The "Bitcoin Generator" Scam

Article · April 2022

DOI: 10.1016/j.bcra.2022.100084

CITATION
1

READS
1,341

3 authors, including:

Emad Badawi
University of Ottawa
7 PUBLICATIONS **47** CITATIONS
SEE PROFILE

Iosif-Viorel Onut
IBM
42 PUBLICATIONS **471** CITATIONS
SEE PROFILE

Blockchain: Research and Applications 3 (2022) 100084

Contents lists available at ScienceDirect

Blockchain: Research and Applications

journal homepage: www.journals.elsevier.com/blockchain-research-and-applications

Research Article

The "Bitcoin Generator" Scam

Emad Badawi [a,*], Guy-Vincent Jourdan [a], Iosif-Viorel Onut [b]

[a] Faculty of Engineering, University of Ottawa, Ottawa, ON K1N 6N5, Canada
[b] IBM Centre for Advanced Studies, Ottawa, ON K1R 7Y6, Canada

ARTICLE INFO

Keywords:
Cryptocurrency
Scam analysis
Cyberattack
Fraud detection
Bitcoin
Blockchain analysis
Data mining

ABSTRACT

The "Bitcoin Generator Scam" (BGS) is a cyberattack in which scammers promise to provide victims with free cryptocurrencies in exchange for a small mining fee. In this paper, we present a data-driven system to detect, track, and analyze the BGS. It works as follows: we first formulate search queries related to BGS and use search engines to find potential instances of the scam. We then use a crawler to access these pages and a classifier to differentiate actual scam instances from benign pages. Last, we automatically monitor the BGS instances to extract the cryptocurrency addresses used in the scam. A unique feature of our system is that it proactively searches for and detects the scam pages. Thus, we can find addresses that have not yet received any transactions.

Our data collection project spanned 16 months, from November 2019 to February 2021. We uncovered more than 8,000 cryptocurrency addresses directly associated with the scam, hosted on over 1,000 domains. Overall, these addresses have received around 8.7 million USD, with an average of 49.24 USD per transaction.

Over 70% of the active addresses that we are capturing are detected **before** they receive any transactions, that is, before anyone is victimized. We also present some post-processing analysis of the dataset that we have captured to aggregate attacks that can be reasonably confidently linked to the same attacker or group.

Our system is one of the first academic feeds to the APWG eCrime Exchange database. It has been actively and automatically feeding the database since November 2020.

1. Introduction

In recent years, the use of cryptocurrencies as an investment platform has gained popularity [1]. At the time of writing, there are 17,343 different cryptocurrencies, with a market capitalization of around 1.89 trillion USD [2]. The most popular cryptocurrencies are Bitcoin and Ethereum, which have market capitalizations of around 783 billion USD and 359 billion USD, respectively [2].

Bitcoin [3] is a decentralized cryptocurrency that became popular in 2009. It is a peer-to-peer electronic currency that does not need the involvement of a trusted authority such as a central bank or an administrator to be exchanged between users [3–5]. Bitcoin has two key features: pseudo-anonymity and transparency [4–6]. It is transparent because the transactions are publicly available in a decentralized ledger called a blockchain. Bitcoin pseudo-anonymity comes from the fact that users use pseudonyms (addresses). These addresses are computed from the user's public key, and they are not directly related to individuals. There is no limit on the number of addresses a user can generate. Thus, users can generate unique addresses for each transaction. This, in turn, creates an additional layer that prevents the addresses from being linked to a specific owner, which ultimately increases privacy [3].

Cybercriminals have leveraged Bitcoin's pseudo-anonymity in their attacks. According to a CipherTrace Spring-2020 report[1], the value of thefts, hacks, and scams more than doubled in 2019 when compared to 2018, and was more than 230 times the value of 2017. In 2019 alone, more than 4.52 billion USD were stolen from cryptocurrency exchanges and users. However, although 2020 had the second-highest value in crypto-crime ever, the crypto-crime value had a significant drop of 57% compared to 2019, going down from 4.5 billion USD to 2.1 billion USD.

As shown in Fig. 1, in 2020, losses from cryptocurrency exchanges, hacks, and thefts reached 516 million USD, while the majority of the losses (1.642 billion USD) were due to fraud and misappropriation of

* Corresponding author.
E-mail addresses: ebada090@uottawa.ca (E. Badawi), gjourdan@uottawa.ca (G.-V. Jourdan), vioonut@ca.ibm.com (I.-V. Onut).

[1] https://ciphertrace.com/2020-year-end-cryptocurrency-crime-and-anti-money-laundering-report/, https://ciphertrace.com/cryptocurrency-crime-and-anti-money-laundering-report-may-2021/.

https://doi.org/10.1016/j.bcra.2022.100084
Received 12 August 2021; Received in revised form 8 February 2022; Accepted 6 April 2022

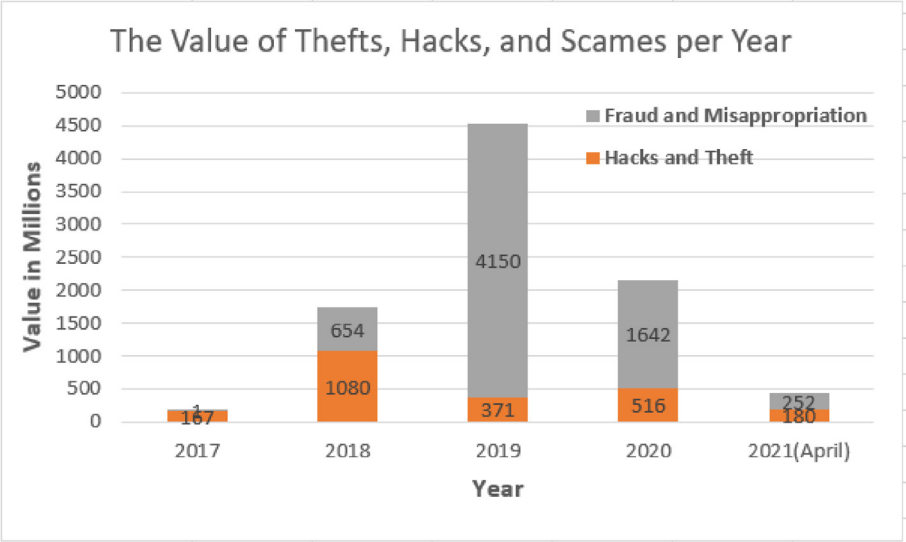


Fig. 1. 2021 cryptocurrency anti-money laundering report (reproduced from CipherTrace report[1]).

funds. For example, KuCoin, the Singapore-headquartered digital asset exchange, was one of the targets of the hacks and theft attacks in 2020. In September 2022, the exchange announced an unauthorized transfer of Bitcoin and Ethereum tokens to an unknown wallet, affecting around 150 million USD in users' funds. On the other hand, the "WoToken" Ponzi scheme defrauded investors of over 1 billion USD during the period its scam was functioning[1].

Lendf.me, a decentralized lending protocol operated by a Chinese DeFi upstart, dForce, was one of the targets of the hacks and theft attacks in 2020. On April 19, 2022, 25 million USD worth of cryptocurrency were stolen from *Lendf.me*. On the other hand, the "EOS Ecosystem" wallet defrauded investors of 52 million USD in a Ponzi scheme by enticing investors with promises of favorable returns. DeFi-related hacks and fraud grow quarter over quarter. In just the first 4 months of 2021, the value of DeFi-related hacks and fraud had already surpassed 2020's all-time high[1].

Cybercriminal attacks using cryptocurrencies take many forms. One of the popular examples of these attacks is "High Yield Investment Programs" (HYIP) [4,5,7,8]. In HYIP, scammers promise investors a high-interest rate, e.g., more than 1%–2% per day [4]. Perhaps the most famous HYIP scammer was Charles Ponzi; in the early 1920s, Ponzi claimed to run an arbitrage in which the investors were promised a 50% profit within 45 days or a 100% profit within 90 days. Because of Ponzi, HYIPs are sometimes called Ponzi schemes [4].

Ransomware [9–11] and money laundering (ML) [12,13] are other common examples of unlawful activities often using cryptocurrencies. Ransomware is a form of malware that locks and encrypts a victim's files until a ransom is paid [9]. Recently, cybercriminals seized and shut down the City of Riviera Beach's computer systems, forcing the officials to agree to pay 65 Bitcoins worth 600,000 USD at the time. The resulting outage forced the local police and fire departments to record hundreds of 911 calls on paper [14].

