PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES
GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

DESAFIOS NA WEB3: RISCOS E VULNERABILIDADES

GUSTAVO MIRANDA DE SOUZA

GOIÂNIA
2023

GUSTAVO MIRANDA DE SOUZA

DESAFIOS NA WEB3: RISCOS E VULNERABILIDADES

Monografia apresentada ao curso de Ciência da Computação da PUC Goiás, como requisito para conclusão do curso em Ciência da computação.

Orientador: Prof. Gustavo Siqueira Vinhal.

GOIÂNIA
2023

GUSTAVO MIRANDA DE SOUZA

DESAFIOS NA WEB3: RISCOS E VULNERABILIDADES

Este Trabalho de Conclusão de Curso julgado adequado para obtenção do título de Bacharel em Ciência da Computação, e aprovado em sua forma final pela Escola de Politécnica e de Artes, da Pontifícia Universidade Católica de Goiás, em _____/_____/_____.

_____
Prof.ª Ma. Ludmilla Reis Pinheiro dos Santos
Coordenadora de Trabalho de Conclusão de Curso

Banca Examinadora:

_____
Orientador: Prof. Me. Gustavo Siqueira Vinhal

_____
Prof.ª. Me. Ana Flávia M. de L. Garrote

_____
Prof. Me. Rafael Leal Martins

GOIÂNIA
2023

DEDICATÓRIA

Dedico este trabalho a todos que fizeram parte da minha jornada acadêmica. Aos meus pais que me apoiaram a todo momento e tornaram este caminho possível, a minha namorada que esteve ao meu lado nos momentos difíceis, aos meus amigos que dividiram essa jornada comigo e aos professores que compartilharam seu conhecimento e sabedoria para meu aprendizado.

RESUMO

Este trabalho investiga os riscos e as vulnerabilidades presentes na Web3, a próxima geração da *World Wide Web* baseada em tecnologias descentralizadas como *blockchain* e contratos inteligentes. O objetivo deste estudo é identificar e analisar as principais vulnerabilidades que podem comprometer a segurança e a privacidade dos desenvolvedores e usuários nesse ambiente emergente. A pesquisa foi conduzida através de uma revisão sistemática da literatura, que permitiu a identificação e a análise crítica de estudos, relatórios e projetos relevantes. As principais vulnerabilidades encontradas foram identificadas na área de desenvolvimento dos contratos inteligentes, nos riscos de se interagir com um contrato inteligente inseguro e na falta de cuidado no armazenamento de informações sensíveis. Os resultados deste estudo contribuem para uma melhor compreensão das vulnerabilidades presentes na Web3 e fornecem *insights* valiosos para desenvolvedores, pesquisadores e usuários interessados nesse ambiente. Ao abordar essas vulnerabilidades de forma proativa, podemos promover a adoção segura e sustentável da Web3, garantindo a proteção dos dados e a confiança dos usuários.

Palavras-chave: Web3, Segurança na Web3, Vulnerabilidades na Web3, Blockchain, Contratos inteligentes.

ABSTRACT

This research investigates the risks and vulnerabilities present in Web3, the next generation of the World Wide Web based on decentralized technologies such as blockchain and smart contracts. The objective of this study is to identify and analyze the main vulnerabilities that may compromise the security and privacy of developers and users in this emerging environment. The research was conducted through a systematic literature review, allowing for the identification and critical analysis of relevant studies, reports, and projects. The main vulnerabilities found were identified in the area of smart contract development, in the risks of interacting with an insecure smart contract, and in the lack of care in storing sensitive information. The results of this study contribute to a better understanding of the vulnerabilities present in Web3 and provide valuable insights for developers, researchers, and users interested in this environment. By proactively addressing these vulnerabilities, we can promote the safe and sustainable adoption of Web3, ensuring data protection and user trust.

Keywords: Web3, Web3 Security, Web3 Vulnerabilities, Blockchain, Smart Contracts.

LISTA DE FIGURAS

LISTA DE SIGLAS

WEB – *World Wide Web*

WEB3 – *World Wide Web Generation 3*

WEB2 – *World Wide Web Generation 2*

DAPP – *Decentralized Application*

NFT – *Non-Fungible Token*

ETH – Ether

UINT – *Unsigned Integer*

DEFI – *Decentralized Finance*

DEX – *Decentralized Exchange*

DOS – *Denial of Service*

SUMÁRIO

1 INTRODUÇÃO

A Web3 representa uma evolução significativa em relação à Web2, que é a internet tradicional que usamos diariamente. Enquanto a Web2 é caracterizada por uma centralização de dados e controle por grandes empresas, a Web3 busca criar um ambiente mais descentralizado e democrático na internet (STACKPOLE, 2023).

Na Web2, nossos dados pessoais são frequentemente armazenados centralmente por empresas, e as interações na web são mediadas por essas mesmas empresas. Por outro lado, a Web3 adota tecnologias como *blockchain*, contratos inteligentes e criptomoedas para permitir que os usuários tenham mais controle sobre seus próprios dados e interações na web. Ela elimina intermediários, possibilitando a criação de aplicativos descentralizados (dApps). Na Web3, a confiança é alcançada por meio de mecanismos descentralizados de consenso e criptografia, em vez de confiar em empresas centralizadas (STACKPOLE, 2023).

A Web3 é altamente relevante devido à sua capacidade de resolver deficiências significativas da Web2. Ela propõe um ambiente descentralizado que dá aos usuários maior controle sobre seus dados e interações online, reduzindo a dependência de intermediários centralizados. Isso resulta em maior segurança e privacidade dos dados dos usuários, além de abrir portas para a criação de aplicativos inovadores baseados em tecnologias como *blockchain* e contratos inteligentes. Além disso, a Web3 viabiliza a posse direta de ativos digitais, como criptomoedas e NFTs, promove a inclusão financeira e aumenta a transparência e confiança nas transações online. Ela também é resistente à censura, o que é vital em regiões onde a liberdade de expressão é limitada (STACKPOLE, 2023).

A Web3, embora promissora, enfrenta uma série de desafios e riscos. A segurança é uma preocupação significativa, especialmente em relação aos contratos inteligentes, que podem conter vulnerabilidades, resultando em grandes perdas financeiras. Além disso os usuários da Web3 precisam estar cientes sobre a responsabilidade de proteger sua senha que é conhecida como chave privada dentro da *blockchain*, pois todos os ativos estarão atrelados a essa chave, que se perdida ou roubada não existe uma forma de recuperá-la (BRAVE, 2023).

A relevância deste trabalho está na sua busca por identificar e mitigar os riscos e vulnerabilidades da Web3, com foco especial nos contratos inteligentes. Ao abordar essas questões, a pesquisa contribui para uma adoção segura e sustentável da

tecnologia, assegurando a proteção dos dados e fortalecendo a confiança dos usuários. A compreensão e prevenção efetiva dessas ameaças são cruciais para o desenvolvimento saudável do ecossistema Web3.

O objetivo geral deste trabalho é demonstrar os riscos e as vulnerabilidades presentes na web3, citando alguns ataques mais comuns a contratos inteligentes e aos problemas corriqueiros enfrentados por um usuário que consome conteúdo nessa nova Web.

O objetivo específico deste trabalho é examinar os ataques de reentrância, *overflow e underflow* em contratos inteligentes na web3, identificar as vulnerabilidades que propiciam esses ataques, e sugerir medidas de segurança para prevenir e reduzir os riscos desses ataques ocorrerem. Adicionalmente, será realizada uma análise sobre como um roubo de chave privada e um ataque de *front running* podem causar prejuízos financeiros a um usuário da web3.

Espera-se que os resultados deste trabalho possam contribuir para que os usuários da Web3 possam compreender quais são os riscos e como acontecem os ataques dentro desse mundo, e como pode se prevenir para garantir a sua segurança.

Esta monografia está estruturada da seguinte forma: neste Capítulo é apresentado o contexto do trabalho, o objetivo e resultados esperados. O Capítulo 2 traz o referencial teórico com conceitos e definições. No Capítulo 3 é descrito o método, mostrando como o trabalho foi desenvolvido e o que foi feito para atingir o objetivo geral. No Capítulo 4 é apresentado alguns dos ataques mais famosos a contratos inteligentes e aos usuários finais de forma direta, identificando as vulnerabilidades que permitem que o ataque aconteça. Finalmente o Capítulo 5 é apresentado a análise dos resultados obtidos, traz as considerações finais do TCC e sugestões para trabalhos futuros.

2 REFERENCIAL TEÓRICO

Este capítulo traz conceitos e definições necessárias, para a compreensão do trabalho.

