UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Luis Benito Sanches Bulhões

Análise Forense sobre o uso do OPRETURN em Transações Bitcoin

Uberlândia, Brasil

2022

UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Luis Benito Sanches Bulhões

Análise Forense sobre o uso do OPRETURN em Transações Bitcoin

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Ciência da Computação.

Orientador: Rodrigo Sanches Miani

Universidade Federal de Uberlândia – UFU

Faculdade de Ciência da Computação

Bacharelado em Ciência da Computação

Uberlândia, Brasil

2022

Luis Benito Sanches Bulhões

Análise Forense sobre o uso do OPRETURN em Transações Bitcoin

Trabalho de conclusão de curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, como parte dos requisitos exigidos para a obtenção título de Bacharel em Ciência da Computação.

Trabalho aprovado. Uberlândia, Brasil, 18 de agosto de 2022:

Rodrigo Sanches Miani
Orientador

Ivan da Silva Sendin
Convidado 1

Silvio Ereno Quincozes
Convidado 2

Uberlândia, Brasil
2022

Resumo

As criptomoedas revolucionaram o mundo, provendo uma forma rápida, segura e descentralizada de realizar transações ao redor do mundo. E, atualmente o mercado de criptomoedas vem se mostrando lucrativo com moedas chegando a valores acima de 60 mil dólares, como foi o caso da *Bitcoin*, e diversas criptomoedas sendo criadas para diversos tipos de finalidades, seja *smart contracts* ou jogos *play to earn*. A padronização da instrução *OP_RETURN* na *Bitcoin* possibilitou a inserção de dados de qualquer formato em transações com maior facilidade. Este trabalho analisa o conteúdo da instrução em transações realizadas entre 2014 e 2021 e busca traçar um comparativo entre tais conteúdos com o perfil dos usuários responsáveis pelas transações. Os resultados mostram que a maioria das transações que utilizam a instrução *OP_RETURN* foram mapeadas como conteúdos não legíveis na linguagem humana. Acerca do conteúdo legível presente em tal campo, foram encontrados potenciais protocolos usados por serviços diversificados, URLs relacionadas a torrents, memes e marcações de eventos relacionados à *Bitcoins*. Desde sua padronização, o *OP_RETURN* teve um crescente ganho de popularidade, atingindo o pico em 2019 com 14% das transações *Bitcoin* utilizaram esta funcionalidade. Entre os protocolos encontrados, notou-se uma grande diversidade, com empresas utilizando o *OP_RETURN* para fins de registros de documentos ou utilizam como parte da infraestrutura de uma outra *blockchain*, como foi visto no caso da *THORChain*.

Palavras-chave: Criptomoedas, *Bitcoin*, *OP_RETURN*, Análise forense.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

API	Application Programming Interface
CTB	Curve-Tor-Bitcoin
UTXO	Unspent transaction outputs
P2PKH	Pay To Pubkey Hash
BTC	Bitcoin

Sumário

1 Introdução

Bitcoin é uma moeda digital que permite pagamento instantâneo para qualquer pessoa, em qualquer lugar do mundo. É utilizada uma infraestrutura de rede *peer-to-peer* onde mineradores, de forma coletiva, trabalham para autenticar as transações. Logo, nota-se que a *Bitcoin* provê operações sem uma autoridade central, ou seja, a gerência de transações e da emissão de dinheiro é executada coletivamente pela rede. *Bitcoin Core* é o nome do software *open source* que habilita o uso desta moeda (AGNER, 2016). Para garantir a consistência e integridade dos dados são utilizados diversos métodos criptográficos, como por exemplo, funções de *hash* para verificar a autenticidade das transações realizadas na rede.

Cada usuário na rede *Bitcoin* possui um par de chaves, pública e privada. Em geral, os usuários mantêm o par de chaves em um arquivo, que é a chamada carteira de *Bitcoin*. O envio de *Bitcoins* exige a assinatura da transação com a chave privada, que deve ser mantida em sigilo pelo usuário. A chave pública é derivada da chave privada e não precisa ser mantida em sigilo, sendo utilizada para o recebimento de *Bitcoins* (ANTONOPOULOS, 2014).

As transações de *Bitcoins* consistem na transferência de recursos entre os usuários da moeda (ANTONOPOULOS, 2014). Elas possuem uma ou mais entradas, que representam a origem dos recursos e uma ou mais saídas, que são os destinatários das *Bitcoins*. Uma interessante inovação diretamente relacionada a *Bitcoin* é o conceito de *blockchain* (NAKAMOTO, 2008), que funciona como o livro razão da *Bitcoin*, onde terá como conteúdo todas as informações principais das transações validadas pelos mineradores. Uma transação somente será inserida na *blockchain* após os nós da rede, ou seja os mineradores, entrarem em um consenso de que aquela transação é válida. Conforme mencionado anteriormente, esse mecanismo funciona como um banco de dados distribuído e elimina a necessidade de uma autoridade central para verificação das transações. Outra funcionalidade importante da *Bitcoin* é o anonimato entre as partes envolvidas na transação. Os usuários são identificados apenas pelos endereços de *Bitcoin* que consistem no hash da chave pública do usuário.

Com a popularidade crescente da *Bitcoin*, que pode ser observado na Figura 1, os desenvolvedores inseriram novas funcionalidades para aumentar as capacidades do usuário ao utilizar o serviço. E, na atualização do cliente *Bitcoin Core version 0.9.0*, ocorrida em 2014, teve-se a padronização da instrução *OP_RETURN* (ANTONOPOULOS, 2014). Tal funcionalidade possibilita usuários inserirem dados conforme o mesmo desejar, permitindo anexar metadados, como *timestamps*, referentes as transações realizadas, mas

também permite inserção de dados que não dizem nada a respeito das transações sejam armazenados na *blockchain*. Portanto, essa atualização trouxe uma divisão entre a comunidade *Bitcoin*, onde um lado defende o seu uso por mostrar a capacidade que a *blockchain* possui e o outro lado coloca em dúvida se a inserção de dados arbitrários poderá prejudicar a performance da rede *Bitcoin* (BARTOLETTI; POMPIANU, 2017).

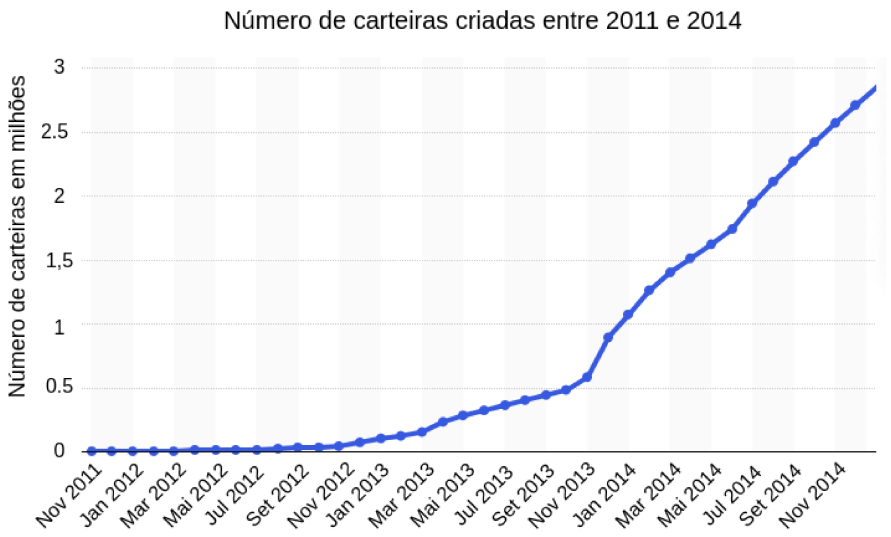


Figura 1 – Criação de carteiras *Bitcoins* entre 2011 e 2014. Nota-se durante o ano de 2014 um grande crescimento de usuários na rede *Bitcoin*. Dados retirados do site (BLOCKCHAIN, 2022)

Além da discordância existente entre a comunidade, alguns trabalhos mostram formas de usar a instrução *OP_RETURN* de forma maliciosa (ALI et al., 2018). Um exemplo é possibilidade de se criar uma *botnet* descentralizada usando o conteúdo do *OP_RETURN* para a troca de mensagens entre o servidor de comando e controle, e os *bots* (ALI et al., 2018).

Por conta destas características, este trabalho visa realizar uma análise sobre o perfil do uso do *OP_RETURN*, por exemplo, investigar os valores gastos nas transações e os tipos de conteúdos inseridos nas mesmas. A análise foi feita levando em consideração transações ocorridas entre 2014 e 2021. Esta escolha de tempo se dá por causa do ano inicial onde ocorreu a padronização da instrução, em 2014, e o ano anterior ao término desta pesquisa, em 2021.

Espera-se obter um melhor entendimento de como foi utilizado a funcionalidade *OP_RETURN* a partir de sua padronização na atualização 0.9.0, observando desde a quantidade de transações realizadas entre 2014 até 2021 e até os tipos de conteúdos inseridos nelas.

Este trabalho têm os capítulos organizados da seguinte maneira:

- No Capítulo 2 será apresentado a fundamentação teórica para este trabalho, além de trazer os trabalhos correlacionados.
- O Capítulo 3 detalhará a metodologia empregada neste trabalho e informa quais ferramentas e linguagens foram usadas.
- Os resultados obtidos estão presentes no Capítulo 4, além dos detalhes de armazenamento dos dados coletados.
- As considerações finais e conclusão do trabalho serão feita no Capítulo 5.

2 Revisão Bibliográfica

Neste capítulo será descrito toda a fundamentação teórica para um melhor entendimento do trabalho. Primeiramente, será explicado sobre a estrutura de dados na qual a rede *Bitcoin* utiliza, a *blockchain*, depois seguirá com uma breve descrição de criptomoedas seguido de uma visão geral de *Bitcoin* e será explicado como funciona a funcionalidade *OP_RETURN*. Por fim, será feita uma breve análise dos trabalhos correlatos.

2.1 Blockchain

A *blockchain* é uma estrutura de dados ordenada de lista encadeadas de blocos de transações, pode ser armazenada tanto como um arquivo ou um banco de dados (ANTONOPOULOS, 2014). Os blocos que fazem parte da *blockchain* são identificados por uma *hash* e a sua formação é aplicada ao sistema *Hashcash proof of work*, que é um processo custoso e demorado cujo objetivo é encontrar um elemento do bloco chamado *nonce*, ao qual é um número aleatório (NAKAMOTO, 2008). Este processo de descoberta do *nonce* é realizado por tentativa e erro pelos mineradores e ao solucionar o problema, o bloco será inserido na *blockchain*, consequentemente gerando uma recompensa aos mineradores que solucionaram o *proof of work* (NAKAMOTO, 2008).