Money laundering describes the process of disguising the sources of illegal profits generated by criminal activity. It aims to hide the link between the original criminal activities and the corresponding funds by passing the money through a complex sequence of banking transfers or commercial transactions [12].

The current state of the art for Bitcoin scam detection usually relies on extracting features from the transaction histories on the blockchain to train a classification model [4–7]. The classification model is trained on features such as the ratio of received/sent transactions to all transactions, the address lifetime, the frequency of transactions, or the "payback" ratio, which is the ratio of addresses that appears in both the input and output sides of address transactions. To acquire these addresses, the authors collected the addresses manually by searching on Bitcoin discussion forums such as Bitcointalk.org [7], or they used semi-automated web crawls of the same forums, followed by manual inspection and address collection [4–6].

However, the number of transactions recorded on the blockchain is increasing over time[2]. This makes it difficult and time-consuming to extract meaningful patterns that can be used in fraud detection [7].

In this paper, we study a social engineering attack that emerged with the rise of cryptocurrencies. We call this attack the Bitcoin Generator Scam (BGS) [15]. Usually, BGS starts when a victim searches for an easy profit using search engines, streaming sites, social media, blogs, etc. For example, the first step in Fig. 2 shows the result of searching for the "free Bitcoin generator online". The search results may directly contain BGS instances (Fig. 2 step 2). In other cases, the search results link to pages that have links to a BGS instance.

We call the BGS instances "generators". These generators are carefully designed web pages that attempt to convey to the victim an impression of the advanced technical abilities of the attacker and a large, satisfied user base for the scam. Some generators display a fake chat box and a pop-up showing the number of claimed current users and the number of mined cryptocurrencies they supposedly gained.

In BGS, the attackers claim that they own a high-speed mining machine or can hack the blockchain ledger and that they can provide the victim with free cryptocurrencies. Once a BGS instance, like the one shown in step two of Fig. 2, is accessed, the victim is asked to provide the number of coins they want to mine and the cryptocurrency address in which the mined coins will be deposited. Once the information is provided, the scammer pretends to perform some "hacking" (Fig. 2, step 3). After that, a message is displayed claiming that the hack was successful, and the victim is then asked to pay a mining fee to collect the funds (Fig. 2 step 4). In many cases, the value of the fees are fixed. In other cases, the attacker promises that the victim will receive some multiple of the amount they provide.

In other variations of the attack, rather than asking for a mining fee, the scammers ask the victims to either complete one or more tasks or download and install a mining executable file to complete the mining process. In the former case, after the success message is displayed, the victim is invited to a "verification" step. During this verification process,

[2] Over 650 million transactions at the time of writing: https://www.blockchain.com/charts/n-transactions-total.

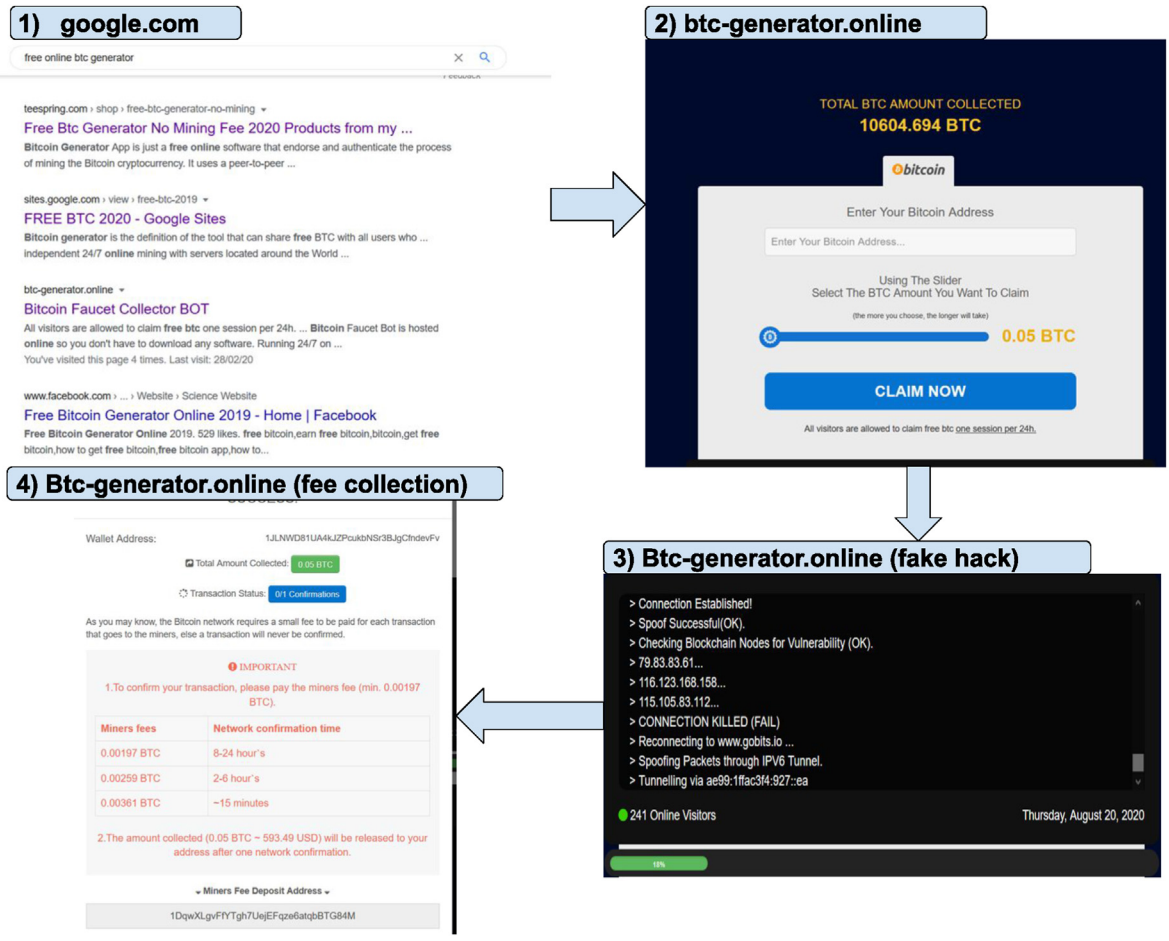


Fig. 2. An example of the Bitcoin generator scam attack.

a screen is shown to the users asking them to complete one or more "offers". These so-called offers include, but are not limited to, filling out "market research" forms, clicking through endless "surveys", getting the victims to subscribe to questionable services, collecting personal information, installing suspicious executable files on their machines, etc. In the latter case, the executable mining files were reported as harmful by Virus total[3].

Some researchers (e.g., Ref. [16]) characterize the Ponzi schemes by their pyramidal structure, in which the payout to existing investors uses funds from new investors. By this definition, BGS does not fall under the Ponzi scheme rubric since most BGS do not require investors to enroll new investors, and as discussed in Section 4.2, we usually do not find any evidence of payout at all. However, some researchers (e.g., Refs. [8,17]) characterize Ponzi schemes by their high rates of return, and BGS certainly falls under this category, with an advertised return rate in the range of 100% in 24 h, or even more.

In this paper, we extend and update our previous work in Ref. [15]. We have incorporated new search queries using the words and phrases that appear in the BGS corpus. We have updated our results using the newly discovered BGS instances. Overall, we have discovered more than 1,000 scam domains and more than 3,000 Bitcoin addresses with at least one transaction associated with them. These addresses have received around 8.7 million USD, with an average of 49 dollars per transaction. Moreover, we have reported two more variations of the BGS attack. The first one requires the installation of an executable mining file on the victim's machine. We collected 12 files that were reported as harmful by Virus total. The second attack asks the victim to complete one or more malicious tasks. Additionally, we looked at the cryptocurrency addresses reused in BGS and other types of scams. Finally, we used various features to cluster the BGS addresses into campaigns controlled by the same scammers.