2.1 World Wide Web 3.0

A ideia da Web 3.0 foi criada pelo cofundador do Ethereum, Gavin Wood, logo após o lançamento do Ethereum em 2014. Essa ideia é considerada uma evolução da Internet que busca transformar a forma como interagimos, compartilhamos informações e realizamos transações online. Enquanto a Web2 se caracterizou pela centralização de dados e poder em plataformas controladas por grandes empresas, a Web3 propõe um ambiente mais descentralizado e empoderado para os usuários (ETHEREUM.ORG, 2023).

Na Web3, a descentralização é essencial, pois as infraestruturas e aplicativos não contam com servidores centralizados, mas sim com tecnologias como *blockchain* e sistemas distribuídos. Isso traz benefícios como segurança, resistência a falhas e transparência (ETHEREUM.ORG, 2023).

A criptografia é muito importante na Web3 para manter a privacidade e a segurança. Ela protege as comunicações, verifica a identidade dos usuários e mantém os dados em segurança. Também controla o acesso a ativos digitais e protege as carteiras digitais usando chaves criptográficas (ETHEREUM.ORG, 2023).

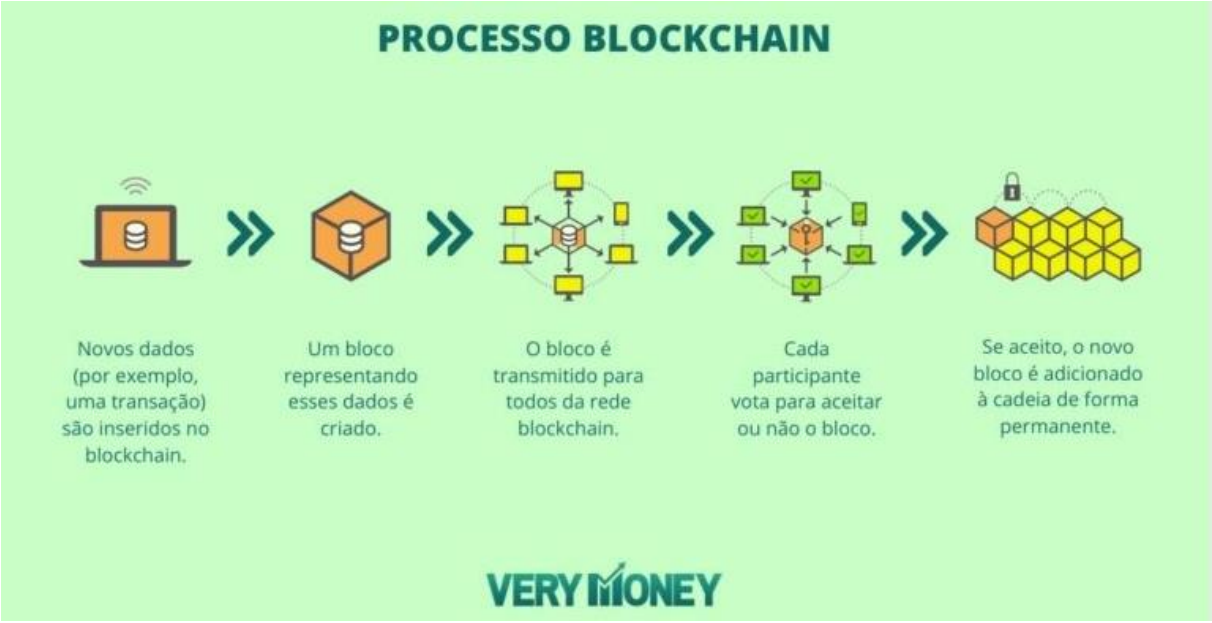
2.2 Blockchain

Uma *blockchain* é um banco de dados distribuído ou um livro-razão de registros compartilhados entre os nós de uma rede de computadores específica. Atualmente, essa tecnologia é mais reconhecida por seu papel essencial em sistemas de criptomoedas, onde seu objetivo é garantir a segurança e descentralização dos registros de transações. No entanto, o uso dessa tecnologia não se limita apenas às criptomoedas, pois uma *blockchain* pode ser aplicada em qualquer setor onde haja a necessidade de tornar dados imutáveis, ou seja, informações que nunca poderão ser alteradas (HAYES, 2023).

O sistema *blockchain* funciona da seguinte forma, o sistema registra regularmente dados transacionais em agrupamentos designados como blocos. Cada bloco é atribuído uma identificação única conhecida como *hash*, representando um código matemático exclusivo. Esses blocos são entrelaçados sequencialmente, formando uma cadeia contínua de blocos, daí o termo "*chain*", organizado de acordo com a cronologia das transações. Se houver tentativa de alterar um bloco anterior, este não é modificado; no entanto, é possível enviar uma nova transação para análise e inclusão em um novo bloco de dados (EXAME, 2021).

Na Figura 1 estão ilustradas as principais etapas do processo de inserção de um dado dentro de uma *Blockchain*. O processo inicia quando se tem um novo dado que quer ser inserido dentro da *blockchain*. Esse dado que é representado por um bloco é transmitido para todos os nós dentro de uma rede, após isso, esses nós entram em consenso para decidirem sobre a aceitação do novo bloco. E finalmente se eles entrarem em consenso e o bloco for aceito, ele é adicionado a cadeia de forma permanente.

Figura 1 – Processo *Blockchain*



Fonte: VERY MONEY (2022)

2.3 Criptomoedas

As criptomoedas são moedas digitais ou virtuais protegidas por criptografia, o que as torna extremamente seguras e quase impossível de serem falsificadas ou fraudadas. Além disso, um aspecto essencial delas é a sua natureza descentralizada, frequentemente fundamentada na tecnologia *blockchain* (FRANKENFIELD, 2023).

Uma característica distintiva das criptomoedas é que geralmente não são emitidas por uma autoridade central, o que as torna teoricamente imunes a interferências ou manipulações governamentais. Essa autonomia das criptomoedas é possível devido aos algoritmos criptográficos complexos que garantem a segurança das transações e a validação das informações por meio de consenso distribuído na rede (FRANKENFIELD, 2023).

Ao contrário das moedas tradicionais emitidas pelo governo, as criptomoedas operam em uma rede descentralizada, onde as transações são registradas de forma transparente e permanente em um livro-razão público. Essa limpidez e a ausência de intermediários garantem maior segurança e privacidade nas transações, além de reduzir os custos e o tempo envolvidos em transferências de dinheiro (FRANKENFIELD, 2023).

2.3.1 Bitcoin

O Bitcoin foi a primeira criptomoeda do mundo, criada em 2009 por uma pessoa ou grupo que usou o pseudônimo Satoshi Nakamoto. Ele introduziu um conceito inovador de moeda digital descentralizada, baseada no sistema *blockchain* (INFOMONEY, 2022).

Uma das características fundamentais do Bitcoin é o seu fornecimento limitado. Existem apenas 21 milhões deles que podem ser minerados, o que significa que a inflação é controlada e previsível. Eles são criados por meio de um processo chamado mineração, no qual os mineradores resolvem problemas matemáticos complexos para adicionar novos blocos ao *blockchain* e receber recompensas em Bitcoin (INFOMONEY, 2022).

A resolução de um problema matemático é chamada de prova de trabalho. A dificuldade dessa tarefa é ajustada periodicamente para manter constante a taxa de criação de novos blocos, garantindo assim um processo equilibrado. A competição

entre os mineradores é um elemento central. Eles concorrem para serem os primeiros a resolver o problema de prova de trabalho e, assim, validar o bloco. Essa competição cria um ambiente onde o consenso na rede é alcançado por meio da resolução desses desafios computacionais (Nakamoto, 2009).

Essa criptomoeda também oferece segurança e privacidade aos usuários por meio de criptografia avançada. Cada transação é protegida por chaves criptográficas públicas e privadas, garantindo a autenticidade e a integridade das transações. Além disso, as carteiras de Bitcoin permitem aos usuários manter o controle total de seus fundos, tornando-se responsáveis pela segurança de suas chaves privadas (INFOMONEY, 2022).

2.3.2 Ethereum

O Ethereum é uma criptomoeda e uma plataforma descentralizada baseada em *blockchain* que permite a criação e execução de contratos inteligentes e aplicativos descentralizados (DApps). Foi proposto em 2013 pelo programador Vitalik Buterin e lançado em 2015 (SMITH, 2023).

Ao contrário do Bitcoin, que se concentra principalmente em ser uma moeda digital, o Ethereum é uma plataforma mais abrangente, projetada para possibilitar a criação de aplicativos descentralizados com funcionalidades mais complexas. Ele opera em sua própria criptomoeda chamada Ether (ETH), que é usada para pagar pelos serviços e transações na rede Ethereum (SMITH, 2023).