Cada bloco gerado terá em sua composição os dados das transações, que são validadas pelos mineradores, o *nonce* e a *hash* do bloco anterior, evitando alterações na *blockchain* da *Bitcoin*, já que para alterar um dado de um bloco deverá ser alterado o bloco anterior a ele (NAKAMOTO, 2008). Logo, observa-se que os mineradores exercem um papel fundamental na rede *Bitcoin* e que a *blockchain* serve como um livro razão para os usuários da rede *Bitcoin*. A Figura 2 ilustra de maneira simples como é a formação da *blockchain*.

2.2 Criptomoedas

Criptomoeda é uma moeda virtual na qual pertencerá a um ecossistema próprio, por exemplo a rede *Bitcoin* utiliza a criptmoeda *Bitcoin* (*BTC*). Vale apontar que no geral criptomoedas são utilizadas em redes *peer-to-peer*, utilizam *blockchains* como livro-razão e podem apresentar diversas funcionalidades, por exemplo a *Ethereum* possui a funcionalidade de *smart contracts*.

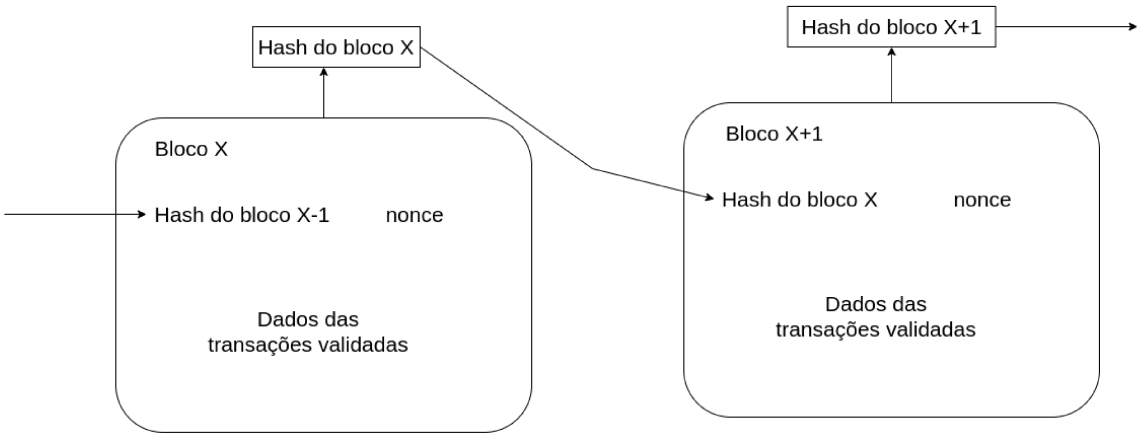


Figura 2 – Exemplo da estruturação dos blocos da *blockchain* da *Bitcoin*. Neste exemplo o bloco X terá a *hash* do bloco X-1 e o bloco X+1 terá a *hash* do bloco X, esse padrão se repete aos blocos antecedentes e posteriores. Adaptado de (NAKAMOTO, 2008)

2.3 Bitcoin

A criptomoeda *Bitcoin* criada em 2008 por Satoshi Nakamoto. A principal ideia o autor era propor um sistema de dinheiro eletrônico descentralizado, onde um usuário pode realizar pagamentos à outros usuários sem a necessidade de uma instituição financeira por trás do sistema (NAKAMOTO, 2008).

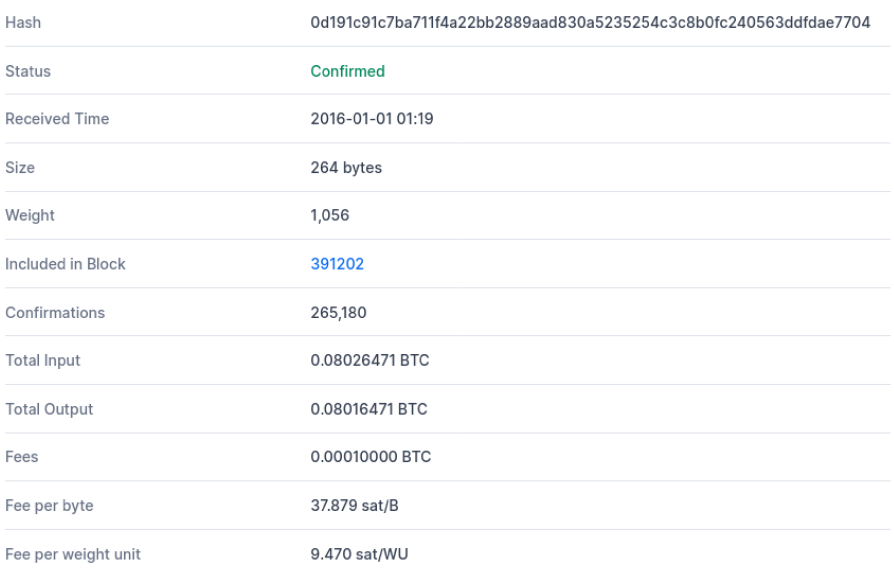
Bitcoin é, em sua essência, um conjunto de princípios e tecnologias que forma o seu ecossistema. A rede *Bitcoin* é conectada de forma *peer-to-peer* que trabalha de maneira coletiva, descentralizada e anonima entre os nós mineradores, para validar as transações realizadas pelos usuários. Com a validação das transações pelos mineradores, a informação referente a ela será inserida em um bloco da *blockchain*. A rede *Bitcoin* utiliza o protocolo *Bitcoin* para realizar as comunicações, que é usado na maioria das vezes via *Internet*, mas também pode ser usada em outras redes de transporte (ANTONOPOULOS, 2017).

As transações são realizadas entre usuários da *Bitcoin* que possuem arquivos que são chamados de carteiras. Cada carteira tem em si uma chave pública e uma privada, onde a pública é a que permitirá o recebimento de *Bitcoins* nas transações e a privada é utilizada para enviar as *Bitcoins* do usuário. As funções *hashes* são o que permitem as validações e criações de transações e carteiras, e sempre quando um minerador valida uma transação ocorre o pagamento de uma taxa pelo trabalho ao qual ele realizou.

Na Figura 3 há um exemplo de uma transação realizada na rede *Bitcoin*. Os seguintes campos merecem destaque:

- Hash: identificador de uma transação, sendo único;
- *Status*: informa se a transação já pertence a um bloco válido ou não;

- *Included in Block*: será a altura do bloco na qual a transação está inserida;
- *Total Inputs*: o valor inserido pelo usuário que será enviado;
- *Total Outputs*: valor total após ser subtraído a taxa;
- *Fees*: taxa do minerador para incluir a transação em um bloco.



Hash	0d191c91c7ba711f4a22bb2889aad830a5235254c3c8b0fc240563ddfdae7704
Status	Confirmed
Received Time	2016-01-01 01:19
Size	264 bytes
Weight	1,056
Included in Block	391202
Confirmations	265,180
Total Input	0.08026471 BTC
Total Output	0.08016471 BTC
Fees	0.00010000 BTC
Fee per byte	37.879 sat/B
Fee per weight unit	9.470 sat/WU

Figura 3 – Exemplo de uma transação *Bitcoin*. Imagem retirada do site (BLOCKCHAIN, 2022)

As transações *Bitcoin* possuem campos de entradas (*inputs*) e de saídas (*outputs*), as Figuras 4 e 5 ilustram como são estes campos na transação.

Inputs HEX ASM

Index	0	Details	Output
Address	1Po1oWkD2LmodfkBYiAktwh76vkF93LKnh	Value	0.00102730 BTC
Pkscript	OP_DUP OP_HASH160 fa0692278afe508514b5ffee8fe5e97732ce0669 OP_EQUALVERIFY OP_CHECKSIG		
Sigscript	3045022100e895c6d4e418b01b57db8c8e63e9d83e2ce17865ee4966a99a37de6a2b09dc5f02203077aaf5c3e259c7a5ac8b89c24f58b59bffd0ab97f048ca423f85a3baa63a8901 04fcf07bb1222f7925f2b7cc15183a40443c578e62ea17100aa3b44ba66905c95d4980aec4cd2f6eb426d1b1ec45d76724f26901099416b9265b76ba67c8b0b73d		
Witness			

Figura 4 – Exemplo do campo de *input* de uma transação *Bitcoin*. Imagem retirada do site (BLOCKCHAIN, 2022)

Com a Figura 4 consegue-se visualizar como é a estrutura do campo. A seguir, cada elemento do campo *input* será detalhado:

- *Index*: refere-se a posição dos dados no campo de *input* da transação;

- *Address*: contém o endereço das carteiras que estão enviando *Bitcoins*;
- *Value*: o valor a ser enviado;
- *Pkscript*: é o que garante que a transação é valida, é também conhecido como *locking script*;
- *Sigscript*: contém as assinaturas necessárias e o *script* que permitirão liberar *UTXO* para gastar;
- *Witness*: usado em transações *SegWit* para liberar as *Bitcoins*.

Com a explicação feita sobre a estrutura do campo de *inputs* de uma transação *Bitcoin*, agora será detalhado o campo de *output*, lembrando que a Figura 5 mostra um exemplo deste campo. O campo de *output* conterá os seguintes elementos:

- *Index*: refere-se a posição dos dados no campo de *output* da transação;
- *Address*: são os endereços recebedores de *Bitcoins*;
- *Value*: o valor a ser recebido;
- *Pkscript*: similar ao campo de *inputs*, mas é aqui onde pode ser encontrado o *OP_RETURN*;

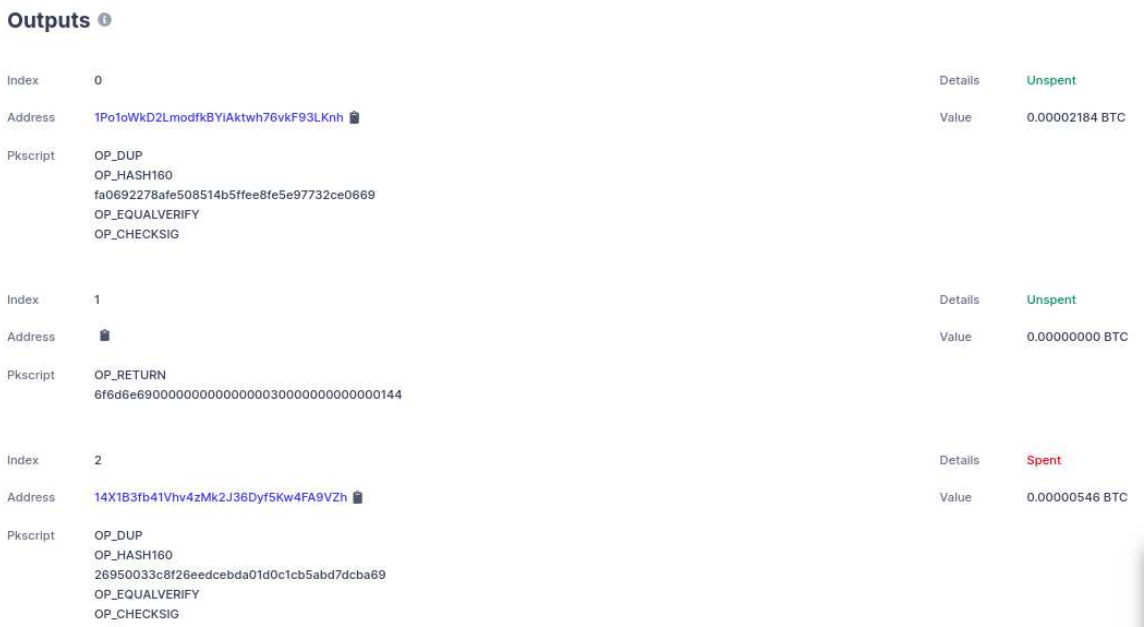


Figura 5 – Exemplo do campo de *outputs* de uma transação *Bitcoin*. Imagem retirada do site (BLOCKCHAIN, 2022)

Um aspecto interessante a ser apontado é que as transações de recompensas para os mineradores, seja pela formação de um bloco ou da validação de uma transação, o

campo de *input* será distinto do que observou-se na Figura 4. Como pode ser visto na 6, o campo de *input* é vazio nas transações de recompensa.