Our main contributions are the following:

- We provide insight into BGS, a new type of scam that targets cryptocurrency users. Although the current research covers different types of attacks that target cryptocurrencies, none of them has investigated the BGS attack. The most closely related studies are those studying the HYIP schemes. Many of the BGS instances promise a high return rate, which is one of the definitions used to characterize HYIP schemes [8, 17]. However, our analysis showed that the BGS instances do not pay back the victims and thus are not HYIP attacks by these definitions. In this study, we collected hundreds of BGS instances and thousands of BGS addresses, and provided an in-depth analysis and insights into this scam scheme.
- Although many anti-scam studies have been proposed, only a few have published their datasets. When datasets are published, often only the scam URLs are provided, and these URLs are usually no longer available at the time of publication. This makes it difficult for subsequent studies to compare their methods with previous work.

In this paper, we compile and publish a large BGS dataset[4], including more than 1,000 BGS URLs that were collected using our live crawler and public sources, such as the Internet Archive[5]. In addition to the URLs of

[3] https://www.virustotal.com/gui/.

[4] https://ssrg.eecs.uottawa.ca/bgsextended/.

[5] https://web.archive.org/.

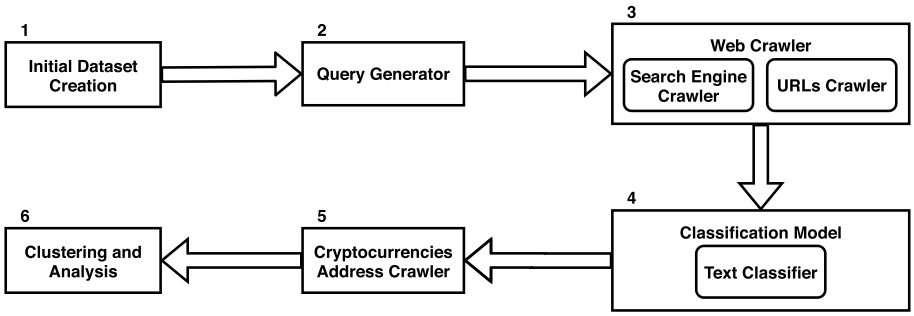


Fig. 3. Bitcoin generator scam detection and analysis model.

scam attacks, we also provide the Document Object Model (DOM) of scam pages, allowing others to compare their methods to ours. Moreover, we publish more than 8,000 Bitcoin addresses used in these attacks. As far as we know, this is one of the largest, if not the largest, of all Bitcoin scam databases published by academic research. Finally, we provide more than 140 scam addresses that are associated with other cryptocurrencies, such as Ethereum, Bitcoin Cash, and Litecoin.

- We developed a new research direction to detect web-based Bitcoin scam attacks. The state of the art in academic work on Bitcoin scam detection is usually based on some manual collection of addresses involved in the scam. The starting point could be a manual search on a forum in which the attack is being discussed, e.g., Bitcointalk.org [7], or it could be by a semi-automated crawl of that same forum, followed by a manual address collection [4–6]. Furthermore, some researchers use "multiplier" techniques, such as the multi-input heuristic clustering algorithm [18], to collect the bulk of addresses controlled by the same scammers [7]. Once scam addresses have been collected, their transaction histories are used to extract distinguishing features and separate benign addresses from scam addresses [4–7,19]. These features are then used to train a classifier [4,7].

In this study, we do not base our analysis on previously reported campaigns only. Instead, we search for new, previously unreported instances. What is more, we do not use existing transactions in the detection phase, which allows us to find addresses that do not have any payments yet. Using our approach, we have detected more than 70% of the current active scam addresses before they received any transactions, which is impossible using traditional detection methods. Relying on blockchain transactions to detect scams inherently implies that the detection occurs too late after victims have already fallen for the scam. Our approach opens the door to shutting down an attack before anyone is victimized.

- We expanded our analysis to provide more insight into scam instances and deduced how the campaign was operated by the same scammer. We used a variety of domain-related and address-related features and identifiers to connect different scam attacks and link them to the same scammer. Our results showed that two scam clusters have received around 5 million USD, which is more than half of the total funds received by the scam addresses.

The remainder of this paper is structured as follows. After this introduction, in Section 2, we detail our methodology. In section 3, we report some basic numbers obtained during our crawling period. In Section 4, we carry out various analyses and discuss the results. In Section 5, we present other variations of the BGS attack. In Section 6, we present our multi-level clustering technique. A literature review is provided in Section 8. In Section 9, we discuss some of the main limitations of our model and possible future enhancement and analysis. Finally, we conclude in Section 10.

2. Methodology

In this section, we describe a data-driven approach to detecting, tracking, and analyzing BGS. Fig. 3 describes our complete system, which includes six modules:

1. **Initial dataset creation.** Initially, our system depends on a manual search for scam pages to obtain a representative dataset on which to train our model. This also helps to get an initial broad understanding of the scam and provides the source needed to automate effective search queries related to the scam.
2. **Search query generator.** This module generates the keywords that are likely to be used in the scam pages.
3. **Web crawler.** This module uses the search engines to search for scam pages using the previous queries as a seed.
4. **Classification model.** This module categorizes the crawled pages as either "scam" or "clean" pages based on their text.
5. **Cryptocurrencies addresses crawler.** This module interacts with the scam pages and provides the requested information needed to detect the scam addresses. We also submit the scam addresses to the Anti-Phishing Work Group (APWG) data warehouse.
6. **Clustering and analysis.** Our system's final module is for analyzing the data to identify similarities and cluster-related scam instances.

2.1. Dataset construction

We start our work by collecting an initial set of BGS instances to train our classification model and extract search queries to search for more scam instances using our web crawler. We use various techniques to collect this initial dataset:

1. **Search Engines:** in many cases, it is challenging to find a reliable source of labeled data to run the experiments on. In such cases, different search engines can be used to collect and label an initial training dataset manually. We manually searched for BGS instances on Google. We used several search queries related to the scam, such as "online Bitcoin generator", "generator free Bitcoin", and "online Bitcoin hack tool". Our search identified an initial set of 52 BGS instances. We also obtained 30 new search queries using Google's automatic "related search" suggestions while doing this initial collection. This gave us our initial set of queries for starting our automated web crawl.
2. **Third parties and blacklists**: many third-party companies and blacklists collect scam datasets that researchers can use in their analysis. For example, Yin and Vatrapu [20] used a dataset provided by Chainalysis.com, and Razali and Shariff [21] used the Nocoin blacklist[6] in their analysis. In our work, we used the site Bitcoin.fr [22], which

[6] https://github.com/hoshsadiq/adblock-nocoin-list.

contains a list of Bitcoin and cryptocurrency scam domains. The list is a collection of several scam lists, including adcfrance.fr, the House of Bitcoin, CryptoFR, badBitcoin.org, and scamBitcoin.com, as well as the testimonies of the site users. At the time of crawling, the list contained 6,230 domains.

3. We also used CuteStat.com, which is a website that collects information related to websites, domains, hosts, IP addresses, usage reports, etc. One of the services that this website provides is a list of up to 100 domains that have content "related" to the search we performed. We used this service to collect 610 new domains that have content related to the search queries collected in step one.
4. Finally, we used the **Internet Archive** [23]**:** this is a digital library that provides a large collection of free digitized materials, including software applications/games, internet sites, movies/videos, and millions of books. We used the Internet Archive to collect thousands of snapshots for the set of domains that we collected data from CuteStat .com and Bitcoin.fr.
5. **Identifying BGS instances:** since the Internet Archive contains thousands of snapshots and we could not manually check all of them, we filtered the snapshots and only considered the snapshots that contain a Bitcoin address in HTML. This reduced the number of possible BGS domains to only 307, a number we could handle manually. We inspected these domain snapshots one-by-one and verified that 252 of these domains were indeed BGS domains. The other 55 domains were different types of scams, such as HYIPs and bogus charity sites.

Following these steps, we collected 304 pages as our initial set of BGS instances. We then manually inspected 400 randomly selected pages that we had collected, but not flagged, during the first week of operation. Of these 400 pages, 374 were benign pages, and 26 were new BGS instances. Therefore, our final dataset consists of 330 BGS pages (304+26), complemented with 330 benign pages randomly selected from the set of 374 pages we had.

2.2. Search query generator

Finding good search queries with a high likelihood of leading to scam pages is an important task. Srinivasan et al. [24] used a context-specific corpus to generate such queries, Kharraz et al. [25] used the Google Trends service, and Badawi et al. [26,27] used both techniques.