O Ethereum introduziu a ideia de contratos inteligentes, que são programas autônomos que executam automaticamente acordos estabelecidos entre as partes. Esses contratos inteligentes são escritos em uma linguagem de programação específica do Ethereum, chamada Solidity. Eles são armazenados e executados na *blockchain* do Ethereum, garantindo a transparência, imutabilidade e segurança das transações (SMITH, 2023).

2.3.2.1 Ether

O Ether é uma criptomoeda usada na rede Ethereum, ela é usada como moeda para transações na plataforma e é usada como “combustível” para executar ações e funções de contratos inteligentes dentro da rede (FRANKENFIELD, 2023).

2.3.2.2 Transação

Uma transação é um registro de envio de criptomoedas, ativação de contratos inteligente ou uma interação dentro da rede Ethereum. Cada transação é registrada na *blockchain*, tornando-se imutável e transparente para verificação. Isso permite que a rede Ethereum funcione como um sistema de registro seguro, rastreando a transferência de ativos e a execução de contratos de forma confiável e honesta (CHEN, 2023).

2.3.2.3 Solidity

Solidity é uma linguagem de programação projetada especificamente para o desenvolvimento de contratos inteligentes em *blockchains* baseadas em Ethereum e em outras plataformas compatíveis com contratos inteligentes (EBUN-AMU, 2023).

Ela é uma linguagem de alto nível, que significa que é mais próxima da linguagem humana do que das linguagens de baixo nível, como a linguagem de montagem. Isso facilita a escrita de contratos inteligentes, mesmo para desenvolvedores que não são especialistas em criptografia ou *blockchain* (EBUN-AMU, 2023).

A linguagem Solidity suporta a orientação a objetos e a programação orientada a contratos, permitindo a criação de classes, herança e a definição de funções que são invocadas em resposta a eventos ou chamadas externas. Ela também fornece uma estrutura para o controle de propriedades, acesso a variáveis de estado, gestão de exceções e outros recursos típicos de linguagens de alto nível (EBUN-AMU, 2023).

Além disso, ela oferece suporte para tipos de dados específicos da *blockchain*, como endereços Ethereum, uint (inteiros sem sinal) e bytes (sequências de bytes). A linguagem também inclui recursos de segurança e testabilidade, como a capacidade

de realizar auditorias de contratos inteligentes para identificar vulnerabilidades e problemas de segurança antes de implantá-los na rede principal (EBUN-AMU, 2023).

2.4 Contratos inteligentes

Um contrato inteligente é um programa inviolável que é executado em uma rede *blockchain* quando certas condições predefinidas são atendidas (CHAINLINK, 2023).

Os contratos inteligentes são formados por um código que estabelece condições específicas que, quando cumpridas, ativam resultados predeterminados. A sua execução ocorre em uma rede *blockchain* descentralizada, em vez de um servidor centralizado, possibilitando que várias partes alcancem um resultado compartilhado de forma aberta, rápida e inalterável (CHAINLINK, 2023).

Esses contratos inteligentes são uma ferramenta poderosa para automatização, uma vez que não dependem de um administrador central e, portanto, não são vulneráveis a pontos únicos de falha. Quando utilizados em acordos digitais envolvendo várias partes, os contratos inteligentes reduzem o risco para as partes envolvidas, aumentam a eficiência, diminuem os custos e proporcionam maior transparência nos processos (CHAINLINK, 2023).

2.5 Aplicativos Descentralizados

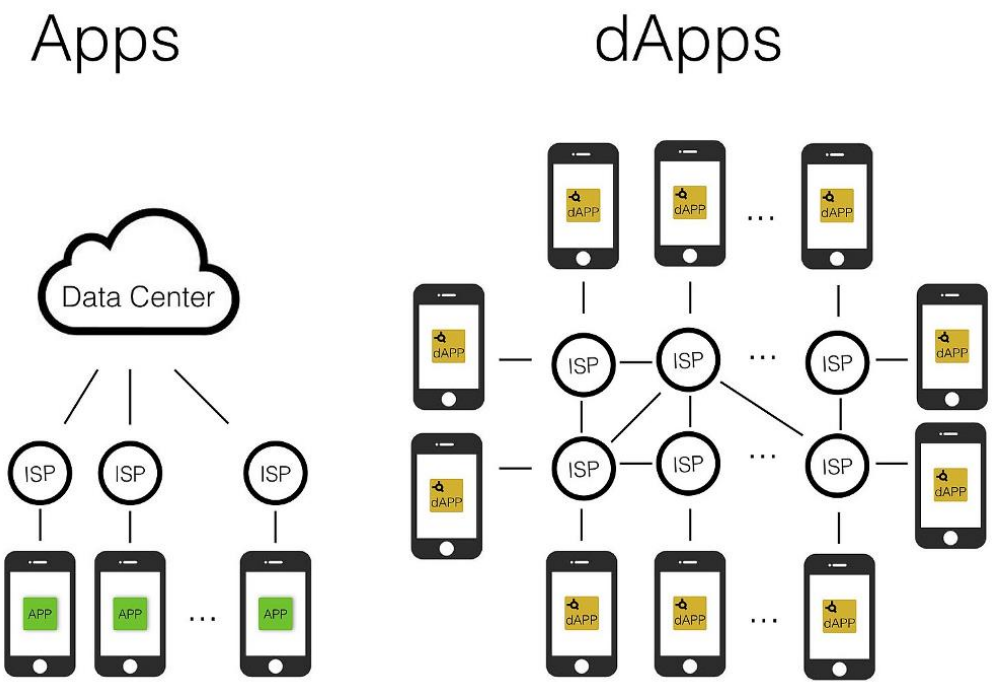
DApps, abreviação de "aplicativos descentralizados" (em inglês, *decentralized applications*), são aplicativos de software que operam na *blockchain* ou em uma rede de criptomoeda descentralizada. Ao contrário dos aplicativos tradicionais, que são executados em servidores centralizados controlados por empresas ou entidades, os dApps funcionam em uma rede distribuída de nós, onde cada nó possui uma cópia do registro de transações (o *blockchain*) (GEORGE, 2022).

Esses aplicativos descentralizados aproveitam a tecnologia *blockchain* para garantir a segurança, clareza e invariabilidade das operações. As interações e transações dentro dos dApps são registradas na *blockchain*, tornando-as públicas e acessíveis a todos os participantes da rede, o que aumenta a confiança e a verificabilidade das informações (GEORGE, 2022).

Os dApps podem ter diversas finalidades, desde soluções financeiras, jogos, plataformas de gerenciamento de dados, sistemas de votação, até aplicativos de governança e muito mais. Além disso, muitos dApps utilizam contratos inteligentes para automatizar as operações e garantir a execução das regras definidas de forma transparente e imparcial (GEORGE, 2022).

A Figura 2 apresenta a diferença estrutural entre os aplicativos tradicionais e os aplicativos descentralizados. Essa figura consegue demonstrar de maneira clara a diferença estrutural entre esses aplicativos. Enquanto no app a requisição é feita a um servidor central, que na imagem está sendo representado por um *data center*, o dApp funciona em um sistema distribuído de nós, onde cada nó possui uma cópia das informações, e fornece essas informações a outros nós quando necessário.

Figura 2 – Apps vs dApps



Fonte: BERTOLUCCI (2023)

2.5.1 Decentralized Finance

DeFi, ou Finanças Descentralizadas, é um termo que se refere a um conjunto de serviços financeiros e aplicativos construídos em redes de *blockchain*, como o Ethereum. A principal característica do DeFi é sua natureza descentralizada, o que

significa que não depende de intermediários, como bancos ou instituições financeiras tradicionais, para facilitar transações e serviços financeiros (SHARMA, 2021).

No DeFi, as transações e os contratos inteligentes são executados automaticamente por meio de código de computador, eliminando a necessidade de intermediários humanos. Isso proporciona maior eficiência e reduz custos. Além disso, o DeFi é acessível a qualquer pessoa com acesso à internet, tornando os serviços financeiros mais inclusivos (SHARMA, 2021).

Os principais casos de uso do DeFi incluem empréstimos, empréstimos agrupados, troca de ativos digitais. Por exemplo, os usuários podem emprestar seus ativos digitais em troca de juros e trocar seus ativos por outros através de DEXs (SHARMA, 2021).