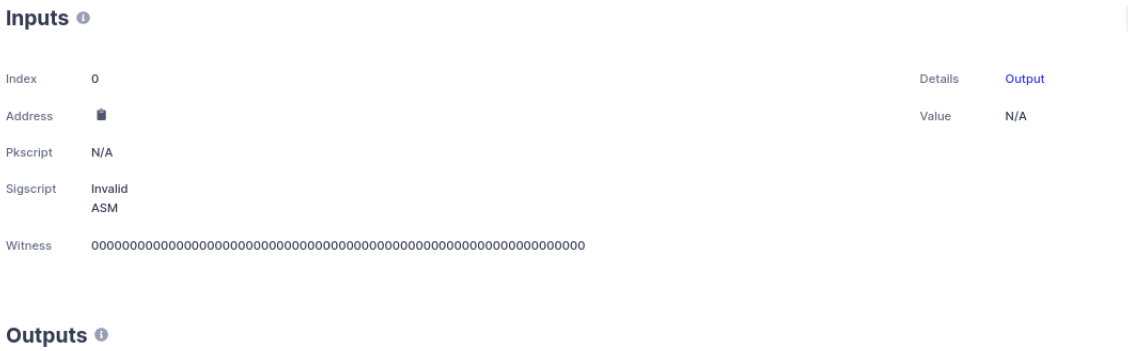


Figura 6 – Exemplo do campo de *input* de uma transação de recompensa para um minerador na rede *Bitcoin*. Imagem retirada do site (BLOCKCHAIN, 2022)

2.4 OP_RETURN

Uma das funcionalidades interessantes da *Bitcoin* é a possibilidade de inserir metadados, conforme o usuário desejar, utilizando um *script opcode*, o *OP_RETURN*, padronizado na *Bitcoin Core version 0.9.0*, possibilitando o usuário inserir cerca de 80 *bytes* de dados, que aparecerá nos campos de saídas (*outputs*) das transações. Com o ganho de popularidade, entre os seis anos de sua padronização, surgiram empresas que proporcionam serviços diversificados utilizando o *OP_RETURN* por meios de protocolos. Os serviços propostos são financeiro, que tem como objetivos manter um registro das transações realizadas pelo usuário e a comprovação de posse. Também há serviços que promovem direitos autorais e integridade de documentos, além disso têm-se protocolos que inserem mensagens que o usuário desejar (BARTOLETTI; BELLOMY; POMPIANU, 2019).

Além de serviços proporcionados, há como utilizar o *OP_RETURN* para se inserir metadados como arquivos, imagens e *scripts*. Arquivos e imagens são comprimidos e serão retransmitidos, podendo ser necessário realizar diversas transações caso o tamanho do arquivo ou imagem seja grande, já que o *OP_RETURN* tem um limite de 80 *bytes*. Já os *scripts* podem variar desde *Python* a jogos (BARTOLETTI; BELLOMY; POMPIANU, 2019).

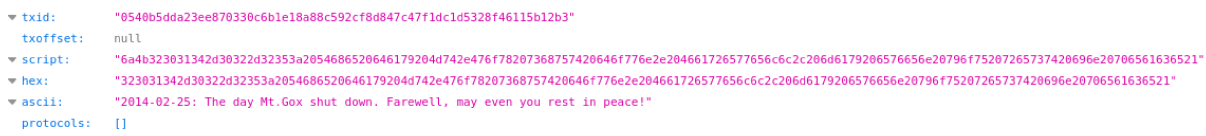


Figura 7 – Exemplo de transação que utiliza o *OP_RETURN*. Imagem retirada do site (COINSECRETS, 2020)

Observando a Figura 7, o campo *txid* refere-se ao *id* da transação realizada, *hex* contém o conteúdo que o usuário quer inserir e o campo *script* possui em seu início o *opcode* 6a no qual se refere ao *opcode* *OP_RETURN* (BITCOIN, 2022). O campo *ascii* era provido pela *API* do *coinsecrets* como uma tentativa de tradução do *hex* e caso fosse algo reconhecido pela *API*, seria marcado no campo *protocols* o protocolo sendo utilizado.

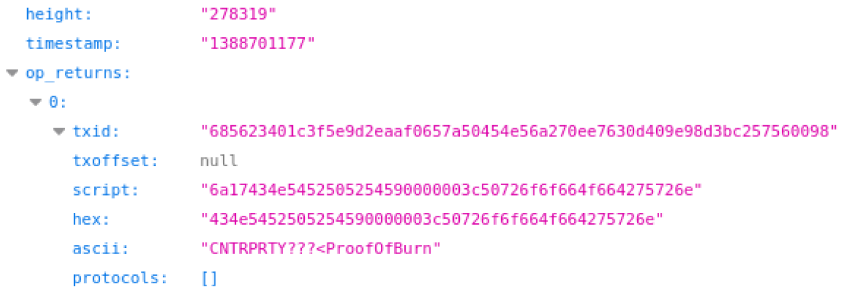


Figura 8 – Mais um exemplo de uso de *OP_RETURN* em uma transação. Imagem retirada do site (COINSECRETS, 2020)

Na Figura 8, a utilização do *OP_RETURN* é usado para inserir dados relacionado ao serviço disponibilizado pela *Counterparty*, que é um protocolo cujo objetivo é validar os direitos autorais do usuário sobre os ativos *Bitcoins*.

Algo interessante a ser apontado é que a própria rede *Bitcoin* chega a utilizar o *OP_RETURN* nas transações de recompensas para os mineradores. A Figura 9 ilustra o campo de *output* de uma transação de recompensa.

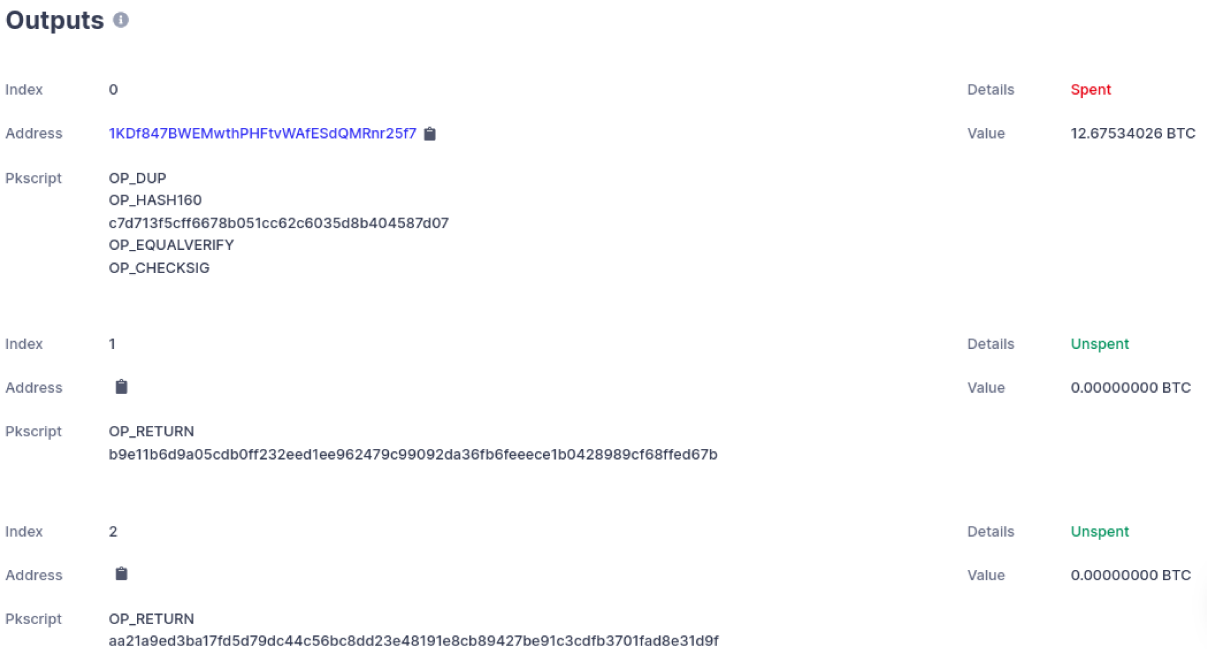


Figura 9 – Exemplo do campo de *output* de uma transação *Bitcoin* que é uma recompensa para um minerador. Imagem retirada do site (BLOCKCHAIN, 2022)

2.5 Trabalhos correlatos

Nesta seção serão discutidos os trabalhos que se relacionam com o tema deste trabalho de forma geral: análises forenses associadas ao uso do *OP_RETURN*.

No trabalho de Bartoletti e Pompianu (2017) foi realizada uma análise sobre os protocolos usados no *OP_RETURN*, sendo classificados de acordo com o prefixo dos dados inseridos com o *script*. Identificaram ataques de estresse e de *spam* na rede *Bitcoin*, no qual foi evidente por conta do grande número de transações que usaram o *OP_RETURN* em um curto período de tempo. Conseguiram estimar o tamanho da ocupação, das transações com *OP_RETURN*, na *blockchain*, assim como a porcentagem das transações realizadas utilizando o *script*. O trabalho utilizou todas as transações com *OP_RETURN* no intervalo de tempo entre 2014 a início de 2017 e notaram a crescente adoção do uso desta funcionalidade pelos usuários *Bitcoin*.

Ali et al. (2018) trabalharam na possibilidade de se criar uma rede *botnet* descentralizada usando mensagens criptografadas via *OP_RETURN* para se comunicar com os *bots*. O experimento obteve sucesso na comunicação entre o *botmaster* e *bots* e concluiu que é algo a ser debatido, já que essa possibilidade permite uma *botnet* de baixo custo e difícil de contra-atacar ou de desmantelar.

Faisal, Courtois e Serguieva (2018) analisaram os seguintes aspectos: a evolução da inserção de metadados sobre as transações e outra análise no uso da rede *Bitcoin* por criminosos. Chegou a identificar muitas peculiaridades sobre atividades na rede que provavelmente têm tendências maléficas como o *ransomware CTB Locker* ao qual usa um método que incorpora uma chave de descriptografia no *OP_RETURN*.