In our previous work, we used two techniques to generate 539 research queries:

- **Search Engines:** we started our work by collecting Google's automatic search suggestions as we manually searched for BGS. We then used these suggestions to create the first set of queries and perform an initial web crawl.
- **The "Keywords" meta tag:** as described in Section 2.1, we were able to collect and manually verify 330 BGS instances from our initial web crawling, as well as from a list of blacklisted domains [22], from the site cutestat.com, and from the Internet Archive. We extracted the contents of the "Keywords" meta-tag from these instances to augment our original queries. The "Keywords" meta-tag represents a comma-separated list of keywords that are relevant to the web page and are used to inform the search engines about its content[7].

In this paper, we further augmented our search queries by utilizing the scam-context-specific corpus. We inspected several BGS pages and found that scammers use specific words in the content of a BGS page, such as the name of the targeted currency and words that advertise the ability of the generator to hack the blockchain and provide the victim with the promised cryptocurrencies. For example, the words "Bitcoin", "btc", "tool", and "mining" were widely used in the scam pages.

Fig. 4. Word cloud based on the text contents of the gathered technical BGS (Bitcoin generator scam) pages.

We utilized this fact to generate more scam-related queries. We extracted a collection of words from our corpus. We found 834 words that have a frequency greater than, or equal to, ten. We selected the 157 words with the highest frequency that have a direct connection to BGS. We then generated our queries using the Markov assumption [28] to approximate *n*-gram probabilities. We generated our *n*-grams for *n* in a range from 3 to 7 (our experiments showed that 8-grams and up did not improve our results.), which gave us 527 *n*-grams, and we then manually selected 207 search queries from them.

Overall, we generated 157 new queries that we included in our set of search queries. Fig. 4 shows the most frequent words used in the BGS pages in the form of a word cloud, where the size of each word correlates with the number of times it appears in the corpus of BGS pages.

Our final query list contains 696 search queries[8].

2.3. Web crawler

The main purpose of this module is to use the previously identified search queries as a seed to perform a daily search for scam pages using search engines such as Google.com, Bing.com, and search.yahoo.com. For each query, the crawler visits the first and second pages returned by each search engine (that is, up to 20 search results). The crawler is based on Python Selenium[9] and ChromeDriver[10]. We then use Python beautifulsoup[11] and the CSS selectors to extract and crawl the URLs collected during the search. For the crawling process, we use a lightweight scripted headless browser, built using Python, by integrating ChromeDriver, Selenium, and Beautifulsoup. The crawler automatically collects data about the crawled URLs, including URL redirections, HTML content, and resources (scripts, CSS files, etc.).

2.4. Classification module

In the crawling process, most of the URLs returned by the search engines, or crawled by our system, are either benign pages having

[7] https://www.w3schools.com/tags/att_meta_name.asp.

[8] The complete list is available at https://ssrg.eecs.uottawa.ca/bgsextended/.

[9] https://selenium-python.readthedocs.io.

[10] http://chromedriver.chromium.org/.

[11] https://pypi.org/project/beautifulsoup4/.

nothing to do with scams or non-scam pages that link to one or more scam instances. Since we are building an automated system, we need a classifier to distinguish scam instances from genuine URLs automatically. To identify the BGS instances from the set of crawled pages, we used a text-based classification model. We tested five different classifiers from the Scikit-learn Python library [29] on our training set: Support Vector Classifier (SVC), Naive Bayes (NB), k-nearest neighbors (KNN), Random Forest (RF), and Multi-layer Perceptron (MLP).

To evaluate our classifiers, we used 10-fold cross-validation on the labeled dataset we prepared in Section 2.1. We used the five classifiers to classify the crawled pages based on the text as seen by the end-user. More precisely, we used the term frequency-inverse document frequency (TF-IDF) of the words displayed to the users to extract the training features.

Our classification model achieved good accuracy, with a True Positive Rate (TPR) above 98.7% and a False Positive Rate (FPR) lower than 1%. We show the results in Table 1. As can be seen, SVC and MLP achieved the highest F1 score of 98.92, followed by KNN at 97.9. The other classifiers also performed fairly well, with RF having the lowest F1 score.

In our analysis, True Negative (TN) is benign page classified as benign, and True Positive (TP) is scam page classified as a scam. False Negative (FN) is scam instance wrongly classified as benign, and False Positive (FP) is benign page wrongly classified as a scam. As usual, the F1 score is derived as follows:

$$F1 = 2 \times (\text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall})$$

where Precision=TP/(TP+FP) and Recall=TP/(TP+FN). The higher F1 is, the better.

We show the Receiver Operating Characteristic (ROC) curves of the 10-fold cross-validation runs for each classifier in Fig. 5. Fig. 5 shows that SVC and MLP achieve the best ROC means among all other classification models with an area under the curve (AUC) equal to 99%. The other classifiers also performed reasonably well, with RF having the lowest AUC value.

Fig. 6 illustrates the performance of the five classifiers. As can be seen, SVC and MLP perform generally better. Based on these results, we used the SVC classifier throughout our experiments.

We used our classifier on newly found pages for a week. We then randomly selected 100 pages classified as BGS instances and 100 pages classified as benign for manual verification. Our model correctly classified 197 of these 200 pages. One BGS instance was misclassified as benign, which yielded a true negative rate of 99%, and two benign pages were misclassified as BGS, which yielded a true positive rate of 98% (see Table 2).

As mentioned earlier, most of the pages we crawl are benign, leading to a class imbalance between BGS and benign pages. We did train our classifiers on a balanced dataset, but the imbalance makes it likely that benign pages are falsely classified as BGS (that is, our false positives are going to be an issue). To avoid this, in practice, we manually inspected and marked every newly detected BGS instance before submitting it to the scam database. Once marked, new addresses used by the pages will be automatically reported without additional manual checks. In our future work, we plan to implement a more efficient way to automatically filter false positive pages to run our crawler in an entirely automatic, unsupervised manner.

2.5. Interacting with BGS instances

This module aims to collect the cryptocurrency addresses that scammers use to collect funds from victims. For this purpose, we interacted with the BGS instances, provided the expected inputs, and followed the specific instructions in order to reach the final stage during which the scam address was provided (the fourth image of Fig. 2). Usually, the fake hacking process requires 5–10 min. During this time, the attacker typically displays a detailed "log" of the hacking process, which is supposed to occur in real-time. This log displays server IP addresses supposedly being hacked, bogus proxy servers names, the ledger's block in which the transaction is supposed to be added, etc.[12] (see, for example, Fig. 2 step 3). However, in some cases, we find the scam address in the HTML of the BGS immediately. For these pages, we collect the scam addresses without further interaction with the BGS instance.

Table 1
Results of a 10-fold cross-validation with five classifiers.

Classifier	Page Type	Classified Clean	Classified BGS	F1
SVC	clean	327	3	98.92
	gen	4	326	
MLP	clean	327	3	98.92
	gen	4	326	
RF	clean	329	1	95.9
	gen	25	305	
NB	clean	327	3	96.58
	gen	19	311	
KNN	clean	319	11	97.9
	gen	3	327	

Note: SVC: support vector classifier, MLP: multi-layer perceptron, RF: random forest, NB: naive bayes, KNN: k-nearest neighbors, BGS: Bitcoin generator scam.

Furthermore, in addition to the "live" crawling, we also crawl the Internet Archive [23] and urlscan.io to collect addresses that have been used by each scam instance in the past. urlscan.io is an online service that scans and analyzes websites. When a URL is submitted to urlscan.io, the website will automatically visit and collect data about the browsed URL, including domains and IPs contacted, the HTML content, a screen shot of the landing page, the resources (JavaScript, CSS, etc.) requested from those domains, technologies used, and cookies created by the page. Furthermore, urlscan.io provides indicators of compromise; it tracks 400 popular brand domains and tries to develop a verdict about whether the scanned URL is suspicious or malicious if it targets any of the 400 brands. Finally, some scam websites provide a video tutorial for the scam in action, which we then follow up and extract the addresses the scammer uses in the tutorial.