2.5.1.1 Decentralized Exchanges

Uma DEX, ou *Decentralized Exchange* (Troca Descentralizada), é uma plataforma de negociação de ativos digitais que opera de forma descentralizada, sem a necessidade de intermediários ou uma autoridade central. Diferentemente das *exchanges* tradicionais, as DEXs funcionam diretamente em *blockchains* ou redes descentralizadas, permitindo que os usuários negociem ativos digitais de forma ponto a ponto, ou seja, diretamente entre si (BENSON, 2021).

A principal característica de uma DEX é a eliminação da necessidade de confiar em uma entidade intermediária para armazenar ou facilitar as transações. Em vez disso, as DEXs usam contratos inteligentes e protocolos *blockchain* para permitir que os usuários realizem negociações de forma autônoma, mantendo o controle total sobre seus ativos (BENSON, 2021).

As DEXs oferecem diversas vantagens, como maior segurança, transparência e menor risco de *hacks*, uma vez que os ativos são mantidos pelos próprios usuários em suas carteiras. Além disso, elas frequentemente têm taxas de transação mais baixas do que as *exchanges* centralizadas (BENSON, 2021).

Essas trocas descentralizadas são amplamente utilizadas no contexto do ecossistema DeFi (Finanças Descentralizadas), onde desempenham um papel fundamental na facilitação de negociações de *tokens*, participação em fundos de investimento compartilhado e outras operações financeiras descentralizadas. As

DEXs também são fundamentais para a criação e o funcionamento de mercados de *tokens* não fungíveis (NFTs) e permitem que os usuários negociem uma variedade de ativos digitais (BENSON, 2021).

No entanto, as DEXs também apresentam desafios, como a necessidade de lidar com problemas de liquidez e escalabilidade, bem como a complexidade de interface para usuários menos experientes em tecnologia *blockchain*. Mesmo assim, elas desempenham um papel vital na evolução das finanças e da tecnologia *blockchain*, proporcionando alternativas viáveis e descentralizadas às *exchanges* tradicionais (BENSON, 2021).

2.5.2 Jogos Blockchain

Jogos de *blockchain*, ou jogos baseados em *blockchain*, são uma categoria de jogos de vídeo que aproveitam a tecnologia de *blockchain* para aprimorar vários aspectos da jogabilidade, propriedade e ativos in-game. Esses jogos fazem parte do movimento mais amplo na indústria de jogos para incorporar *blockchain* e criptomoedas nas experiências de jogo (CHAINLINK, 2023).

A característica distintiva dos jogos de *blockchain* é a integração com redes *blockchain*, que oferece vários benefícios únicos. Esses jogos frequentemente usam *tokens* não fungíveis (NFTs) para representar ativos *in-game*, personagens e itens, permitindo que os jogadores tenham verdadeira propriedade desses itens digitais. Os NFTs são registrados na *blockchain*, garantindo sua escassez e procedência (CHAINLINK, 2023).

Os jogos de *blockchain* permitem que os jogadores comprem, vendam e negociem ativos *in-game* com outros jogadores, tanto dentro como fora do ecossistema do jogo. Isso promove uma economia impulsionada pelos jogadores, onde o valor dos ativos *in-game* é determinado pela oferta e demanda. Os jogadores também podem usar esses ativos em diferentes jogos que suportam os mesmos padrões de *blockchain*, o que adiciona interoperabilidade e valor aos seus ativos (CHAINLINK, 2023).

Além disso, a tecnologia *blockchain* garante transparência e segurança na propriedade de ativos. Registros de propriedade, histórico de transações e escassez são imutáveis e verificáveis na *blockchain*. Isso significa que os jogadores têm controle

total sobre seus ativos e podem verificar sua autenticidade de forma confiável (CHAINLINK, 2023).

2.5.2.1 Tokens Não Fungíveis

NFTs, ou *Tokens* Não Fungíveis, são ativos digitais únicos que representam a propriedade de itens ou conteúdo digital, registrados em uma *blockchain*. A característica fundamental que os diferencia de outros tipos de criptomoedas, como o Bitcoin, Ethereum ou *tokens* fungíveis, é a sua singularidade.

Cada NFT é exclusivo e tem um valor distintivo, o que significa que não pode ser substituído por outra unidade de igual valor, daí o termo "não fungível" (SHARMA, 2023).

Esses ativos digitais são usados para representar uma ampla variedade de itens, como obras de arte digitais, vídeos, música, itens de jogos, colecionáveis virtuais e até mesmo propriedades virtuais. Quando um criador emite um NFT para um item, ele é registrado na *blockchain*, que atua como um registro público e imutável, confirmando a autenticidade, propriedade e histórico de propriedade desse item (SHARMA, 2023).

A propriedade de NFTs é garantida pela tecnologia *blockchain*, o que significa que os proprietários têm controle absoluto sobre seus ativos digitais. Eles podem comprar, vender ou trocar NFTs em mercados especializados, criando um mercado em crescimento para itens digitais exclusivos. Além disso, NFTs permitem que os criadores sejam recompensados de forma justa por seu trabalho, já que podem receber *royalties* sempre que seu NFT for revendido (SHARMA, 2023).

2.5.2.2 Tokens Fungíveis

Tokens são unidades digitais representativas de valor, frequentemente emitidas em redes *blockchain* ou protocolos descentralizados. Eles desempenham uma variedade de funções, podendo servir como ativos digitais, instrumentos de utilidade ou *tokens* de governança, dependendo da plataforma ou protocolo em que são criados (IREDALE, 2023).

Os *tokens* são criados com base em contratos inteligentes, que definem suas regras e funcionalidades específicas. Esses contratos são imutáveis e garantem a autenticidade e a segurança dos *tokens*, bem como a execução de ações programadas, como transferências ou votações (IREDALE, 2023).

Alguns *tokens* são projetados para serem usados como meio de troca, armazenamento de valor e transações. Outros representam ativos digitais, como títulos, propriedades virtuais em jogos ou ativos colecionáveis. Há também *tokens* de utilidade que concedem acesso a serviços ou funcionalidades específicas dentro de aplicativos descentralizados. Além disso, *tokens* de governança permitem que seus detentores participem na tomada de decisões relacionadas ao desenvolvimento ou governança de uma plataforma descentralizada (IREDALE, 2023).

2.6 Carteira digital

Uma carteira digital também conhecida de carteira *blockchain* é uma ferramenta essencial para quem usa criptomoedas e ativos digitais em uma *blockchain*. Ela é importante para armazenar com segurança chaves privadas usadas para controlar esses ativos (FRANKENFIELD, 2022).

As carteiras *blockchain* podem ser encontradas em diferentes tipos. Há carteiras de hardware, que são dispositivos físicos que são projetados para armazenar chaves privadas offline, proporcionando um nível muito alto de segurança. Também existem as carteiras de *software*, que são aplicativos em dispositivos computadores e *smartphones*. As carteiras móveis são apps para celulares que facilitam a gestão de transações de criptomoedas. E tem carteiras web, que são acessíveis através de um navegador da web e podem ser usadas em qualquer dispositivo conectado à internet (FRANKENFIELD, 2022).

Cada tipo de carteira tem suas próprias características de segurança e usabilidade. As carteiras de hardware são conhecidas por serem as mais seguras, pois mantêm as chaves privadas offline e fora do alcance de hackers, mas ela é menos prática. Já as carteiras móveis e de *software* são mais fáceis de usar, mas precisam de proteção extra, como senhas fortes e autenticação em duas etapas (FRANKENFIELD, 2022).

2.6.1 Metamask

MetaMask é uma das carteiras de criptomoedas e extensões de navegador mais populares para interagir com aplicativos descentralizados (DApps) baseados em *blockchain*, como os encontrados no ecossistema Ethereum. Ela atua como uma ponte entre os usuários e a *blockchain*, permitindo que as pessoas armazenem, gerenciem e transacionem com criptomoedas e *tokens* diretamente de seus navegadores da web (PHILLIPS, 2022).

A principal característica do MetaMask é a sua interface de usuário amigável, que simplifica a interação com DApps e a gestão de ativos digitais. Os usuários podem instalar o MetaMask como uma extensão de navegador em navegadores populares, como Google Chrome e Mozilla Firefox (PHILLIPS, 2022).

O MetaMask oferece funcionalidades como carteira de criptomoedas, integração com DApps, chaves privadas e frase de recuperação, suporte a múltiplas redes *blockchain*, segurança e gerenciamento de ativos digitais. É uma ferramenta essencial para quem deseja participar ativamente do ecossistema de *blockchain* e DeFi, facilitando a conexão com uma variedade de DApps e o gerenciamento de ativos digitais diretamente do navegador. É amplamente utilizado por desenvolvedores, *traders* e entusiastas de criptomoedas em todo o mundo (PHILLIPS, 2022).