Böck et al. (2019) estudou o caso de *botnets* utilizando a *blockchain* para realizar comunicação entre *botmaster* e os *bots*. Dentre as *botnets* estudadas, a *ZombieCoin* é considerada uma das de menor custo, reforçando a ideia do artigo Ali et al. (2018) do baixo custo operacional, mas perde na questão de ocultamento, já que se usa um espaço da transação que ficará visível para qualquer um.

Os benefícios e malefícios que a inserção de dados não vinculados à transação na *blockchain*, como o *OP_RETURN*, foi estudado no artigo publicado por Matzutt et al. (2018). Os pontos positivos são os serviços que se aproveitam da inserção de metadados para comprovações de documentos. Esta escolha se dá por conta do fato de que os dados na *blockchain* são praticamente imutáveis, uma vez inseridos nela. Além disso, "pesquisadores"apontaram o uso da *blockchain* para armazenar dados de denunciantes (*whistleblowers*) de informações sigilosas de governos e de grandes entidades privadas, tal fato é comprovado, pois eles apontam ter encontrado *links* para *backups* dos dados do *WikiLeaks*. Apesar destes pontos positivos, os negativos são preocupantes, pois foram encontrados transações que infringem direitos autorais, propagando *links* para o *download*

de conteúdos pirateados, possíveis técnicas de *doxing*, com diversas informações privadas de indivíduos, sendo elas números de telefones, endereços, contas bancárias, senhas e identidades *onlines* e, por fim, encontraram um *hidden file* que era um *backup* de listas de *links* com conteúdos de pornografia infantil.

Mols e Vasilomanolakis (2020) desenvolveram um software capaz de classificar o conteúdo inserido no *OP_RETURN*, e com os testes que eles realizaram descobriram imagens e arquivos escondidos, e conseguiram classificar os protocolos utilizados durante as transações.

Foi apontado no artigo feito por Strehle e Steinmetz (2020) o crescente uso do *OP_RETURN* por serviços como *Veriblock* e *Omni*, o primeiro é um projeto de *blockchain* que utiliza o conceito de *Proof of Proof* (STREHLE; STEINMETZ, 2020), já o segundo é um serviço que promove uma das *stablecoins*, a *Tether*. Foi concluído que estes serviços predominam o uso do *script OP_RETURN*, com o *Veriblock* chegando a ser um pouco mais da metade das transações que utilizam esta funcionalidade.

Bartoletti, Bellomy e Pompianu (2019) exploram métodos de inserções de metadados em transações, entre eles o *OP_RETURN*, e analisaram mais afundo o uso desta funcionalidade dentre os anos de 2014 a 2017. Classificaram os protocolos utilizados em financeiro, registros de cartórios, registros autorais, mensagens e *Subchain*. Abordaram os principais problemas de se inserir metadados nas transações, como consumo de espaço, UTXO *bloating effect* e picos de transações. Apontaram que apesar de que na própria documentação oficial da *Bitcoin*, desencoraja a inserção de dados nas transações. Nota-se uma crescente adoção de usuários desta funcionalidade.

Os trabalhos Bartoletti e Pompianu (2017), Strehle e Steinmetz (2020), Faisal, Courtois e Serguieva (2018), Matzutt et al. (2018) e Bartoletti, Bellomy e Pompianu (2019) apresentam semelhanças com o presente trabalho, ou seja, uma análise do uso e evolução do *OP_RETURN*. Nestes artigos apenas analisou-se determinados protocolos disponibilizados como serviços de terceiros, mas apenas nas pesquisas de Matzutt et al. (2018) e Faisal, Courtois e Serguieva (2018) é explorado mais profundamente o uso do *opcode*, chegando a atividades de caráter criminoso. Em Mols e Vasilomanolakis (2020) desenvolveram uma ferramenta com um grande potencial em auxiliar na extração de dados inseridos em uma transação que utilizou o *script OP_RETURN*, de acordo com o artigo, foi feita uma catalogação de protocolos conhecidos, além de conseguir recriar arquivos do tipo imagem, inseridos através de uma conversão para hexadecimal. Por fim, Böck et al. (2019) e Ali et al. (2018) com as análises sobre o uso de *botnets*, onde Böck et al. (2019) reforça as ideias que foram abordadas por Ali et al. (2018), destacando o potencial das *botnets* com comunicações via transações. Apesar dos trabalhos citados possuírem similaridades com o que está sendo abordado nesta pesquisa, será analisado uma linha de tempo maior que as análises feitas anteriormente, de 2014 a 2021. Além

disso, será estudado os conteúdos das transações com *OP_RETURN*, a fim de tirar as devidas conclusões de como está sendo usado o *opcode*.

3 Desenvolvimento

Este capítulo é dedicado a discutir sobre o desenvolvimento do trabalho, detalhando cada etapa realizada. Primeiramente, será discutida a metodologia empregada, deixando claro quais ferramentas e linguagens foram usadas, depois ocorrerá o detalhamento das etapas descritas.

3.1 Metodologia

A metodologia deste trabalho foi dividida em quatro passos que podem ser vistos, de maneira geral, na Figura 10.

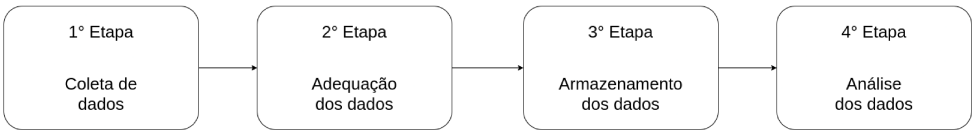


Figura 10 – Ilustração com cada etapa relacionada ao desenvolvimento deste projeto.

As ferramentas utilizadas para auxiliar na execução deste trabalho foram o *MongoDB*, um banco de dados não relacional, e a linguagem *Python* na versão 3.8.1, por conta da existência bibliotecas que facilitaram o alcance dos objetivos deste trabalho. As seguintes bibliotecas do Python foram usadas:

- *plotly*: para construção de gráficos;
- *pymongo*: integra o código com o banco de dados;
- *networkx*: para construção de grafos;
- *re*: biblioteca de expressões regulares;
- *requests*: usada para o *download* dos dados.

3.2 Coleta de dados

A primeira etapa, a de coleta, foi conduzida com o auxilio dos seguintes serviços: *Coinsecrets* (COINSECRETS, 2020) e *Blockchair* (BLOCKCHAIR, 2022). Para a coleta no *Coinsecrets* utilizou-se a *API* do próprio site, que permitia trazer informações relacionadas à transação e havia uma tentativa de identificar o conteúdo presente no *script*. As Figuras 7 e 8 ilustram duas transações distintas no site *Coinsecrets*. E, como pode ser

visto, o *Coinsecrets* apenas disponibilizava algumas informações das transações, o resto dos dados foram extraídos do site *Blockchain* (BLOCKCHAIN, 2022), usando a *API* para buscar as transações, pelos *hashes* delas extraídos no *Coinsecrets* anteriormente, com isso foi possível obter as outras informações que faltavam, como endereços de carteiras de entrada e saída, quantidades de *Bitcoins* transferidos e entre outros dados. Com o *Coinsecrets* obteve-se os dados de 2014 até 2019, e como o *site Coinsecrets* foi desativado, este trabalho passou a utilizar os *datadumps* do *Blockchair* (BLOCKCHAIR, 2022), que contém os *outputs* das transações *Bitcoin*, que se mostrou satisfatório para este trabalho, e todos esses dados são originados do site *Blockchain*. Para extrair os dados do *Blockchair* foi implementado um script para automatizar o download de todos os *datadumps* de 2020 até 2021. Os arquivos baixados possuem a extensão *.tsv*, que é um arquivo de texto simples separado por tabulação.

As Figuras 11 e 12 mostram como os *datadumps* do *Blockchair* estão organizados. O campo inicial, *block_id* se refere à altura do bloco da transação e o *transaction_hash* é o próprio *hash* da transação. O *index* indica qual a posição deste dado na transação, inicia-se em 0 este valor, isso se dá pela possibilidade de termos mais de um *output* ou *input* em uma transação *Bitcoin*, como visto anteriormente. Os campos *value* e *value_usd* referem-se aos valores, em dólares, destinados ao endereço de carteira encontrado no campo *recipient*. O campo *type* mostrará qual o *script* foi usado para o envio de *Bitcoins*, caso o *OP_RETURN* foi utilizado, como é o caso da Figura 11, *type* ficará com o valor *nulldata*, mas também pode indicar que a saída da transação está vazia. Já a Figura 12 mostra o valor *pubkeyhash* na transação, logo foi usado o *script P2PKH* para o envio de *Bitcoins*. O *script* utilizado na transação é encontrado no *script_hex*, a transação encontrada na Figura 11 usa o *OP_RETURN*, evidente por ter os dois primeiros caracteres 6a, já a transação da Figura 12 não está usando o *OP_RETURN*. E, por fim, *is_from_coinbase* indica se é recompensa ou não, em ambas imagens usadas como exemplo o valor é 0, portanto não é recompensa.

```
block_id: 610691
transaction_hash: 748b26c5570a8c95250817f1a77ba5075769f5e16adb25af92025fe0284abc3e
index: 1
time: 2020-01-01 00:03:05
value: 0
value_usd: 0.0
recipient: d-49d2e143ac285aa1ab2086050b20495c
type: nulldata
script_hex: 6a206b698f7ac974e32e2cee14d20aa1b829457633a67d4e802a73df2d33ddd7c4d7
is_from_coinbase: 0
```

Figura 11 – Exemplo de dados de uma transação no *datadump* do *Blockchair*.

```
block_id: 611343
transaction_hash: d10a7f5438292a4e066de39e175f2765d31411280206d5a334a25ec3bd9992fc
index: 6
time: 2020-01-05 00:13:31
value: 6700
value_usd: 0.4939
recipient: 1EwZ4s5BNKBydhEJ6NjwiFD3pqUEL2jrct
type: pubkeyhash
script_hex: 76a91498eadd6f8261dd5908622a7c1bc7b34ac3a0c63888ac
is_from_coinbase: 0
```

Figura 12 – Outro exemplo de dados de uma transação no *datadump* do *Blockchair*.

3.3 Adequação dos dados

De posse dos dados, foi possível iniciar a adequação dos mesmo. Primeiramente, foi realizada uma separação do tipo de conteúdo presente no *OP_RETURN*, em conteúdos legíveis e não legíveis. Conteúdo legível é quando o conteúdo é possível de ser lido por humanos, já os conteúdos não legíveis são aqueles não possíveis de leitura humana. Os valores das transações foram armazenadas em *satoshis*, um *satoshi* equivale a 10^{-8} *Bitcoins*, é a menor unidade de *Bitcoin*.