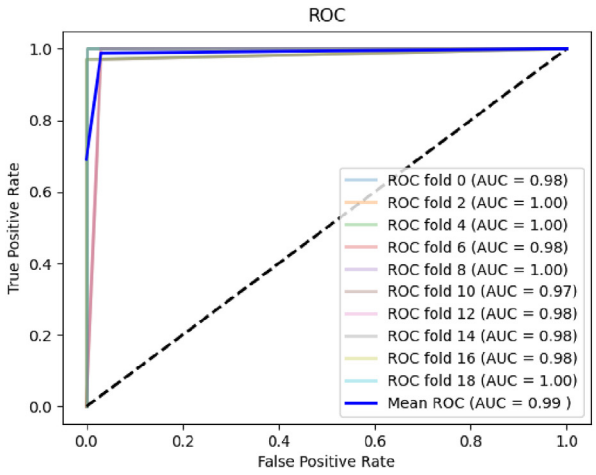
Feeding the BGS Addresses to the Anti-Phishing Work Group (APWG) data warehouse: our analysis in Section 3.1 shows that our system can detect many scam addresses before they are recorded on the blockchain (i.e., before the victims transfer any funds to the scammers). These data are now sent automatically to the APWG[13] eCrime eXchange (eCX)[14] data warehouse in real-time. APWG is an international coalition that unifies the global response to cybercrime, such as phishing and online fraud across government, industry, NGO communities, and law-enforcement sectors. ECX represents a data warehouse containing cyber threat data modules, including thousands of phishing and malicious domains. It also contains more than 70K cryptocurrency addresses used in different types of cybercriminal activities. We hope that feeding the addresses to a blacklist in the early stages will reduce the number of victims.

3. Scam collection and measurement

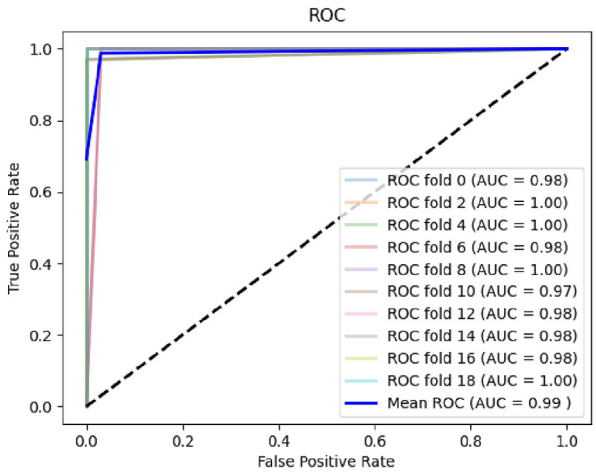
Our experiments were run on our university's server as well as on dedicated servers provided by Compute Canada[15]. The results reported in this paper come from data collected from November 2019 to February 2021. In this section, we present some basic numbers obtained directly from our crawler and classifier.

Overall, our model was able to collect 8,714 Bitcoin addresses involved in BGS. Of these addresses, 3,008 have at least one transaction. However, one particular BGS instance is responsible for most of the transaction-less addresses; the domain bitmake.io has a hard-coded list of 5,001 addresses,

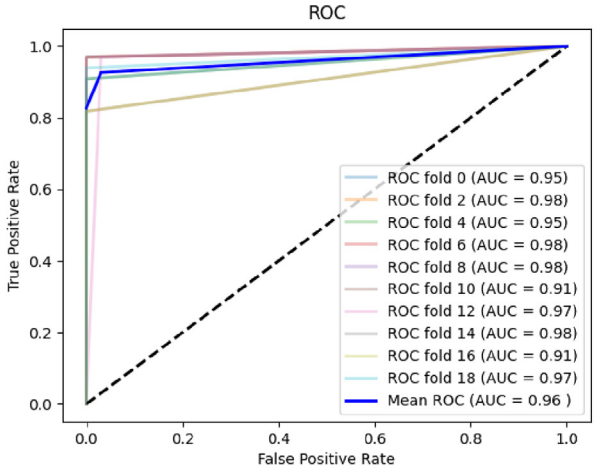
[12] A complete example of one such log is presented in our public data repository.
[13] https://apwg.org/.
[14] https://apwg.org/ecx/.
[15] https://www.computecanada.ca/research-portal/.



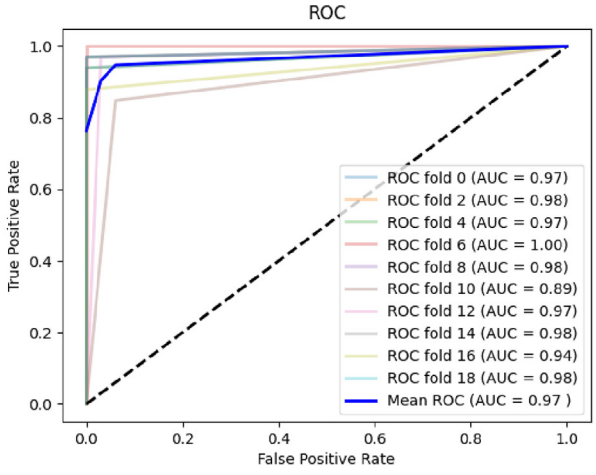
(a) SVC



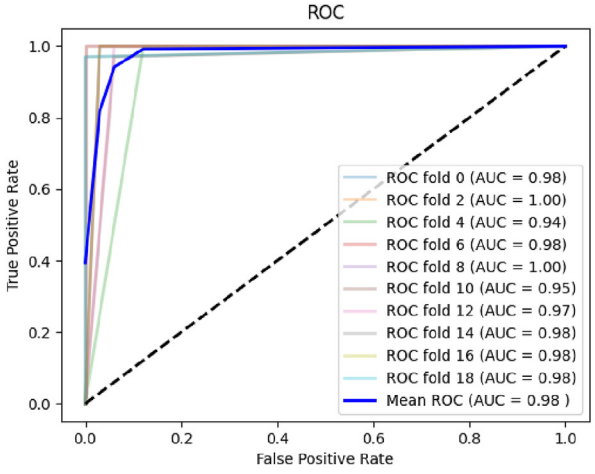
(b) MLP



(c) RF



(d) NB



(e) KNN

Fig. 5. ROC (receiver operating characteristic) curves of all folds for each classifier. SVC: support vector classifier, MLP: multi-layer perceptron, RF: random forest, NB: naive bayes, KNN: k-nearest neighbors.

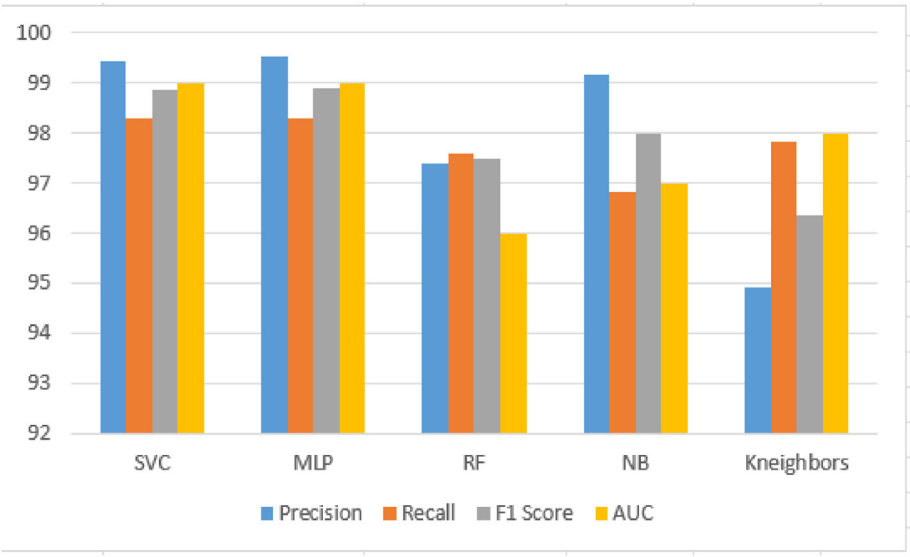


Fig. 6. Performance comparison of five classifiers to predict BGS (Bitcoin generator scam) instances. SVC: support vector classifier, MLP: multi-layer perceptron, RF: random forest, NB: naive bayes, KNN: k-nearest neighbors, AUC: area under the curve.

Table 2
Classifier Accuracy on 100 randomly selected pages that have not been observed in the training phase. Labeling the ground truth manually.

	Actually Clean	Actually BGS pages
Classified clean	99	1
Classified BGS	2	98

Note: BGS: Bitcoin generator scam.

and one of these addresses is selected randomly when a payment is made. At the time of writing, during that particular BGS instance, only 39 of the 5,001 addresses had transactions, so that site alone is the source of 4,962 of the 5,706 transaction-less addresses in our database (that is 86.96% of them). Without that site, around 80% of the addresses have transactions. These addresses have been found on 1,010 unique scam domain names[16].