2.6.2 Chave pública

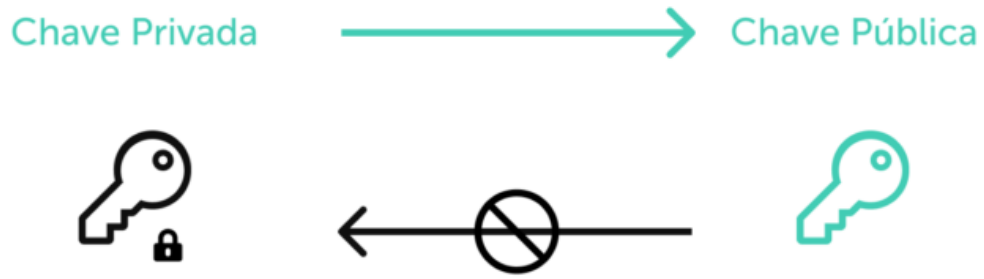
A chave pública é uma sequência longa de caracteres alfanuméricos gerada a partir da chave privada correspondente. Ela é usada para criptografar informações que somente podem ser descriptografadas pela chave privada correspondente. Essa relação matemática entre a chave pública e a chave privada é o que torna possível a criptografia de chave pública (ABROL, 2022).

A chave pública pode ser compartilhada livremente com qualquer pessoa, e é frequentemente usada para fins de verificação e identificação. Por exemplo, quando alguém deseja enviar criptomoedas para outra pessoa, ele pode usar a chave pública do destinatário para garantir que a transação seja direcionada à pessoa certa. Além disso, a chave pública pode ser usada para verificar a autenticidade de mensagens digitais assinadas pela chave privada correspondente, fornecendo um meio de verificação de autenticidade em ambientes online (ABROL, 2022).

Uma característica importante da criptografia de chave pública é que, embora seja fácil gerar uma chave pública a partir de uma chave privada, é matematicamente computacionalmente inviável derivar a chave privada a partir da chave pública. Isso garante que as informações criptografadas com a chave pública possam ser protegidas com segurança, pois somente o detentor da chave privada correspondente será capaz de descriptografá-las (ABROL, 2022).

De acordo com a figura 3 apresentada é possível observar que é possível chegar em uma chave pública através da privada, mas é impossível chegar na privada a partir da pública.

Figura 3 – Criptografia entre chave pública e privada



Fonte: MORELAND (2022)

2.6.3 Chave privada

Uma chave privada é uma sequência de caracteres alfanuméricos que serve como segredo criptográfico. A chave privada deve ser mantida estritamente em segredo e nunca deve ser compartilhada com ninguém, pois é a chave para descriptografar informações que foram criptografadas usando a chave pública correspondente (ABROL, 2022).

A chave privada é frequentemente usada para assinar digitalmente informações, como transações em *blockchain* ou mensagens digitais. Essa assinatura digital é gerada usando a chave privada e pode ser verificada por qualquer pessoa que tenha acesso à chave pública correspondente. Isso fornece um método confiável de verificação de autenticidade e integridade de informações online. (ABROL, 2022)

Manter a chave privada em segurança é de extrema importância, pois qualquer pessoa que tenha acesso a ela pode controlar os ativos e as informações associadas a essa chave. Isso é particularmente relevante em criptomoedas, onde a posse da chave privada é essencial para acessar e gastar os ativos digitais associados a uma carteira (ABROL, 2022).

3 PROCEDIMENTOS METODOLÓGICOS

Esta pesquisa, segundo sua natureza é um resumo de assunto, na qual busca esclarecer a esfera de conhecimento do projeto, com a finalidade de examinar e relacionar as áreas de conhecimento envolvidas, promovendo assim, uma compreensão mais profunda de suas causas e explicações (WAZLAWICK, 2014).

Segundo os objetivos é uma pesquisa exploratória e descritiva. A descritiva, busca dados mais consistentes sobre o assunto, porém, não ocorre a interferência do pesquisador, apenas expõe os fatos de como eles são (WAZLAWICK, 2014).

A pesquisa exploratória geralmente é considerada como a primeira parte do processo de pesquisa, pois o autor não necessariamente tem um objetivo ou uma hipótese definida (WAZLAWICK, 2014). Essa pesquisa tem como objetivo a maior familiaridade do autor com o problema, facilitando assim a criação de novas hipóteses. Muitas das vezes é considerada uma pesquisa flexível, pois considera os variados aspectos referentes aos fenômenos ou fatos estudados (GIL, 2017).

Quanto aos procedimentos técnicos, será uma pesquisa bibliográfica e experimental. Uma pesquisa bibliográfica exige o estudo de artigos, teses, livros, entre outros. A pesquisa experimental é caracterizada por ter variáveis experimentais que podem ser coordenadas pelo pesquisador (WAZLAWICK, 2014).

A pesquisa bibliográfica, consistirá na análise de materiais previamente publicados, abrangendo uma variedade de fontes, como livros, testes, materiais disponibilizados na internet, revistas e outros. Sua principal vantagem reside na capacidade de investigar um aspecto mais abrangente de fenômenos em comparação com a pesquisa direta (GIL, 2017).

4 PRINCIPAIS PROBLEMAS DE SEGURANÇA E ATAQUES

Este capítulo apresenta seis (6) tipos de ataques que ocorrem dentro da web3, sendo que quatro (4) deles estão diretamente relacionados a vulnerabilidades presentes dentro de um contrato inteligente e outros dois (2) estão relacionados a engenharia social praticada contra usuários.

4.1 Ataques e vulnerabilidades em contratos inteligentes

As vulnerabilidades em contratos inteligentes são basicamente falhas de segurança em código que podem ser exploradas por atacantes. Essas vulnerabilidades podem levar a perdas de ativos ou violação de segurança em aplicativos descentralizados (IMMUNEBYTES, 2022).

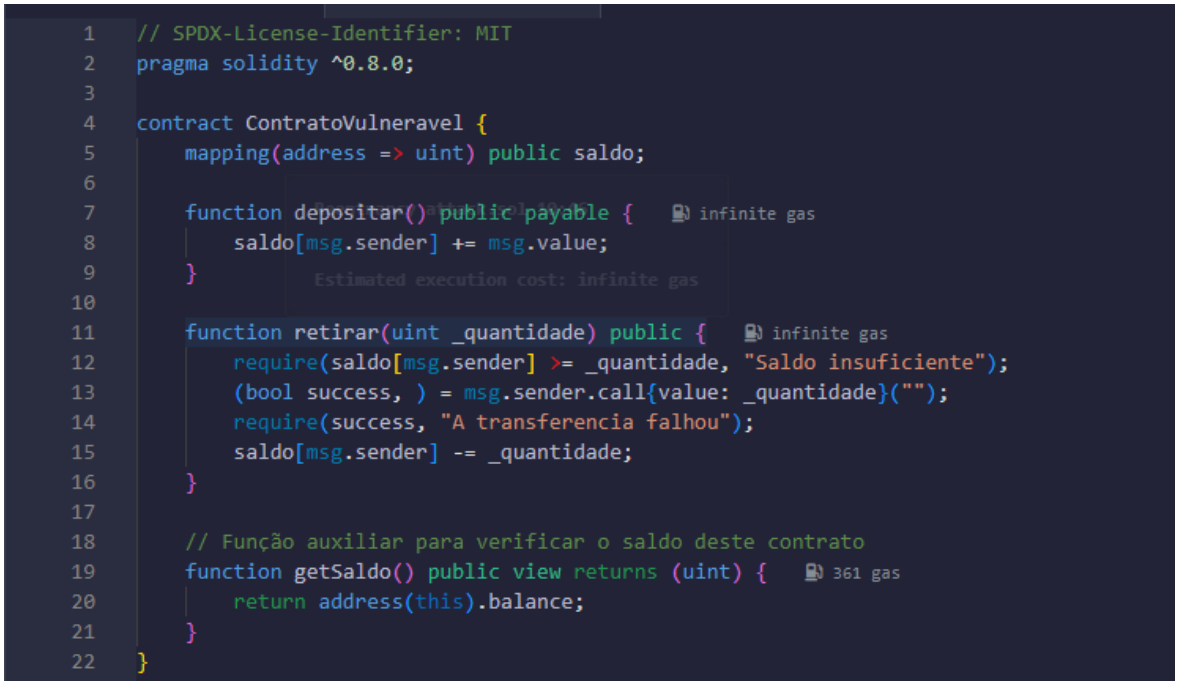
4.1.1 Reentrancy Attack

Reentrancy Attack ou ataque de reentrância ocorre quando um invasor explora uma fraqueza em um contrato inteligente para executar um código malicioso de forma inesperada. Isso normalmente acontece quando uma função ou método permite que o invasor chame outras funções ou contratos sem a devida verificação ou restrição (NELSON, 2020).