3.4 Armazenamento dos dados

Com a distinção feita sobre os dados na etapa anterior, os dados foram armazenados em um banco de dados não relacional, no caso deste trabalho foi utilizado o *MongoDB*, como dito anteriormente. Vale apontar que foram criados *collections*, termo equivalente as tabelas nos bancos de dados relacionais, no banco de dados referentes aos anos, logo criaram-se oito *colletions* para armazenar os dados entre 2014 até 2021. Como os dados foram extraídos de diferentes fontes e tempo, houve uma diferença da maneira na qual os dados foram armazenados, primeiramente será explicado como os dados de 2014 até 2019 foram salvos.

- _id: campo padrão do *MongoDB*, onde o próprio banco pode gerar uma *string* única, mas também pode ser inserido um dado qualquer, desde que seja única e não se repita para nenhum outro dado inserido. No caso deste projeto, este campo tem o hash da transação de *bitcoin*, já que eles são únicos;
- *script*: terá todo o *script* utilizado na transação;
- hex: possui o conteúdo inserido pelo usuário;
- protocol: campo originado por conta do *Coinsecrets* que tentava associar o conteúdo inserido na transação com um protocolo conhecido pela plataforma. Este campo não

foi utilizado neste projeto já que se procurou uma maior compreensão acerca dos conteúdos inseridos;

- ascii: uma tentativa de conversão para leitura humana pela plataforma do Coinsecrets;
- addresses_senders: contem os endereços remetentes de uma transação. Para as transações de recompensa para mineradores, este campo foi preenchido com a *string* *Newly_generated_coins*;
- values_inputs: valores de entrada de uma transação;
- addresses_receivers: contem os endereços destinatários da transação;
- values_outputs: valores de saída de uma transação;
- block_height: altura do bloco que a transação se encontra;
- day: dia do mês que a transação foi realizada;
- month: mês que a transação foi realizada;
- year: ano de realização da transação.

Como o site *Blockchain* foi parte da coleta de dados de 2014 até 2019, a *API* retornava os campos de *inputs* e *outputs* em ordem, idêntico como pode ser visto nas Figuras 4 e 5. Este detalhe facilitou na etapa de análise, quando se analisou os valores gastos.

É importante ressaltar que inicialmente o escopo do trabalho ainda estava sendo determinado enquanto se analisava os dados extraídos entre 2014 e 2019. Com o passar do tempo a proposta do projeto foi ajustada e a coleta de dados de 2020 e 2021 refletiu o amadurecimento. Com isso esclarecido, abaixo será pontuado como se deu o armazenamento no banco de dados para os anos de 2020 e 2021.

- _id: campo padrão do *MongoDB*, onde o próprio banco pode gerar uma *string* única, mas também pode ser inserido um dado qualquer, desde que seja única e não se repita para nenhum outro dado inserido. No caso deste projeto, este campo tem o hash da transação de *bitcoin*, já que eles são únicos;
- readable_scripts: campo reservado para *scripts* de conteúdos legíveis;
- ascii: terá a tradução de um *script* legível;
- not_readable_scripts: campo reservado para o *scripts* de conteúdos não legíveis;

- addresses_receivers: possuirá os endereços destinatários;
- values_outputs: terá a quantidade de *bitcoins* mandadas para os destinatários.

3.5 Análise dos dados

Com os dados adequados e armazenados, prossegue-se com a etapa de análise. Nesta etapa serão feitas as seguintes análises:

- A classificação e contagem de transações com *OP_RETURN* entre 2014 e 2021;
- Análise dos valores transferidos nas transações *Bitcoin* entre 2014 e 2019;
- Análise dos protocolos inseridos nas transações;
- Expor outros achados que vão além de protocolos;
- Análise da interações dos usuários que utilizaram o *OP_RETURN*.

A classificação das transações é dita em legíveis e não legíveis, lembrando que legíveis serão conteúdos de *scripts* possíveis de leitura humana e não legíveis não são possíveis de leitura. A contagem será feita de maneira simples, onde ao identificar um *script* legível será incrementado o contador de *scripts* legíveis, o mesmo vale para os não legíveis. E, como foi dito anteriormente, os *scripts* de 2014 até 2019 serão traduzidos neste momento.

A análise dos valores transferidos calculou-se a média anual em dólares durante os anos de 2014 até 2019, os anos de 2020 e 2021 foram excluídos, pois não havia como identificar quais valores eram de troco. Detalhando mais sobre o cálculo feito, como não foi salvo no banco de dados os valores em dólares de 2014 até 2019, realizou-se uma conversão para dólares, onde foi pego o valor em dólares a cada final de mês de cada ano, os dados foram extraídos do *Blockchain* (BLOCKCHAIN, 2022). Vale ressaltar que os valores no banco de dados estavam em *satoshis* e foram somados desta maneira para evitar problemas de *overflow* durante as operações matemáticas. Além disso, os dados de 2014 até 2019 possuem informações do campo de *input* das transações, logo possibilitou com maior facilidade identificar quais valores eram de troco.

Com o auxílio de uma expressão regular, que pode ser encontrada no Capítulo 4 no trecho de código 4.4, foi possível identificar diversos conteúdos inseridos com o *OP_RETURN*, que variaram de protocolos até conteúdos ilegais. Primeiramente, foi feito um levantamento dos protocolos usados, que em grande parte foram usados com maior frequência, logo facilitou encontrar diversos protocolos. Para obter um maior entendimento sobre os protocolos utilizou-se o próprio *Google* para buscar por referências e até mesmo

páginas referentes as empresas e equipes por trás dos protocolos, mas alguns protocolos não foram possíveis encontrar nenhuma referência ou não deu para obter certeza sobre o seu uso. Para encontrar os protocolos foi utilizada uma expressão regular que encontra *strings* com três caracteres iniciais distintos, o motivo desta escolha se deu após uma análise minuciosa e conclui-se que os protocolos em geral utilizam três ou mais caracteres, os conteúdos encontrados com esta busca foram armazenados em um arquivo *.csv* para facilitar a leitura. Para cada conteúdo foi colocado um contador para obter-se os *hits* de cada conteúdo, assim descobrindo quantas vezes ele foi inserido nas transações. Foram armazenados nos arquivos *.csv* os conteúdos encontrados e o contador referente a cada dos conteúdos dos *scripts*, a Tabela 1 mostra um exemplo das doze primeiras linhas do *csv* criado para o ano de 2020.

Conteúdo	Quantidade
omni	2.478.959
RSKBLOCK	25.004
WWW	9819
POET	2756
btt	2707
Bitzlato	2011
ChainX	1551
BTCKEY	998
ver	817
xb9	674
PHOTECTOR	645
BERNSTEIN	450

Tabela 1 – Tabela exemplificando como os arquivos *csv* estão organizados.

Depois do levantamento dos protocolos, realizou-se uma análise do restante dos dados, obteve-se diversos achados que variaram desde *links* até mensagens que diversos usuários inseriram, nas quais variam desde mensagens comemorativas até *memes*.

Por fim, realizou-se uma análise de como os usuários da rede *Bitcoin* que utilizaram o *OP_RETURN* se interagem, onde utilizou-se agrupamentos para classificar se um usuário mandou *Bitcoins* para ele mesmo, ou se mandou para uma ou duas carteiras diferentes, ou se mandou para mais de três carteiras diferentes. Para esta análise foram usados os anos de 2014 até 2017, pois o agrupamento do ano de 2017 chegou a demorar mais de um mês, logo os outros levariam muito mais tempo, por conta da quantidade de dados, o que não foi viável para este trabalho.

4 Resultados

Neste capítulo serão mostrados os resultados das análises das transações *Bitcoins*, conforme discutido no capítulo anterior. Primeiramente, será mostrado a quantidade de transações legíveis e não legíveis durante os anos de 2014 até 2021, depois serão apresentados os resultados obtidos com os valores usados nas transações. Serão mostrados em detalhes os conteúdos encontrados, dividido em duas seções, uma dedicada aos protocolos encontrados e a outra exibe o que foi encontrado ao ser feito uma análise minuciosa dos outros conteúdos. E, por fim serão exibidos os resultados encontrados nos agrupamentos de usuários que realizaram as transações utilizando o *OP_RETURN*.

4.1 Classificação de transações e análise da quantidade de transações

As transações coletadas na etapa de coleta foram divididas em dois tipos: legíveis e não legíveis. As legíveis são possíveis de leitura humana. A Figura 13 ilustra um exemplo de uma transação com conteúdo legível e a Figura 14 demonstra uma transação com conteúdo não legível. A Tabela 3 mostra a quantidade de transações com conteúdos legíveis e não legíveis entre 2014 até 2021, juntamente com o total. Logo, com os dados da Tabela 3 nota-se uma tendência na inserção de dados não legíveis utilizando o *OP_RETURN*.

Os dados apresentados na Tabela 2 mostram que entre 2014 e 2019 houve um aumento na quantidade de transações com *OP_RETURN*, entretanto houve uma grande queda depois do ano de 2019. Vale apontar que teve uma queda no número de transações diária de *bitcoins* depois de 2019, conforme observado na Figura 15.

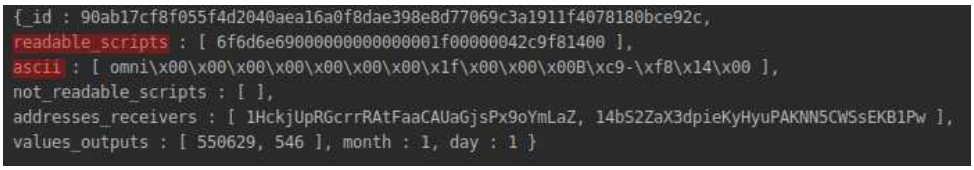


Figura 13 – Exemplo de uma transação com o conteúdo legível, onde grifado em vermelho esta o *script* usado e a conversão dele para leitura humana. A imagem foi retirada do próprio banco de dados usado nesta pesquisa.

As traduções dos conteúdos inseridos com *OP_RETURN* foram feitas com o auxílio do código 4.1.

```
if transaction_data['script_hex'][:2] == "6a":
    ascii_string=\
```

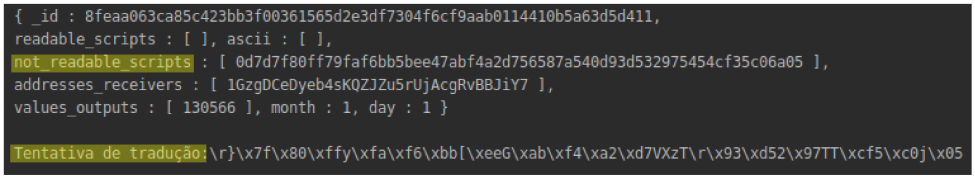


Figura 14 – Exemplo de uma transação com o conteúdo não legível, onde grifado em amarelo tem o *script* usado e também uma tentativa de conversão para leitura humana, mas como pode ser visto, não é possível compreender o que é tal conteúdo.