About half of the BGS domains (463 of them) contain a single payment address. At the other extreme, 70 of these domains (7%) are associated with at least ten addresses. We found 144 addresses that belong to cryptocurrencies other than Bitcoin. Fifty-nine are Ethereum addresses, 26 are Litecoin (LTC) addresses, 17 are Bitcoin Cash (BCH) addresses, and 42 addresses belong to other currencies such as Dash and Zcoin. Since the vast majority of the addresses are Bitcoin addresses, we will focus on that currency in the rest of our analysis. Finally, our analysis also showed that none of the Alexa top 1K domains[17,18] contains actual BGS instances. Therefore, we only report results for URLs hosted on domains outside the Alexa top 1K.

In Figs. 7 and 8, we present the number of BGS URLs and addresses detected per day. Here, we do not include the URLs and addresses found in the Internet Archive in order to only count newly discovered and currently active BGS instances.

On average, our model detected about 2.2 new BGS instances and

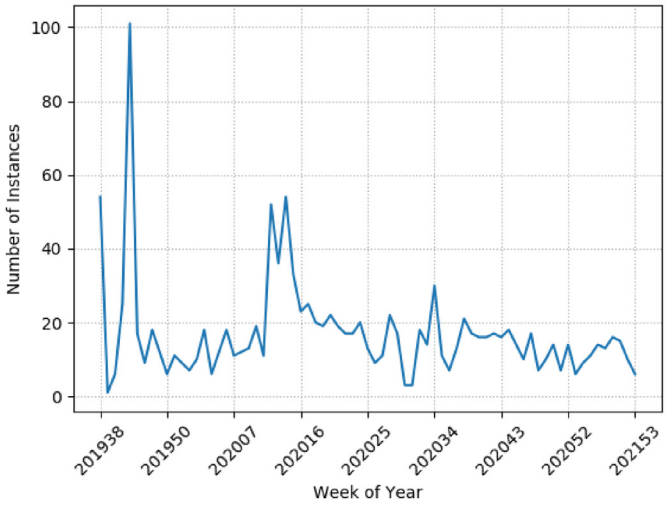


Fig. 7. Number of BGS (Bitcoin generator scam) URLs detected per week.

4.4 new Bitcoin addresses every day from November 2019 to February 2021[19]. These numbers are relatively stable throughout the period. Therefore, we can extrapolate that our system will identify more than 800 new BGS instances and more than 1,500 Bitcoin addresses per year.

3.1. Crawler effectiveness

In this section, we discuss the ability of our crawler to detect scam addresses before it receives any transactions. We collect scam addresses in two ways: first, we revisit, daily, all the BGS instances that we have previously discovered. Therefore, if an instance publishes new addresses, our system will pick them up within 24 h. We also look at other sources, such as the Internet Archive [23], data published by urlscan.io, and tutorial videos published by scammers. In that way, we collect some of

[16] In general, we only consider second-level domain names when comparing scam URLs, except for hosting services, for which we consider the third-level domain name. So generatorbitcoin.epizy.com and miningbtc.epizy.com are counted as two separate attacks even though they are on the same second-level domain name because they are both using the hosting service epizy.com.

[17] https://www.alexa.com/.

[18] However, we include the hosting domains and the public bloggers in our analysis.

[19] Note that a new BGS instance does not necessarily mean a new address since there are some addresses that are shared among instances.

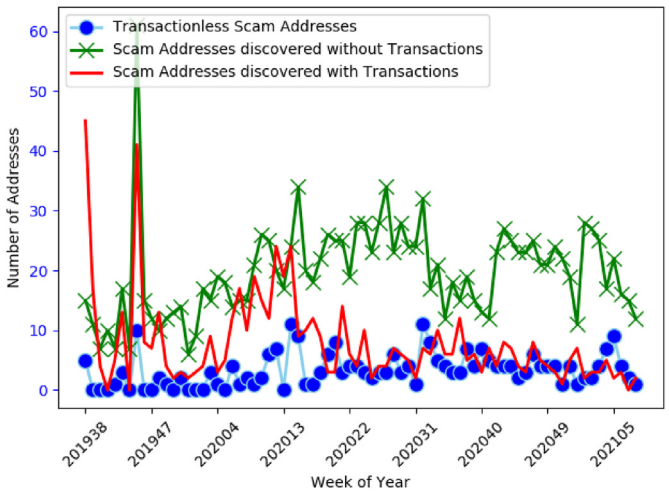


Fig. 8. Number of Bitcoin addresses detected per week.

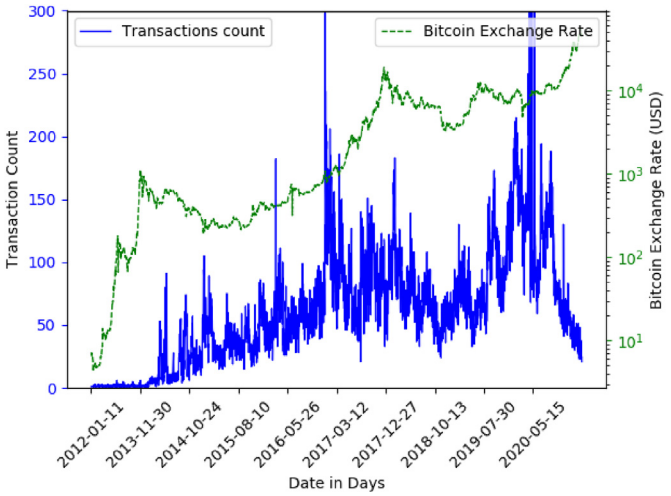


Fig. 9. Daily incoming transactions to BGS (Bitcoin generator scam) addresses.

the addresses that have been used in the past, before we discovered the instance. Our database is thus a mix of currently active addresses and addresses that have been active months or years ago.

Overall, we discovered 3,008 Bitcoin addresses with at least one transaction. Of these addresses, 2,040 (67.8% of the total) were detected by the online crawler and did not exist in the other sources. Nine hundred and ten addresses (30.2% of the total) were extracted from other sources but were never found by our live crawler. Finally, the remaining 58 addresses (1.92% of the total) were found both by our live crawler and by the other sources.

1,501 of the 2,098 addresses found by our live crawler were found before they had any transactions. Transactions eventually arrived (recall that we are here only looking at addresses that have eventually received transactions) but only after the address had been flagged by us. That is one of the unique strengths of our model, i.e., the ability to detect a suspicious address before it receives any funds. The percentage of addresses that we discovered before receiving transactions increased with time. The current value is 71.54% of the eventually active addresses being discovered before any transactions are received, compared to 55% reported in our previous work.

4. Analysis

In this section, we use our database of Bitcoin addresses to estimate how much money was stolen through the BGS. We also look at the few cases in which Bitcoins were actually transferred back to the initial address. We discuss a couple of techniques used by scammers that are making systems like ours less effective. We present the basic statistics of the BGS domains and addresses. Finally, we discuss the reuse of addresses in the BGS domains along with other types of attacks.

4.1. Bitcoin addresses Payment analysis

We first measured the scale of BGS by analyzing the transactions involving the Bitcoin addresses that we had found. Overall, we collected 3,008 addresses with at least one transaction. These addresses have received 177,952 transactions from 286,840 unique addresses. On average, the addresses received 0.018537 Bitcoins per transaction, accumulating 3,298.43 Bitcoins overall.

We then used the average exchange rate on the day of the transaction, obtained from bitcoincharts.com, to convert the value of the transactions to USD. In total, the addresses have received 8,762,177 USD. The transactions occurred between November 2013 and February 2021, when this analysis ended (the attack is, meanwhile, still active at the time of writing).

The total number of transactions and their corresponding total value in USD, compared to the exchange rate of Bitcoin, are presented in Figs. 9 and 10 respectively. As shown in the figures, there is a clear correlation between the success of BGS and the market value of Bitcoin, which is certainly not surprising. Additionally, BGS is still going steady and continues deceiving victims daily.