O nome “reentrância” se refere ao fato de que o atacante pode “entrar novamente” no contrato inteligente atacado enquanto a primeira chamada ainda está em andamento. Isso pode criar um ciclo de chamadas que permite o atacante desviar fundos ou recursos repetidamente antes que o contrato original possa responder ou se defender. Essa exploração pode levar a perdas financeiras, vazamento de dados sensíveis ou interrupção do funcionamento normal do sistema (HACKEN, 2023).

Um exemplo de ataque de reentrância: Considere um contrato inteligente que permite a retirada de fundos como foi apresentado pelo figura 4. Esse contrato pode conter uma função retirar que permite que um usuário retire seus fundos. No entanto, se a função retirar não verificar o saldo do usuário antes de processar a retirada, um invasor pode criar um contrato malicioso exemplificado pelo figura 5 que chama repetidamente a função retirar até que o saldo total do contrato seja esgotado.

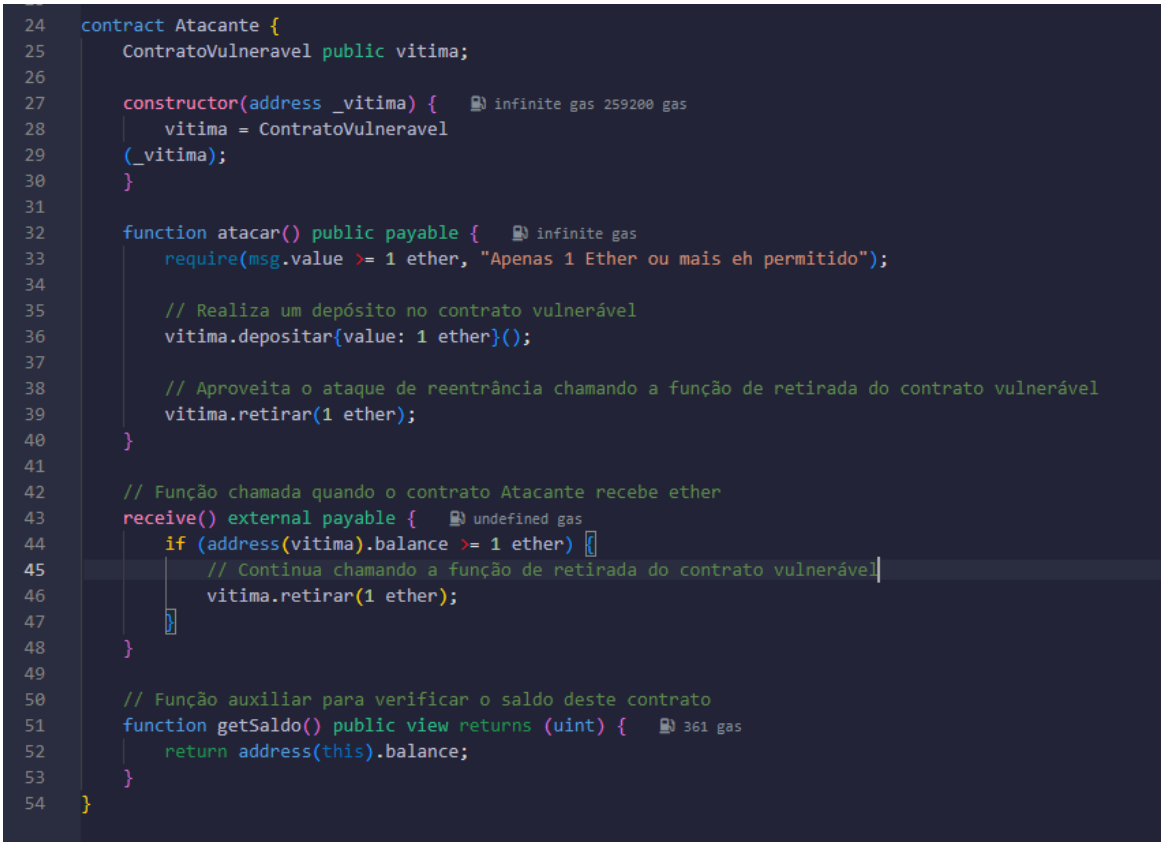
Figura 4 - Contrato vulnerável a ataques de reentrância



```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract ContratoVulneravel {
    mapping(address => uint) public saldo;

    function depositar() public payable {    infinite gas
        saldo[msg.sender] += msg.value;
    }

    function retirar(uint _quantidade) public {    infinite gas
        require(saldo[msg.sender] >= _quantidade, "Saldo insuficiente");
        (bool success, ) = msg.sender.call{value: _quantidade}("");
        require(success, "A transferencia falhou");
        saldo[msg.sender] -= _quantidade;
    }

    // Função auxiliar para verificar o saldo deste contrato
    function getSaldo() public view returns (uint) {    361 gas
        return address(this).balance;
    }
}
```

Fonte: Elaborada pelo autor (2023)

Figura 5 - Contrato atacante de contratos vulneráveis a reentrância



```
contract Atacante {
    ContratoVulneravel public vitima;

    constructor(address _vitima) {    infinite gas 259200 gas
        vitima = ContratoVulneravel
    (_vitima);
    }

    function atacar() public payable {    infinite gas
        require(msg.value >= 1 ether, "Apenas 1 Ether ou mais eh permitido");

        // Realiza um depósito no contrato vulnerável
        vitima.depositar{value: 1 ether}();

        // Aproveita o ataque de reentrância chamando a função de retirada do contrato vulnerável
        vitima.retirar(1 ether);
    }

    // Função chamada quando o contrato Atacante recebe ether
    receive() external payable {    undefined gas
        if (address(vitima).balance >= 1 ether) {
            // Continua chamando a função de retirada do contrato vulnerável
            vitima.retirar(1 ether);
        }
    }

    // Função auxiliar para verificar o saldo deste contrato
    function getSaldo() public view returns (uint) {    361 gas
        return address(this).balance;
    }
}
```

Fonte: Elaborada pelo autor (2023)

O ataque funciona da seguinte maneira:

1- O contrato Atacante realiza um depósito de 1 Ether no contrato “ContratoVulneravel”.

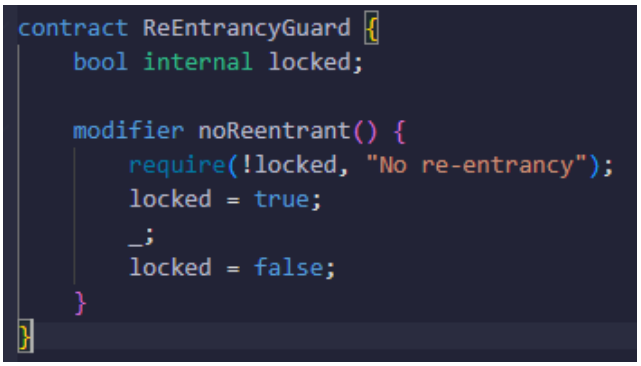
2- Em seguida, ele chama a função de retirada do contrato “ContratoVulneravel”, que tenta transferir 1 Ether de volta ao Atacante.

3- O contrato Atacante recebe a transferência e chama a função de retirada do contrato “ContratoVulneravel” repetidamente através da função *receive*.

Isso cria um ciclo em que o contrato Atacante pode retirar repetidamente fundos do contrato “ContratoVulneravel” enquanto o saldo total do contrato não estiver esgotado.

Existem diversas formas de se proteger contra ataques de reentrância. Dentre elas estão: A utilização de padrões de projetos seguros, a utilização de modificadores de estado, que vai garantir que a ordem de execução seja segura e que não seja possível interromper o processo, fazer verificação e validações, garantindo a validação de entradas e condições e estados antes de realizar operações que possam ser exploradas por um invasor. E de forma geral sempre se manter atualizado usando as melhores práticas de segurança em desenvolvimento de software e *blockchain*.

Figura 6 - Modificador de proteção a ataque de reentrância



Fonte: Elaborada pelo autor (2023)

Uma solução básica para o exemplo citado anteriormente, seria basicamente a utilização de modificadores que juntamente com uma variável booleana garantiriam que a chamada da função só fosse permitida novamente enquanto não houvesse

nenhuma chamada em andamento, fazendo com que a reentrada fosse evitada como é mostrado no figura 6.