Ano	Quantidade	Legíveis	Não legíveis
2014	4.005	1.417	2.588
2015	125.916	62.203	63.713
2016	443.876	93.122	350.754
2017	1.285.805	220.595	1.065.210
2018	3.115.893	85.603	3.030.290
2019	16.716.374	140.920	16.575.454
2020	6.334.442	2.573.696	3.760.746
2021	2.861.451	1.306.215	1.555.236

Tabela 2 – Tabela com o total de transações realizadas disponíveis nos *sites coinsecrets.org* e *blockchair.com*

Tipo	Quantidade
Legíveis	4.483.771
Não legíveis	26.403.991
Total	30.887.762

Tabela 3 – Quantidade de transações que usaram *OP_RETURN* entre 2014 e 2021, divididas em legível e não legível, onde as legíveis são transações com *scripts* legíveis quando são traduzidos.

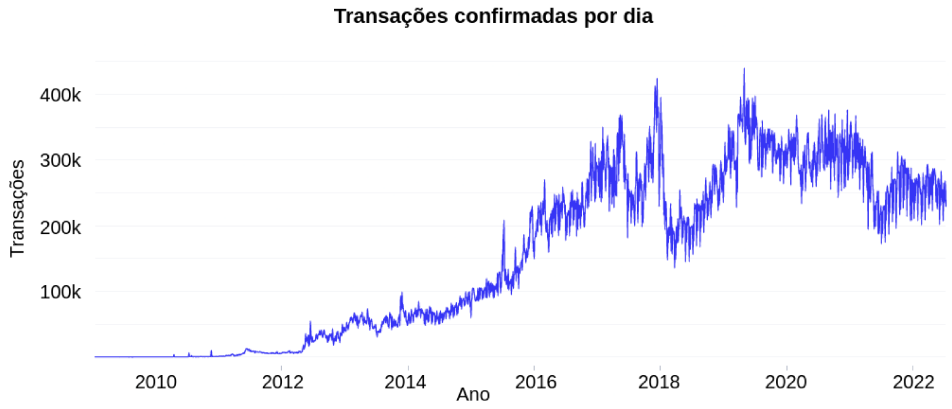


Figura 15 – Quantidade diária de transações de *bitcoin*. As informações foram retiradas do site (BLOCKCHAIN, 2022)

```
    str(bytes.fromhex(transaction_data['script_hex'][4:]))
    ascii_string = ascii_string[2:len(ascii_string) - 1]
```

Listing 4.1 – Trecho do código onde será verificado se o *script* tem o *opcode* do *OP_RETURN*.

O código 4.1 primeiro identifica se a transação está utilizando o *opcode* (6a), caso esteja será feita a conversão de hex para bytes e depois para *string*. No final é retirado os caracteres b' ' que são inseridos na *string* quando ocorre a conversão para bytes, assim deixando a apenas o conteúdo que um usuário inseriu na transação com o *OP_RETURN*. A classificação do conteúdo como legível e não legível foi efetuada da seguinte maneira:

```
if "\\" not in ascii_string[:3]:
    ascii_script.append(ascii_string)
    readable_script.append(tx_data['script_hex'][4:])
else:
    not_readable_script.append(tx_data['script_hex'][4:])
    not_ascii.append(ascii_string)
```

Listing 4.2 – Trecho de código responsável pela classificação de legíveis e não legíveis. Note que apesar de ter o \\no *if* a linguagem *Python* só irá reconhecer o segundo \.

Com a tradução feita anteriormente, o trecho de código 4.2 permite identificar se os três primeiros caracteres não possuem o carácter \. Esta escolha foi feita após uma análise minuciosa dos conteúdos de diversas transações e concluiu que transações que têm \no início são não legíveis , já que não foi possível encontrar conteúdo legível após este tipo de início.

A Figura 16 retrata a porcentagem de transações que usaram o *OP_RETURN* na rede *Bitcoin* entre 2014 e 2021. Observa-se um significativo aumento no número de transações *OP_RETURN* após a sua padronização em 2014. Logo, observa-se uma grande adoção dos usuários da rede *Bitcoin*, chegando a 14% de transações *bitcoins* em 2019. Entretanto, em 2020 e 2021 houve um diminuição bem significativa no percentual de transações utilizando o *opcode*. Lembrando que além da diminuição de transações com *OP_RETURN* a rede *Bitcoin* passou por um diminuição diária no número de transações, como pode ser observado na Figura 15.

4.2 Análise de valores

Outra análise conduzida neste trabalho está relacionada aos valores de transferência em dólares entre 2014 até 2019 em um período anual. Como dito anteriormente, foram utilizados os dados do *site Blockchain*, onde foi possível obter os valores em dólares de

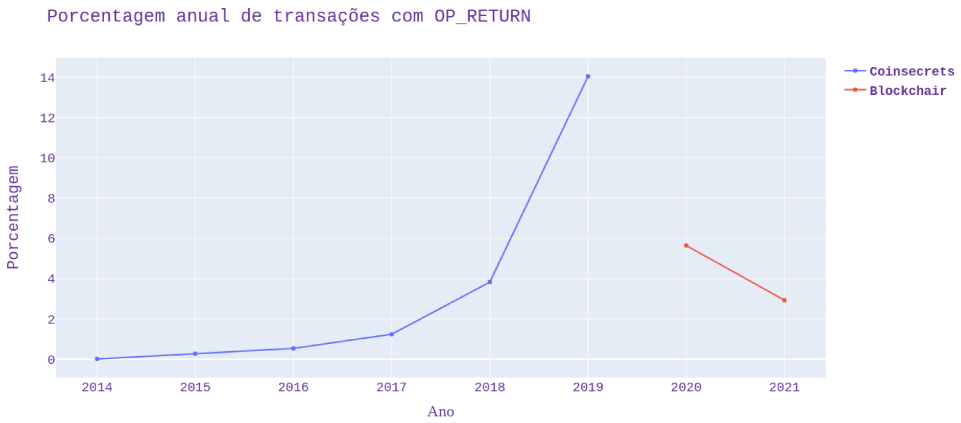


Figura 16 – Porcentagem de transações, entre 2014 e 2021, que utilizaram *OP_RETURN*. Pode ser observado que os dados de 2014 até 2019 foram extraídos do *Coinsecrets* e de 2020 até 2021 foram do *Blockchair*.

cada mês entre os anos de 2014 até 2019. O resultado desta análise é mostrado na Figura 17.

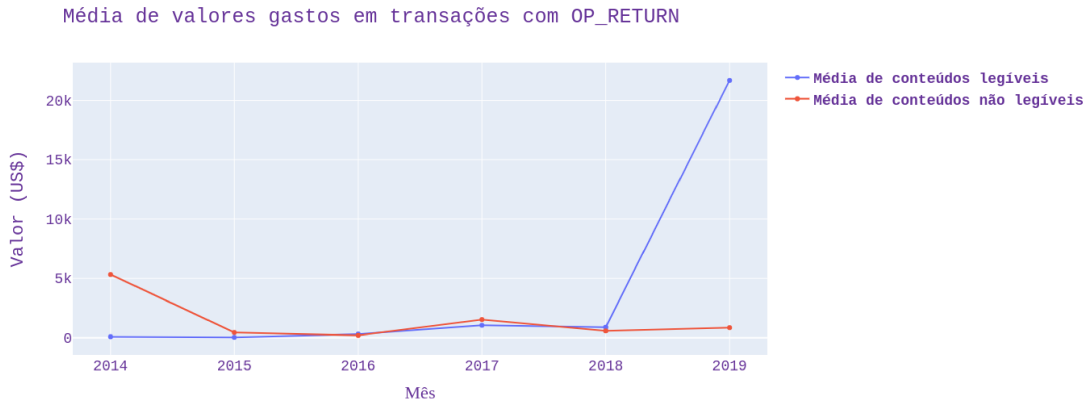


Figura 17 – Valores médios anuais em dólares de transações com *OP_RETURN* realizadas entre 2014 até 2019. Foi divido entre conteúdos legíveis e não legíveis.

Nota-se que nos anos de 2015 até 2018 não foram encontradas muitas divergências de valores entre as duas classificações, diferente do que foi visto em 2014 e 2019. Em 2014 há uma divergência onde os conteúdos não legíveis chegam a uma média de 5 mil dólares. Mas ficou evidente que no o ano de 2019 é o que apresentou a maior divergência com os conteúdos legíveis chegando a ultrapassar uma média de 20 mil dólares, enquanto os não legíveis estavam com uma média mais baixa que 5 mil.

Esta análise ocorreu da seguinte maneira: a utilização dos dados do *Blockchain* possibilitou a obtenção da conversão dos valores de saída das transações em dólares, onde cada cotação foi feita usando o valor médio do último dia dos meses dos anos de 2014 até 2019; Para a exclusão das transações de recompensa dos mineradores, utilizou-se o

campo de *input*, já que essas transações tiveram o campo de endereços de entrada preenchido com a *string Newly_generated_coins*. Como foi visto anteriormente no capítulo de desenvolvimento, foram ignorados todas transações que tinham os endereços de entrada com essa característica. Por fim, necessitou-se retirar todas as carteiras de trocos e seus valores, para isso, fez-se um comparativo entre os endereços de entrada e os de saída, caso tiver algum endereço igual, foi pego o índice da carteira presente no endereços de saída e, com isso, retirou-se o valor do campo de valores de saída com o mesmo índice.

```
for document in collection.find({'addresses_senders': \
{'$ne': "Newly_generated_coins"}}):

    month = document['month']

    tx_ascii = str(bytes.fromhex(document['script'][4:]))
    tx_ascii = tx_ascii[2:len(tx_ascii) - 1]

    if "\\" not in tx_ascii[:3]:
        content = 'readable'
    else:
        content = 'not_readable'

    sum_values = 0
    senders = document['addresses_senders']
    receivers = document['addresses_receivers']
    values = document['values_outputs']

    for sender in senders:
        if sender in receivers:
            ind = receivers.index(sender)
            receivers.pop(ind)
            values.pop(ind)

    for value in values:
        sum_values += value

    if sum_values != 0.0:
        value_month_usd =\
        data_values_usd[f'{year}'][f'{month}']
        years_values[year]\
        [content].append(sum_values*value_month_usd)
```

```
t_contents = ['readable', 'not_readable']

sum_values_year_know =\
sum(years_values[year][t_contents[0]])
sum_values_year_unknown =\
sum(years_values[year][t_contents[1]])

med_know =\
sum_values_year_know / len(years_values[year][t_contents[0]])
med_unknown =\
sum_values_year_unknown / len(years_values[year][t_contents[1]])

average_readable.append(med_know * 10 ** -8)
average_not_readable.append(med_unknown * 10 ** -8)
```

Listing 4.3 – Código responsável pela conversão para dólares e cálculo das médias anuais das transações realizadas entre 2014 até 2019.

Como pode ser observado no trecho de código 4.3, primeiramente o *for* itera sobre todos os dados da *collection* sendo vista, excluindo todas as transações que tiverem a *string Newly_generated_coins*. Depois será visto se o *script OP_RETURN* é legível ou não, seguido da remoção dos valores no campo de saída destinados a carteiras presentes no campo de entrada. Então será feita a soma e em seguida da conversão em dólares para a cotação do mês sendo visto, vale relembrar que os valores somados estavam em *satoshis*. Com as operações anteriores realizadas, o último passo sera calcular a média do ano sendo visto e realizar a multiplicação de 10^{-8} para obtermos os valores em dólares. Os valores em dólares usados para conversão estão armazenados no dicionário *data_values_usd*, onde a chave *year* possibilitará o acesso a um dicionário de com os meses, as chaves serão números de 1 até 12.