We can also notice two peaks in Fig. 9 that do not show a clear correlation with the Bitcoin price. We have checked these peaks and found that they are caused by three addresses that received a high number of transactions. At the first peak, in early 2017, we had two addresses that received 74% of the transactions with an average of 17.3 USD per transaction. At the second peak, in May 2020, we had an address that received 82% of the transactions with an average of 11.3 USD per transaction. These addresses have caused three peaks on four different days, but we do not know why they have suddenly received a high number of transactions. The scammers tend to transfer the funds away from the receiving address as soon as they are received, as can be seen in Fig. 11.

Finally, in our analysis, if we try to estimate the accuracy of our numbers, we note that we have reported the number of addresses and instances without extrapolation (e.g., without using clustering techniques such as the multi-input heuristic algorithm [18]). However, since we certainly did not detect all of the scam addresses, the actual number of addresses and instances are underestimated. As for the total value of received dollars, we cannot distinguish between the payments made to

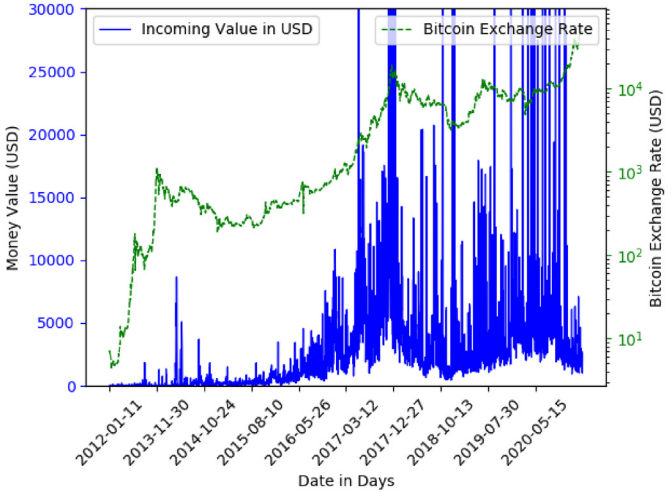


Fig. 10. Daily deposited money to BGS (Bitcoin generator scam) addresses.

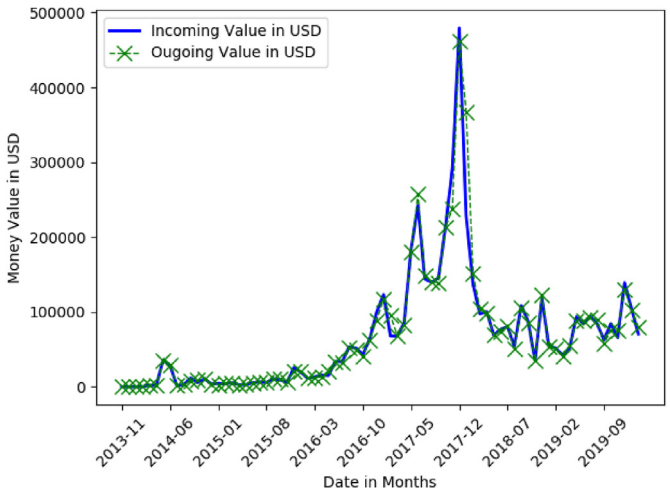


Fig. 11. A comparison between the incoming and outgoing deposits in BGS (Bitcoin generator scam) addresses.

the scammers and the payments made to the scam addresses for other reasons; thus, the results of our analysis might be overestimated, especially when it comes to early transactions.

4.2. Payback analysis

In some types of attacks, such as the Ponzi scheme, scammers provide some payback to some of the victims as part of the scheme. Additionally, it is possible that scammers conduct a transaction at a scam address and then create a reverse transaction with a multiple of the value, or they may randomly pick some benign address with transaction behavior, similar to what the scammers advertise, to convince the victims of the scan legitimacy. However, with the anonymity of Bitcoin, it is unclear whether paying back to entice the victims to invest more in the scam, or to attract new victims to invest, is effective. Although some of the scam addresses have sent some money back to addresses from which they have received payment, our analysis did not show evidence that there is an actual payback in the BGS attack.

In this section, we look at the subset of addresses that sent Bitcoins to the scam addresses and received something back from the same address[20], the way it would be if the scam was working as advertised.

Out of the 270,347 addresses that have sent Bitcoins to the scammers, 1,019 addresses have both sent and received Bitcoins from at least one of the scam addresses. Overall, 139 scam addresses (4.62% of the 3,008 scam addresses) were implicated in these back transactions. Naturally, we would like to understand if those 1,019 addresses received money back from the scam or if, in fact, both addresses belong to the scammer. To do that, we look at the transaction history of the scam addressees and further divide the 1,019 addresses into two groups:

Scam addresses: 49 of the 1,019 addresses are in this group; they are scam addresses that were already identified as scam addresses and belonged to our BGS dataset. Thus, they are internal transactions in the scam and do not represent any payback.

Normal addresses: 170 of the 1,019 addresses belong to this group. We have no concrete evidence that the scammers control these addresses (which certainly does not mean that they do not). These addresses collectively sent 163.49 Bitcoins (108,179 USD) to the scam and received 197.72 Bitcoins (185,310 USD). However, four of these addresses sent 0.026 Bitcoins and received back 28.7 Bitcoins. Excluding these outliers, on average, these addresses received more or less what they have sent. Although at this time we cannot conclusively prove that these addresses are an integral part of the scam, we can at least state that, overall, they do not impact or change our general results, as illustrated in Table 3.

Table 3
Detailed analysis for the scam addresses payback (Transac refers to Transactions).

	Total scam	Addresses that received some payment back		Addresses that did not receive anything
		Scam	Normal	
#addresses	286,840	49	970	285,821
#inTransac	177,952	288	1,984	175,680
#outTransac	71,736	365	3,721	67,650
#inBTC	3,298.43	17.28	163.49	3,117.66
#outBTC	3,279.5626	19.82	197.72	3,062
#inUSD	8,762,177	15,826	108,197	8,638,1663
#outUSD	9,438,521	16,925	185,310	9,236,280

4.3. Scam addresses delivery techniques

In this section, we look at two techniques that the scammers use to provide the scam deposit addresses that make our analysis harder. First, many of the domains regularly change Bitcoin addresses during their lifetime. Second, some of the domains generate a unique address for each victim. We are not sure of the underlying intent of these techniques. However, it limits the ability of automated systems like ours to find the BGS domains and extract their addresses. It is not surprising if scammers use these methods to prevent detection and extend the lifespan of their attacks.

Regularly changing the Bitcoin address. Four hundred and sixty of the BGS domains that we have found (that is 46.8% of the domains) have used at least two different addresses. Furthermore, in 70 of these domains (7% of the 460 domains), we found 10 or more addresses. The domains with the highest number of addresses have 5,001 addresses, 150 addresses, 143 addresses, 100 addresses, and 95 addresses that we know of. In some cases, the scam address presented to the victim is selected randomly from an array of static choices (for example, see Fig. 12). As previously explained, the most extreme case that we have detected is bitmake.io, which contains a list of 5,001 addresses, but only 39 of these addresses have any transactions associated with them. Of course, periodically changing the addresses reduces the number of transactions per address, making it more difficult to detect using techniques based on transaction history. In fact, as can be seen in Section 4.4, around 50% of the addresses have only received 1 or 2 transactions.

Distinct address per victim. In some cases, the BGS instance generates a unique scam address for each victim. We detected eight domains that use such a technique. We continuously crawled these domains and found that the attacker generates a unique address for each deposit address the victim is using. As a result, none of the addresses detected by these domains have any transactions. One consequence of this is making the attack hard to detect by detection systems that depend on transaction history.

To further study these domains, we manually searched on Google and YouTube to find addresses related to those domains that have transactions. We only found 5 Bitcoin addresses with funds related to the domain doublebitcoin.win from 3 YouTube reviews published by a scam researcher[21], Each of the addresses received a single transaction with a total of 0.026579 Bitcoins. The transactions are related to the review provided by the researcher. We examine the connections between the 5 addresses in Appendix A.

[20] A limitation of this analysis is that we consider the payback that is made to the same address from which it was received. This is not necessarily the case in a Bitcoin transaction [49], and we would miss the hypothetical transactions for which this is not the case. However, the scam instances that we inspected did not include any other way to get payments back.