4.1.2 Arithmetic Overflow e Underflow

Arithmetic Overflow and Underflow são vulnerabilidades resultantes da falta de tratamento em operações matemáticas. Se uma operação de adição, multiplicação ou qualquer outra operação matemática resultar em um valor que ultrapasse o valor permitido pelo tipo de dado, ocorre um *Arithmetic Overflow* (HACKEN, 2023).

Isso pode ser explorado por atacantes para criar *tokens* ilegal de maneira imprevisível, causando perdas substanciais nos contratos inteligentes (HACKEN, 2023).

Já o *arithmetic underflow* é o oposto, ele ocorre quando uma operação matemática resulta em um valor menor do que zero ou menor do que o valor mínimo permitido pelo tipo de dado. Por exemplo, se você subtrair 1 de 0 em um número inteiro sem sinal, ocorrerá um *underflow*. O *underflow* geralmente é explorado por atacantes para causar efeitos inesperados no contrato inteligente, permitindo assim a retirada excessiva de fundos ou *tokens* de maneira indevida (HACKEN, 2023).

Um exemplo de ataque de *Arithmetic Overflow* e *Underflow:* Considere um contrato vulnerável a esse tipo de ataque, em que não há tratamento nenhum, nem controle sobre o valor das variáveis como evidenciado no Figura 7.

Figura 7 - Contrato vulnerável a ataque de *Arithmetic overflow* e *underflow*

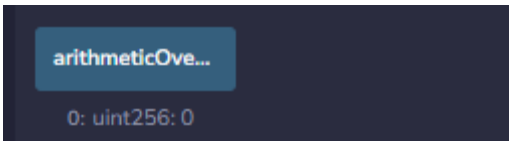
```
1  pragma solidity ^0.5.0;
2
3  contract ArithmeticVulnerabilityExample {
4      function arithmeticOverflow() public pure returns (uint) {    245 gas
5          uint maxUint = 2**256 - 1; // Maior valor possível de um uint256
6          uint result = maxUint + 1; // Isso causará um Arithmetic Overflow
7          return result;
8      }
9
10     function arithmeticUnderflow() public pure returns (uint) {    223 gas
11         uint minUint = 0; // Menor valor possível em um uint256
12         uint result = minUint - 1; // Isso causará um Arithmetic Underflow
13         return result;
14     }
15 }
```

Fonte: Elaborada pelo autor (2023)

Após essas duas funções vulneráveis serem executadas, esses serão os 2 retornos obtidos:

1. A função arithmeticOverflow soma um (1) a variável maxUint do tipo uint, que possui o valor máximo possível dois (2) elevado a duzentos e cinquenta e seis (256) menos um (1), fazendo com que haja overflow, colocando a variável no valor 0 da forma que foi apresentada no figura 8.

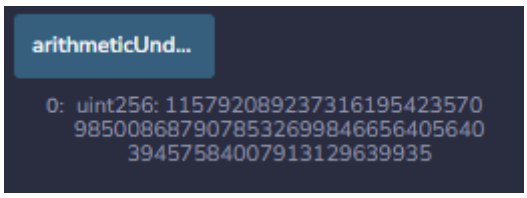
Figura 8 - Retorno da função arithmeticOverflow()



Fonte: Elaborada pelo autor (2023)

2. Já a função arithmeticUnderflow diminui um (1) da variável minUint do tipo uint, que possui o valor mínimo possível zero (0), fazendo com que haja underflow, colocando a variável no valor de dois (2) elevado a duzentos e cinquenta e seis (256) como foi ilustrado no figura 9

Figura 9 - Retorno da função arithmeticUnderflow()



Fonte: Elaborada pelo autor (2023)

Essa vulnerabilidade só está presente em contratos inteligentes que possuem a versão do solidity abaixo da 0.8, pois dela em diante essas vulnerabilidades são tratadas, retornando um erro. Porém para versões anteriores uma solução básica para resolver essas vulnerabilidades seria utilizar bibliotecas de matemática seguras, que verificam limites antes de realizar operações aritméticas. Um exemplo dessa biblioteca segura, seria a SafeMath da organização Openzepplin (COBB, 2023).

4.1.3 Denial of Service

Denial of Service (DOS) ou negação de serviço é uma categoria de ataque muito abrangente, que pode acontecer de diversas formas. Porém todos tem o mesmo objetivo, implica basicamente na intenção de tornar um contrato inteligente inativo de forma temporária ou até mesmo de forma permanente (HACKEN, 2023).

Esse ataque pode congelar os fundos contidos nesse contrato por um período indefinido ou para sempre, Além disso os ataques de DOS podem violar a lógica de um contrato inteligente (HACKEN, 2023).

4.2 Ataques e vulnerabilidades a usuários diretos

As vulnerabilidades e ataques a usuários de forma direta são basicamente problemas que qualquer usuário da web3 pode vir a ter. Os dois ataques que serão citados, são duas complicações muito comuns nesse mundo, porém o primeiro é um problema que depende totalmente do descuido do usuário e o outro é um problema ainda sem soluções definitivas.

4.2.1 Roubo de chave privada

O risco de roubo da chave privada é uma das ameaças mais significativas e críticas que um usuário na web3 pode ter.

A chave privada é a peça central que concede controle sobre todos os seus ativos digitais, como criptomoedas e *tokens*, e é usada para assinar transações e acessar informações confidenciais. Se alguém obtiver acesso à chave privada, essa pessoa terá controle total sobre os ativos e informações associadas a ela (ABROL, 2022).

O roubo dessa chave pode acontecer de diversas maneiras como: O uso de engenharia social por parte de cibercriminosos, que é um método em que eles conseguem acesso à chave por meio de aplicativos maliciosos, e-mail falsos que solicitam informações confidenciais ou até uso de sites que se passam por serviços legítimos.

4.2.2 Front Running

A vulnerabilidade Front Running é um problema específico que ocorre no ambiente de contratos inteligentes dentro da *blockchain*, onde participantes atacantes podem aproveitar informações privilegiadas para obter vantagem financeira às custas de outros usuários (PIXELPLEX, 2022).

Na Web3, as transações e contratos inteligentes são públicos e transparentes, o que significa que as informações sobre transações pendentes ou ordens de compra e venda são acessíveis a todos. Os *"Front Runners"* exploram essas informações para identificar transações de alto valor agregado que estão prestes a acontecer (DARKRELAY, 2023).

Quando um *"Front Runner"* identifica uma transação pendente que pode afetar o preço de um ativo, eles tomam ação antecipadamente, submetendo uma transação própria que se beneficia da mudança de preço que a transação original causará. Isso prejudica o usuário que iniciou a transação original, pois ele recebe um preço menos favorável (DARKRELAY, 2023).

Um exemplo comum disso na Web3 envolve contratos de finanças descentralizadas (DeFi). Se um usuário estiver prestes a comprar um *token*, um *"Front Runner"* pode antecipar essa ação e, assim, ajustar o preço do ativo antes que o usuário original execute sua transação, o que resulta em um preço menos favorável para o usuário original (DARKRELAY, 2023).

Existem diversas técnicas que estão sendo desenvolvidas para amenizar essa vulnerabilidade, como o uso de oráculos confiáveis que são basicamente sistemas que trazem dados de fora da *blockchain* para dentro, para controlar essas transações pendentes e garantir uma execução justa e segura em contratos inteligentes (DARKRELAY, 2023).

5 CONSIDERAÇÕES FINAIS

O objetivo geral deste trabalho foi o de apresentar riscos e vulnerabilidade presentes na web3, mostrar os ataques mais famosos a contratos inteligentes e riscos a usuários de forma direta, além de sugerir formas básicas de como se prevenir e garantir a segurança contra eles.

O estudo permitiu concluir que as vulnerabilidades em contratos inteligentes geralmente são resultadas de erros de programação e que são exploradas por atacantes para realizar roubos de fundos. Foi possível observar que a natureza irrevogável e autoexecutável dos contratos inteligentes os torna suscetíveis a explorações, tornando a segurança uma prioridade crítica. Além disso foi possível ver os riscos diretos que usuários da Web3 enfrentam, incluindo a perda de ativos digitais devido a falhas na segurança da carteira, exposição a práticas fraudulentas e desafios relacionados a identificação e confiança em um ambiente descentralizado.

Embora a pesquisa tenha destacado desafios e riscos, é importante reforçar que a Web3 continua sendo uma fronteira emocionante da inovação. Ela oferece oportunidades significativas de empoderamento e autonomia para os usuários, bem como promete disrupção positiva em diversos setores.