4.3 Análise de conteúdos - Protocolos

A próxima etapa da análise envolveu identificar dos conteúdos legíveis quais são os mais utilizados e descobrir se há alguma associação com alguma entidade ou grupo. Com auxílio da função *findall* do *regex* da linguagem Python, foi possível separar cada conteúdo. Para isso, foi usado como argumento no *findall* uma expressão regular que identifica três caracteres distintos. Escolheu-se esta quantidade de caracteres após uma análise preliminar nas transações com conteúdos legíveis, notou-se que muitos protocolos de serviços diversificados são identificados usando, ao menos, três caracteres. Com isso,

possibilitou encontrar 84.738 conteúdos com alta possibilidade de esta sendo usado por alguma entidade que utiliza o *opcode OP_RETURN* com um protocolo ou com outro tipo de finalidade. A Tabela 4 contém alguns dos serviços que foram encontrados durante esta inspeção.

```
def insert_dict(key):
    ascii_content[key] = {
        'count': 1,
    }

tx_ascii = document['ascii']
if len(tx_ascii) > 0:
    for ascii_script in tx_ascii:
        matches = re.findall('\\b\\w{3,}',\
        ascii_script,re.DOTALL)
        if ascii_content.get(matches[0]):
            ascii_content[matches[0]]['count'] += 1
        else:
            insert_dict(matches[0])
```

Listing 4.4 – Este código foi usado para identificar todos os conteúdos legíveis que tenham os três primeiros caracteres distintos.

O código 4.4 foi utilizado na etapa de análise de conteúdos do *OP_RETURN*. O dicionário *document* possuirá os dados de um dos anos sendo analisados, 2014 até 2021. Logo, primeiramente será pego o vetor *ascii* da transação e caso ela tenha algum conteúdo legível, prosseguirá para o próximo passo. Cada conteúdo legível será passado para a função *findall* discutida anteriormente, e o que for achado será inserido em um dicionário, sendo a chave a *string* encontrada no *findall*.

Com os dados mostrados pela Tabela 4, pode ser observada a grande diversidade de serviços que utilizam o *OP_RETURN*, como serviços voltados a direitos autorais, criptomoedas e serviços relacionados a mineração de criptomoedas, e assim por diante. Logo percebe-se os benefícios que tal funcionalidade traz, proporcionando maiores utilidades para a *bitcoin* e a valorizando.

Um gráfico de dispersão foi construído para analisar a intensidade de aparições que alguns dos protocolos encontrados tiveram durante os anos de 2014 até 2021. Esta análise foi bem simples de ser feita, passou-se por cada transação feita em ordem diária de cada ano e pegou todos os conteúdos legíveis inseridos na transação. Todos esses dados foram passados para arquivos de textos, onde serão abertos e inseridos em um vetor que será usado na análise. Abaixo está o código da criação do gráfico.

Serviço	Descrição	Identificador
Safex	Usado para *Ecommerce*.	*Safex1*, *Safex2*
Photector (antigo *Peirmobile*)	Serviço de seguro de transportadoras.	*PHOTECTOR*, *PEIRMOBILE*
Mathwallet	Carteira digital com suporte diversas criptomoedas.	*mathwallet*
Babel Finance	Plataforma financeira de criptomoedas que disponibiliza serviços diversificados.	BabelBank_bitfaith_No
Nodeasy	Empresa voltada a análises e monitoramento de *Masternodes*. Além disso, oferecem serviços de desenvolvimento de *Masternode*.	*nodeasy*
OriginalMy	Serviço voltado ao ramo de autenticação com certificados e assinaturas digitais.	*ORIGMY*
Omni Layer	Possibilita a criação de ativos através da *Bitcoin*.	*omni*
po.et	Não há informações sobre o do que se trata.	*POET*
RSK	Utiliza a tecnica de *Merge Mining*;	*RSKBLOCK*
POTX	Não há informações sobre o do que se trata.	*POTX*
ChainX	Usado para identificar as transações relacionadas a *ChainX*.	*ChainX*
Bernstein	Empresa voltada ao ramo de direitos autorais.	*BERNSTEIN*
THOR	Utilizado na funcionalidade da *THORChain*.	*THOR*
DOCPROOF	Voltado ao ramo de direitos autorais.	*DOCPROOF*

Tabela 4 – Serviços que utilizam o *OP_RETURN* com a sua funcionalidade e identificador

```
years = [2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021]

op_return_by_year = []

for year in years:
    with open(f'readable-content-{year}', 'r') as file:
        content_op_return = file.read().split('\n')
    op_return_by_year.append(content_op_return)
```

```
target_words = ['CNTRPRTY', 'PX2', 'omni', 'RSKBLOCK', 'PHOTECTOR',
'PEIRMOBILE', 'ion', 'THOR', 'Safex1', 'Safex2', 'IDEA', 'POTX', 'POR',
'BERNSTEIN', 'PROOFSTACK', 'PPk', 'Bitzlato', 'DOCPROOF']

visualizer = DispersionPlot(target_words,\
    title="Frequencia de protocolos")
visualizer.fit(op_return_by_year)
visualizer.show()
```

Listing 4.5 – O código gera um gráfico de dispersão, com o objetivo de analisar a frequência dos protocolos encontrados.

No código 4.5 o vetor *target_words* terá os protocolos que serão analisados e o *content_op_return* terá os conteúdos das transações começando com o ano de 2014 e terminando no ano de 2021. A Figura 18 é o gráfico resultante do código mostrado.

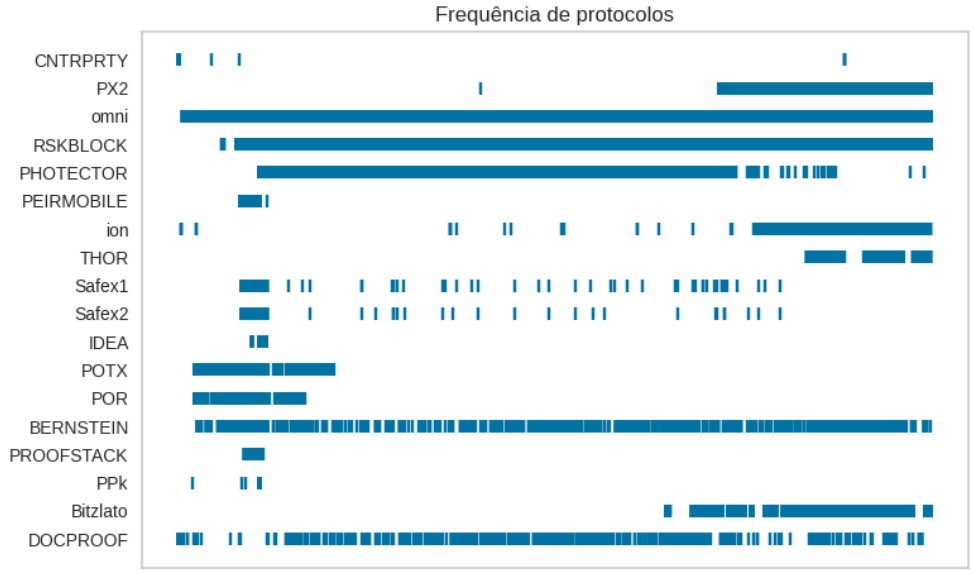


Figura 18 – Gráfico de dispersão da frequência de protocolos usados durante 2014 até 2021.

Com a Figura 18 nota-se que alguns protocolos estiveram presentes desde que o *OP_RETURN* foi padronizado, em 2014. Além disso, observa-se que muitos protocolos deixaram de existir após um certo tempo, neste caso foi a *Safex*, que depois de um tempo não foi possível encontrar mais nenhum de seus identificadores. O mais predominante é o *Omni*, que foi apontado por outros artigos sobre a sua predominância na rede *Bitcoin*. E, também conseguiu encontrar protocolos mais novos, como o *THOR* e o *Bitzlato*.

4.4 Conteúdos que vão além de protocolos

Além dos protocolos, também foram encontrados *links* para *downloads* via *torrents* que infringem os direitos autorais de jogos eletrônicos e álbuns de bandas. Além disso, também foram encontradas diversas mensagens arbitrárias como comemorações de ano novo, marcações de algum evento tanto relacionado a *Bitcoins* quanto globais e *memes*. Foi possível encontrar diversas carteiras que, de forma periódica, realizam transações com elas mesmas repassando uma quantidade pequena de *Bitcoin*. O conteúdo encontrado no *OP_RETURN* destas transações não é legível e sempre se encontra quantidades altas de *Bitcoins* repassadas a estas carteiras. A Figura 19 ilustra uma destas carteiras, com uma breve demonstração da sua atividade periódica, que neste caso durou aproximadamente 3 anos.

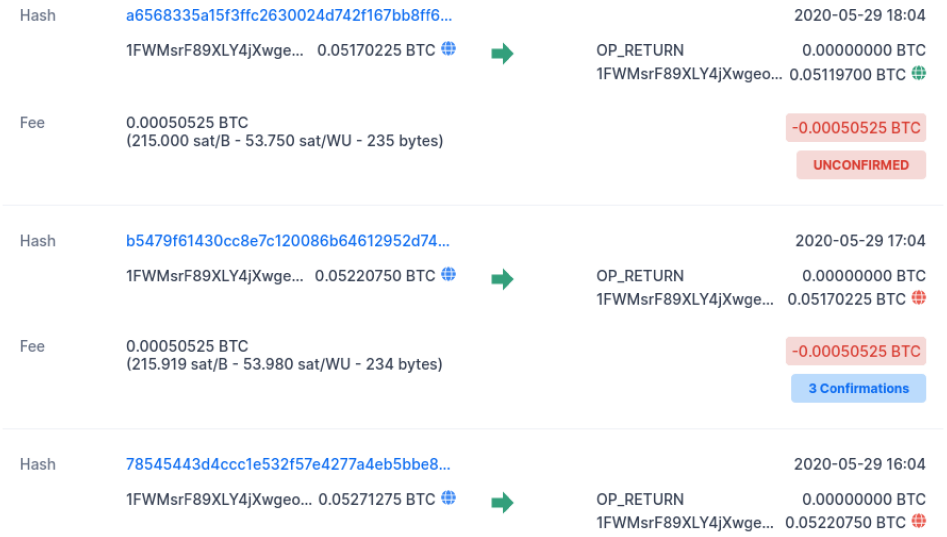


Figura 19 – Imagem com atividade de uma carteira. Imagem retirada do site blockchain.com.

