas coisas: Inicia uma sessão do jogo e resolve - ou termina - uma sessão.

6.5.2 O Cliente

O cliente é uma aplicação web, Javascript e HTML5 / WebGL que pode se comunicar com a blockchain e o Servidor. No futuro, clientes poderão ser aplicações nativas nos dispositivos móveis, mas, por enquanto, eles podem trabalhar diretamente dentro de um navegador móvel, dando acesso à mais ampla gama possível de clientes.

6.5.3 O Servidor

O servidor é muito similar ao cliente, mas não possui interação direta do usuário.

Após o jogador iniciar uma sessão do jogo, tanto o cliente quanto o servidor criam uma cadeia de hash reverso. O contrato "Canal de Sorte" verifica todas as interações dos jogadores e cria uma

sessão na blockchain, bloqueandos os fundos e publicando o evento na blockchain.

6.5.4 O Protocolo

Durante o progresso do jogo, o cliente e o servidor trocam mensagem, assinadas pelo remetente.

Se o estado da sessão tiver avançado em uma determinada mensagem, o novo estado é assinado e o estado anterior (já assinado pela outra parte) é co-assinado. Para cada resultado em uma sessão de jogo, o Servidor cria o RNG puxando o próximo hash de sua Cadeia Reversa, combinando-o com o próximo hash do cliente (fornecido pelo cliente) e tomando o hash desse.

Em seguida, executa a lógica do jogo usando este RNG - determinando uma ação na máquina de estado do jogo. Enquanto isso, o Cliente usa os mesmos hashes para gerar o RNG da mesma maneira que o Servidor fez. Ele usa sua própria implementação da lógica do jogo para verificar ganhos e outros resultados do jogo correspondem aos do Servidor.

Para finalizar uma sessão, o Jogador pressiona um botão de retirada de caixa. O cliente assina o estado negociado mais recentemente, o servidor co-assina e chama o contrato "Canal de Sorte" com uma mensagem "End Session". O contrato verifica amplamente todos os dados relevantes. Se tudo estiver bem, o contrato paga o saldo final e termina a sessão.

Essa implementação inicial permite segurança e um jogo justo. No entanto, também gostaríamos de validar isso na blockchain e não apenas off-chain. Para fazer isso, precisamos de dois métodos de contrato adicionais:

Verificação na Cadeia Para verificar cada uma das posições de Hash Reversas criadas pelo Cliente e Servidor, o último hash usado é postado na cadeia - o smart contract "Canal de Sorte" pode, então, ter o número correto de vezes e recriar o "hash final" da cadeia.

Verificação de Jogo Para verificar um jogo individual, implementamos a máquina de estado do jogo em um contrato separado na blockchain, um por tipo de jogo. Isso pode ser uma função constante, tomando as seeds iniciais, o jogo e a saída do jogo, e retornando verdadeiro ou falso. Porque as funções constantes não modificam a blockchain e elas são gratuitas para chamar e executar para todos os participantes.

7 Canais de Sorte v2

O próximo passo do desenvolvimento vai trazer duas melhorias:

Endereço de Canal Efêmero No "Canal de Sorte" versão 1, o cliente gera chaves temporárias para assinar as mensagens do canal. Os clientes podem esquecer sua chave de assinatura de endereço se eles esperam muito ou não desejam reescrever uma transação de seu cliente web3 para reinvidicar os fundos. Os clientes podem sair porque as solicitações de interface aumentam a fricção ou simplesmente porque elas se distraem e fecham uma janela do navegador. A chave de assinatura do cliente pode ser armazenada criptografada no contrato do "Canal de Sorte" ao lado da sessão. Acreditamos que isso poderia ser ampliado para recuperar sessões parciais também.

Máquina de Estado Unificada "Canais de Sorte" v1 armazenam a lógica em três lugares: o smart contract, o Cliente e o Servidor. Às vezes estes são implementados em diferentes linguagens - Solidity, Javascript e - atualmente - C#. Nós estamos explorando como unificá-los.

As possibilidades incluem uma estrutura de máquina de estado padronizada ou um instrumento de Cliente e Servidor para usar apenas as funções constantes da blockchain do Ethereum.

8 "Canais de Sorte" v3: Jogos com múltiplos estados

O Caça Níquel da FunFair é uma prova de que o "Canal de Sorte" pode alimentar um jogo mais rápido e mais justo que não acumula os custos de transação ou que gera spam na blockchain. Somos a única empresa de jogos experiente no mundo a chegar até aqui. Iremos mais longe. Máquinas caça-níquel, jogos de dados e roleta são "fogo e esqueça": uma vez que um jogador gira a roda ou rola os dados, não há nada que ela possa fazer no meio do jogo para mudar sua aposta ou afetar o resultado que está por vir.

Os jogos "multi-estado" são mais populares a nível mundial. Quando os jogadores podem afetar o jogo de alguma forma, talvez mudando sua aposta (Blackjack), ou baseando uma nova aposta em resultados anteriores (Craps), eles estão mais envolvidos.

Para suportar jogos multi-state como o Blackjack, o FunFair irá expandir o protocolo "Canal de Sorte". O "Canal de Sorte" v3, incorporando essas mudanças, está com o lançamento programado no Q3 2017.

9 "Canais de Sorte" v4: Completamente P2P

Os "Canais de Sorte" imaginam um servidor e um cliente. O FunFair pretende permitir que os operadores executem um Servidor diretamente como parte de um processo simples de dois cliques. Esta capacidade de permitir que qualquer um lance e opere seu próprio ambiente de jogo é uma parte fundamental da nossa proposição de valor: tornar mais fácil para operadores de cauda longa, grupos particulares, celebridades e outros para operar jogos de qualidade é o que fazemos.

Gostaríamos de remover o componente Servidor completamente da especificação do "Canal de Sorte"e acreditar que temos uma maneira de fazer isso. Se tivermos sucesso, esses operadores obterão benefícios completos da natureza distribuída e descentralizada da blockchain.

Esta é uma inovação tecnicamente desafiadora, e estamos apenas com 95% de certeza de que podemos fazê-lo, mas as primeiras indicações são boas, e o benefício para todas as partes interessadas da FunFair é imenso.

Atualmente, estamos pensando para o Q4 2017 para jogar P2P completo sem servidores, tudo isso cumprindo os nossos principais requisitos, que cada jogo seja divertido, rápido e justo.

10 Nota sobre a Criptografia Limiar (Threshold Cryptography)

A Threshold Cryptography, popularizada mundialmente pela String Labs para o projeto Dfinity, mantém a promessa de um número aleatório verdadeiramente seguro, pelo menos um por bloco. Essas primitivas criptográficas estão sendo incorporadas no lançamento da Ethereum's Metropolis.

Poderíamoss atualizar nossos mecanismos para incorporar essa tecnologia, mas ainda vamos

confiar nos "Canais de Sorte" para a parte "instantânea" do jogo - os custos do gás seriam proibitivos para usar esses RNGs com mais freqüência do que no início e no final da criação do canal .

Estamos entusiasmados com o ritmo de desenvolvimento da infra-estrutura Ethereum e estamos avaliando continuamente novos recursos e pesquisas que poderiam melhorar ainda mais nossa oferta.

11 Mais Informações

Para mais informações técnicas, visite nosso website https://funfair.io ou nos envie um email no info@funfair.io