Conclui-se que os resultados deste trabalho foram satisfatórios, pois atingiram os objetivos.

Sugestão para trabalhos futuros:

- Elaborar um passo a passo de como esses e outros ataques funcionam tecnicamente dentro de uma EVM (Máquina virtual do Ethereum)
- Elaborar um trabalho que demonstre como funciona de maneira específica o processo de inserção de um dado dentro de uma blockchain.
- Elaborar um trabalho explicando o funcionamento de um MEV BOT.

REFERÊNCIAS

ABROL, Ayushi. **Private Key Vs Public Key – How They Work?**. [*S. l.*], 27 jul. 2022. Disponível em: https://www.blockchain-council.org/blockchain/private-key-vs-public-key/. Acesso em: 3 out. 2023.

BENSON, Jeff. **What Is a Decentralized Exchange (DEX)?**. [*S. l.*], 2 abr. 2021. Disponível em: https://decrypt.co/resources/what-is-decentralized-exchange-dex. Acesso em: 12 out. 2023.

BERTOLUCCI, Gustavo. **O que são DAPPs e qual sua importância?**. [*S. l.*], 5 out. 2023. Disponível em: https://livecoins.com.br/o-que-sao-dapps-e-qual-sua-importancia/. Acesso em: 5 out. 2023.

BRAVE. **Introdução à Segurança na Web3**. [*S. l.*], 22 dez. 2023. Disponível em: https://brave.com/pt-br/web3/intro-to-web3-security/. Acesso em: 26 out. 2023.

COBB, Michael. **9 smart contract vulnerabilities and how to mitigate them**. [*S. l.*], 25 maio 2023. Disponível em: https://www.techtarget.com/searchsecurity/tip/Smart-contract-vulnerabilities-and-how-to-mitigate-them. Acesso em: 23 out. 2023.

CHAINLINK. **What Are Blockchain Games?**. [*S. l.*], 21 jan. 2023. Disponível em: https://blog.chain.link/blockchain-gaming/. Acesso em: 12 out. 2023.

CHAINLINK. **What Is a Smart Contract?**. [*S. l.*], 24 maio 2023. Disponível em: https://chain.link/education/smart-contracts. Acesso em: 17 jul. 2023.

CHEN, JAMES. **Transaction**. [*S. l.*], 20 fev. 2023. Disponível em: https://www.investopedia.com/terms/t/transaction.asp. Acesso em: 28 out. 2023.

DARKRELAY. **Web3 Security Vulnerabilities: Comprehensive Guide to Protecting Digital Assets**. [*S. l.*], 1 jul. 2023. Disponível em: https://www.darkrelay.com/post/web3-security-guide. Acesso em: 26 out. 2023.

EBUN-AMU, CALVIN. **What Is Solidity and How Is It Used to Develop Smart Contracts?**. [*S. l.*], 19 abr. 2023. Disponível em: https://www.makeuseof.com/what-is-solidity/. Acesso em: 11 out. 2023.

ETHEREUM.ORG. **Introdução à Web3**. [*S. l.*], 13 jul. 2023. Disponível em: https://ethereum.org/pt-br/web3/. Acesso em: 19 jul. 2023.

EXAME. **Como funciona a tecnologia blockchain?**. [*S. l.*], 24 dez. 2021. Disponível em: https://exame.com/future-of-money/como-funciona-a-tecnologia-blockchain/. Acesso em: 18 dez. 2023.

FRANKENFIELD, JAKE. **Blockchain Wallet: What It Is, How It Works, Security Issues**. [*S. l.*], 13 jan. 2022. Disponível em: https://www.investopedia.com/terms/b/blockchain-wallet.asp. Acesso em: 3 out. 2023.

FRANKENFIELD, JAKE. **Cryptocurrency Explained With Pros and Cons for Investment**. [*S. l.*], 21 abr. 2023. Disponível em: https://www.investopedia.com/terms/c/cryptocurrency.asp. Acesso em: 13 jul. 2023.

FRANKENFIELD , JAKE. **What Is Ether (ETH)? Definition, How It Works, Vs. Bitcoin**. [*S. l.*], 31 maio 2023. Disponível em: https://www.investopedia.com/terms/e/ether-cryptocurrency.asp. Acesso em: 28 out. 2023.

GEORGE, Benedict. **What Is a Dapp? Decentralized Apps Explained**. [*S. l.*], 12 jan. 2022. Disponível em: https://www.coindesk.com/learn/what-is-a-dapp-decentralized-apps-explained/. Acesso em: 17 jul. 2023.

GIL, Antônio Carlos. Como Elaborar Projetos de Pesquisa. 6. ed. São Paulo: Editora Atlas Ltda., 2017.

HACKEN. **Most Common Smart Contract Attacks**. [*S. l.*], 19 set. 2023. Disponível em: https://hacken.io/discover/most-common-smart-contract-attacks/. Acesso em: 23 out. 2023.

HAYES, Adam. **Blockchain Facts: What Is It, How It Works, and How It Can Be Used**. [*S. l.*], 23 abr. 2023. Disponível em: https://www.investopedia.com/terms/b/blockchain.asp. Acesso em: 12 jul. 2023.

HUSSEY, Matt. **What is MetaMask? How to Use the Top Ethereum Wallet**. [*S. l.*], 3 maio 2022. Disponível em: https://decrypt.co/resources/metamask. Acesso em: 12 out. 2023.

IMMUNEBYTES. **DApps Security: All You Need To Know**. [*S. l.*], 4 ago. 2022. Disponível em: https://www.immunebytes.com/blog/dapp-security/#Benefits_of_DApps_Security. Acesso em: 21 out. 2023.

INFOMONEY. **Guia sobre Bitcoin: conheça a origem da primeira criptomoeda do mundo**. [*S. l.*], 8 nov. 2022. Disponível em: https://www.infomoney.com.br/guias/o-que-e-bitcoin/. Acesso em: 16 jul. 2023.

IREDALE, Gwyneth. **The Difference Between Fungible And Non-Fungible Tokens**. [*S. l.*], 15 out. 2023. Disponível em: https://101blockchains.com/fungible-vs-non-fungible-tokens/. Acesso em: 15 out. 2023.

MORELAND, Kirsty. **O que são Chaves Públicas e Chaves Privadas?**. [*S. l.*], 12 dez. 2022. Disponível em: https://www.ledger.com/pt-br/academy/seguranca/o-que-sao-chaves-publicas-e-chaves-privadas. Acesso em: 5 out. 2023.

NAKAMOTO, Satoshi. **A Peer-to-Peer Electronic Cash System**. [*S. l.*], 3 jan. 2009. Disponível em: https://bitcoin.org/bitcoin.pdf. Acesso em: 18 dez. 2023.

NELSON , Jude. **Bringing ‘Clarity’ to 8 Dangerous Smart Contract Vulnerabilities**. [*S. l.*], 11 mar. 2020. Disponível em: https://stacks.org/bringing-clarity-to-8-dangerous-smart-contract-vulnerabilities/. Acesso em: 23 out. 2023.

PIXELPLEX. **Most Common Smart Contract Vulnerabilities and How to Prevent Them**. [*S. l.*], 15 dez. 2022. Disponível em: https://pixelplex.io/blog/smart-contract-vulnerabilities/. Acesso em: 26 out. 2023.

SMITH, Corwin. **INTRO TO ETHEREUM**. [*S. l.*], 12 abr. 2023. Disponível em: https://ethereum.org/en/developers/docs/intro-to-ethereum/. Acesso em: 16 jul. 2023.

SHARMA, RAKESH. **What Is Decentralized Finance (DeFi) and How Does It Work?**. [*S. l.*], 21 set. 2021. Disponível em: https://www.investopedia.com/decentralized-finance-defi-5113835. Acesso em: 11 out. 2023.

SHARMA, RAKESH. **Non-Fungible Token (NFT): What It Means and How It Works**. [*S. l.*], 6 abr. 2023. Disponível em: https://www.investopedia.com/non-fungible-tokens-nft-5115211. Acesso em: 15 out. 2023.

STACKPOLE, Thomas. **What Is Web3?**. [*S. l.*], 22 out. 2023. Disponível em: https://hbr.org/2022/05/what-is-web3. Acesso em: 26 out. 2023.

VERY MONEY. **O que é blockchain e porque ela é tão importante em 2022?**. [*S. l.*], 24 set. 2022. Disponível em: https://verymoney.com.br/blockchain/. Acesso em: 13 jul. 2023.

WAZLAWICK, R. S. Metodologia da Pesquisa para Ciência da Computação. 2ª. ed. [S.l.]: Campus, 2014.