4.5 Análise da interação dos usuários que utilizaram *OP_RETURN* na rede *Bitcoin*

Para identificar como as carteiras interagem entre si na rede *Bitcoin* foi feito um agrupamento das carteiras. Carteiras são agrupadas quando um endereço de entrada de uma transação é igual ao de entrada de outra transação e foram apenas incluídas as carteiras que realizaram mais de uma transação. Os agrupamentos foram classificados como: isolados, quando uma carteira envia *Bitcoin* para ela mesma; Uma ou duas saídas, será para carteiras que enviaram para uma ou mais carteiras diferentes; Múltiplas saídas, é quando há envio para três ou mais carteiras. A Figura 20 mostra um exemplo de comportamento das transações que usam *OP_RETURN*. Os vértices destacados em vermelho

são as carteiras de entradas e os azuis são os de saída. Na Tabela 5 encontra-se o número de transações que foram agrupadas e a característica deste agrupamento.

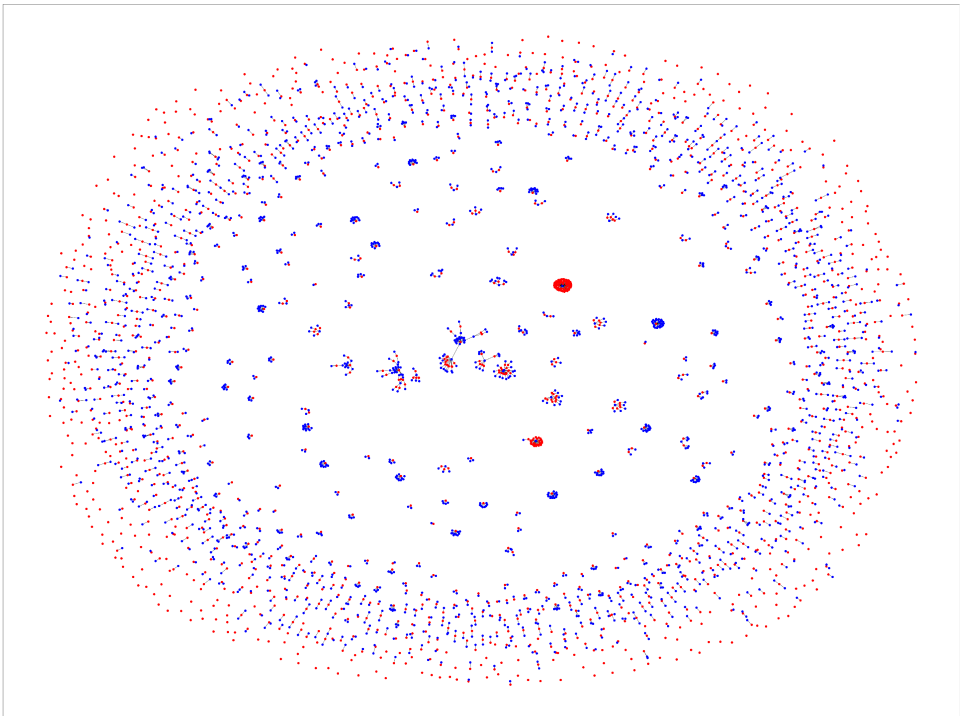


Figura 20 – Grafo com a interações das carteiras no ano de 2014. Os vértices vermelhos que não possuem nenhuma conectividade com um vértice azul são classificados como isolados.

Ano	Isolado	uma ou duas saídas	múltiplas saídas	total
2014	402	1143	155	1700
2015	1310	15636	1756	18702
2016	2476	50222	3345	56043
2017	7998	117364	4826	130188

Tabela 5 – Informações anuais das clusterizações. Um agrupamento isolado é quando a carteira de entrada é a mesma que a de saída em uma transação, já de uma ou duas e múltiplas saídas será quando os endereços forem diferentes nos campos de entrada e de saída.

Na Tabela 5 observa-se que grande parte das atividades dos agrupamentos está em transações com uma ou duas saídas, mostrando que as atividades dos usuários do *OP_RETURN* são mais isoladas e direcionadas a um ou dois grupos. Além disso, com os dados da Tabela 5 foi possível identificar que apenas uma pequena parcela dos usuários

enviou *Bitcoins* para eles mesmos, fato comprovado ao observar os nós isolados vermelhos, vértices cujos endereços de entradas são os mesmos do que os de saída.

4.6 Resumo dos resultados

A Tabela 2 evidencia o grande crescimento que o uso do *OP_RETURN* teve ao longo oito anos, além disso notou-se que os conteúdos não legíveis são os mais predominantes durante estes períodos. A análise de valores permitiu concluir que em 2015 até 2018, tanto para conteúdos legíveis e não legíveis, não houve grandes divergências notáveis na média de valores usados nas transações. Entretanto, em 2014 a média de conteúdos não legíveis chegou a alcançar uma média aproximada de 5 mil dólares, e a maior divergência ocorreu em 2019, com os conteúdos legíveis chegando a ultrapassar uma média de 20 mil dólares.

Foi possível identificar diversos protocolos com finalidades distintas, desde protocolos voltados a seguradoras e direitos autorais até protocolos utilizados por outras *blockchains*, como é o caso da *THORChain*. A frequência de uso desses protocolos é visivel graças ao gráfico de dispersão da Figura 18, onde pode ser visto o uso de alguns protocolos de 2014 até 2021.

Por fim, diversas carteiras com atividades peculiares foram encontradas, e uma delas é ilustrada na Figura 19. A análise de como as carteiras que utilizaram o *OP_RETURN* pode ser visto na Tabela 5 e no grafo ilustrado na Figura 20, com estes dados observa-se que a categoria mais predominante é a de nós com uma ou duas saídas, ou seja, ocorreram mais transações onde tinha-se uma ou duas carteiras de saída.

5 Conclusão

O objetivo proposto pelo trabalho foi analisar e descobrir como o *opcode OP_RETURN* estava sendo utilizado durante os anos de 2014 até 2021, e para isso ser realizado foram categorizadas as transações em legíveis e não legíveis. Depois realizou-se análise sobre os valores gastos nas transações, análises de conteúdos no qual permitiu encontrar diversos protocolos e conteúdos diversificados. E, por fim foi possível com os agrupamentos identificar como os usuários da rede *Bitcoin*, que usaram o *OP_RETURN*, se interagiam.

Este trabalho conseguiu alcançar os principais objetivos ao apresentar uma análise detalhada sobre o uso do *OP_RETURN* entre os anos de 2014 até 2021. Entretanto, por conta da limitação dos dados não foi possível obter os valores gastos em dólares em 2020 e 2021, além disso teve-se a impossibilidade de entendimento de grande parte dos conteúdos inseridos com *OP_RETURN*. Outra limitação que barrou a coleta de mais resultados foi a demora nos agrupamentos de carteiras, onde foi viável fazer apenas os agrupamentos de quatro dos oito anos analisados.

Em um trabalho futuro sugere-se que sejam feitas análises de conteúdos de outros meios de inserção de dados arbitrários nas transações, como o *unspendable outputs*, que utiliza o campo de *outputs* de uma transação, e traçar um comparativo com os dados inseridos através do *OP_RETURN*, a fim de observar as semelhanças e diferenças entre os tipos de dados encontrados.

6 Anexo

Link para o repositório no *Github* que contém os códigos usados neste trabalho. Sanches (2022)

Referências

AGNER, M. Bitcoin para Programadores. *GitHub*, 2016. Disponível em: <https://btcparaprogramadores.marcoagner.org/transacoes.html>. Citado na página 9.

ALI, S. T. et al. Zombiecoin 2.0: managing next-generation botnets using bitcoin. *International Journal of Information Security*, Springer, v. 17, n. 4, p. 411–422, 2018. Citado 3 vezes nas páginas 10, 18 e 19.

ANTONOPOULOS, A. M. *Mastering Bitcoin: Unlocking Digital Cryptocurrencies - Andreas M. Antonopoulos - Google Books*. [S.l.: s.n.], 2014. ISBN 978-1-449-37404-4. Citado 2 vezes nas páginas 9 e 12.

ANTONOPOULOS, A. M. *Mastering Bitcoin: Programming the Open Blockchain - Andreas M. Antonopoulos*. [S.l.: s.n.], 2017. ISBN 978-1-449-37404-4. Citado na página 13.

BARTOLETTI, M.; BELLOMY, B.; POMPIANU, L. A journey into bitcoin metadata. *Journal of Grid Computing*, Springer, v. 17, n. 1, p. 3–22, 2019. Citado 2 vezes nas páginas 16 e 19.

BARTOLETTI, M.; POMPIANU, L. An analysis of bitcoin op_return metadata. In: SPRINGER. *International Conference on Financial Cryptography and Data Security*. [S.l.], 2017. p. 218–230. Citado 3 vezes nas páginas 10, 18 e 19.

BITCOIN. *Referencial scripts opcodes da bitcoin*. 2022. Ultimo acesso em 11 de novembro de 2020. Disponível em: <https://en.bitcoin.it/wiki/Script#Opcodes>. Citado na página 17.

BLOCKCHAIN. *Website Blockchain*. 2022. Ultimo acesso em 11 de novembro de 2020. Disponível em: <https://www.blockchain.com/>. Citado 9 vezes nas páginas 4, 10, 14, 15, 16, 17, 22, 25 e 28.

BLOCKCHAIR. *blockchair.com*. 2022. <https://blockchair.com/dumps>. Citado 2 vezes nas páginas 21 e 22.

BÖCK, L. et al. Assessing the threat of blockchain-based botnets. In: IEEE. *2019 APWG Symposium on Electronic Crime Research (eCrime)*. [S.l.], 2019. p. 1–11. Citado 2 vezes nas páginas 18 e 19.

COINSECRETS. *Website coinsecrets*. 2020. Ultimo acesso em 29 de maio de 2020. Disponível em: <http://coinsecrets.org//>. Citado 4 vezes nas páginas 4, 16, 17 e 21.

FAISAL, T.; COURTOIS, N.; SERGUIEVA, A. The evolution of embedding metadata in blockchain transactions. In: IEEE. *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2018. p. 1–9. Citado 2 vezes nas páginas 18 e 19.

MATZUTT, R. et al. A quantitative analysis of the impact of arbitrary blockchain content on bitcoin. In: SPRINGER. *International Conference on Financial Cryptography and Data Security*. [S.l.], 2018. p. 420–438. Citado 2 vezes nas páginas 18 e 19.

MOLS, J.; VASILOMANOLAKIS, E. Visualizing the bitcoin's op_return operator. In: *Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*. [S.l.: s.n.], 2020. p. 305–306. Citado na página 19.

NAKAMOTO, S. Bitcoin whitepaper. *URL: https://bitcoin. org/bitcoin. pdf-(: 17.07. 2019)*, 2008. Citado 4 vezes nas páginas 4, 9, 12 e 13.

SANCHES, L. B. *repositorio github*. 2022. <https://github.com/benitoSan/codigos-tcc>. Citado na página 40.

STREHLE, E.; STEINMETZ, F. Dominating op returns: The impact of omni and veriblock on bitcoin. *Blockchain Research Lab*, 2020. Citado na página 19.