[21] https://www.youtube.com/watch?v=oic8YfMge2g.

```
// ==========================================
// Bitcoin Crypto Finder Address Database
// ==========================================
var Address=new Array() // do not change this
Address[0] = "<input type='text' class='form-control' size='38' onClick='this.select();' class='depositAddy'
value='1NXDDChyj2qgRYFzCqJiXNkPixZdYluXYz' readonly>";
Address[1] = "<input type='text' class='form-control' size='38' onClick='this.select();' class='depositAddy'
value='1Jk9HD76cFc4eMSJokm2DN7G3dH673qYke' readonly>";
Address[2] = "<input type='text' class='form-control' size='38' onClick='this.select();' class='depositAddy'
value='1DE9ZrWvqjWdyVAzQBxvaYBWQG5zcgGhjz' readonly>";
Address[3] = "<input type='text' class='form-control' size='38' onClick='this.select();' class='depositAddy'
value='1PKK6bsVFDimNiMSYkXThRkEdh7fL3DZwz' readonly>";
Address[4] = "<input type='text' class='form-control' size='38' onClick='this.select();' class='depositAddy'
value='172pjGKt6sbojVoJfvJ4Zv7QYRLagSWbTJ' readonly>";
// ========================================
// Do not change anything below this line
// ========================================
var Q = Address.length;
var whichAddress=Math.round(Math.random()*(Q-1));
function showAddress(){document.write(Address[whichAddress]);}
showAddress();
```

Fig. 12. A real world example of a BGS (Bitcoin generator scam) instance in which the payment address is selected randomly from a list.

4.4. BGS addresses statistics

In this section, we report some basic numbers about our scam address transaction history:

- The lifetime of the active addresses is defined as the number of days between the first and last incoming transaction.
- The longest period during which an address was inactive, was counted in days.
- The fraction over time of the total number of transactions received from the day of the first transaction to the day of the last transaction.
- The number of days a scam instance was active, defined as the number of days between the day of the first address discovered in the domain and the day of the domain becomes inactive. For active domains, we count until the day of the analysis.

Of course, these numbers are biased by the end of our experiment since addresses and scam instances are still active afterward. We believe that the relatively short lifespan of most scam addresses, much shorter than the duration of our experiment, ensures that this analysis is still quite informative.

Table 4 shows our data. We can see that around 40% of the addresses have lived at most a month. On the other hand, 40% of them have lived at least a hundred days. For the number of days, an address was idle without receiving any transactions, most of the addresses had a short inactive time; around 70% of the addresses were inactive for less than 54 days. On the other hand, around 10% of the addresses were idle for more than 200 days at some point. Finally, the majority of the addresses have received a low number of transactions. Approximately 90% of the addresses have received at most 60 transactions, and 50% of them have received fewer than 3 transactions.

In the case of BGS domains, they have a relatively long life span. Less than 20% of the domains lived less than a month, and around 60% of the domains were active for more than a hundred days. The average lifetime of BGS domains is 250 days, and the median lifetime is 140 days[22]. Comparing the BGS active time to other cryptocurrency-related scams, such as HYIP schemes, we see that the active time is much higher. Vasek and Moore [8] reported that the median lifetime of the Bitcoin HYIP scheme is 37 days and the bridge HYIPs[23] is 125 days.

Table 4
General statistics.

Fraction of data	Addresses active life time	Addresses longest inactive time	Transactions per address	Domains active life time
0.1	2	1	1	15
0.2	5	2	1	38
0.3	12	4	1	73
0.4	27	8	1	107
0.5	60	16	2	140
0.6	115	29	5	192
0.7	222	54	10	255
0.8	363	94	23	344
0.9	728	200	60	671
1	2657	1997	1542	2113

4.5. Addresses reuse

In this section, we look at the addresses reused. We first investigate BGS address reuse in our scam domains database. Second, we crosscheck our addresses with public datasets maintained by other authors.

Our analysis has shown that some addresses have appeared in different scam domains. Overall, we identified 266 addresses that were used in more than one scam domain. Twelve of these addresses have been used in more than five domains, and the most reused address has appeared in ten domains. We cannot ascertain the underlying intent of reusing addresses, but it may help convince victims to transfer funds to the scam: Since Bitcoin transaction history is publicly available, an address with a history of receiving and sending transactions, or a large balance, may convey more credibility. For example, in one of the BGS instances[24], the attacker advertised the ownership of a Bitcoin address with a high fund. The attacker claimed that the address was maintained to pay back the received funds from the victims.

In our second analysis, we look at addresses used both for BGS and other types of scams. For this purpose, we crosschecked our addresses with 10 public datasets maintained by other authors, which we collected in Ref. [30]. None of these databases are about BGS. However, as shown in Table 5, 92 BGS addresses have been found in 3 datasets. This suggests that there is some level of address reuse across different types of scam attacks.

5. Other BGS cases

In this section, we discuss two other types of BGS that we found

[22] Since some of the domains are still active, the average lifetime of the domains may be underestimated. However, because we are using historical data in our analysis, we cannot verify if the domain was used for other purposes during its active time or it was inactive for some period of time; thus, our analysis might be overestimated.

[23] HYIP schemes that first start as traditional HYIP attacks before being used in the Bitcoin ecosystem through posts on Bitcointalk.org.

[24] https://pastebin.com/sf0vMVAE.

Table 5
Crosschecking the BGS (Bitcoin generator scam) dataset with other public datasets.

Reference	#addresses	#inCommon	Year	Crime type	URL
[31]	1246	0	2018	HYIP	https://bit.ly/3nLcB9E
[7]	52	7	2018	HYIP	https://goo.gl/ToCho7
[9]	3	0	2018	Ransom	Hardcoded in the paper
[32]	126	0	2018	Ransom	Hardcoded in the paper
[6]	2026	1	2018	General	https://goo.gl/sQJKdx
[33]	1853	0	2019	Honeypot	https://honeybadger.uni.lu/
[19]	1566	0	2019	HYIP	https://goo.gl/k5PCOZ
[16]	182	0	2020	HYIP	https://goo.gl/CvdxBp
[34]	3750	84	2020	General	https://cryptoscamdb.org/scams
[35]	2179	0	2020	General	https://bit.ly/32 pmC2A

Note: HYIP: high yield investment program.

through our analysis. In the first type, the victim is asked to install an executable mining file on their machine. In the second type, the victim is asked to complete one or more tasks instead of paying the mining fees.

5.1. Malicious executables

In some cases, the victim is provided software that can supposedly hack the blockchain. In this case, the attacker provides an executable file. During our analysis, we collected 12 executable files, all targeting the Windows OS.

We scanned the 12 files using Virus total[25]. Virus total scans any file or URL with over 70 antivirus scanners and URL/domain blacklisting services. An example of the results returned by Virustotal is presented in Fig. 13. All of the 12 files were flagged by at least one antivirus scanner, and eight (66.66%) of the files were flagged by at least five scanners. Many of these files were flagged by Avast[26], Avg[27], BitDefender[28], and Kaspersky[29]. Traces of Trojan, Malware, Bitcoin miner, Coin miner, Dropper, and Adware were reported.

5.2. Click Per Action (CPA) Scam

In other cases, when the victim provides the information needed by the generator and the success message is displayed, a new screen is shown to the user, asking him/her to complete one or more "offers" for verification purposes (as shown in Fig. 14). This screen is called a "content-locker" (CL) by the creators of these scams. The "CL" with its set of offers, is what the scammer ultimately wants the victim to see in this type of BGS attack. These so-called offers represent the final payload and include, but are not limited to, clicking through endless "surveys", filling out "market research" forms, collecting personal information, getting the victims to subscribe to questionable services, installing suspicious executable files on their machines, etc. An example of subscription offers is presented in Fig. 15. This attack variation is similar to the "Game Hack" scam, which was recently investigated in Refs. [26,27]. Both attacks use similar templates and lead to the same final payload.

Our dataset contains 40 (4% out of the 983) domains that present this kind of offer as the final verification process. Four of these domains used a mix of offer verification and mining fees to collect the funds.

6. Scam clustering

In this section, we apply two clustering techniques to the BGS addresses collected in Section 3. With the first technique, we try to infer more Bitcoin addresses controlled by the same scammers. In the second technique, we use a variety of features to cluster BGS addresses into campaigns controlled by the same scammers.

6.1. Inferring new scam addresses using the "multi-input heuristic"

When studying attacks that use cryptocurrencies as a payment medium, such as HYIPs and ransomware, many authors have difficulties collecting a large number of addresses or collecting a set of addresses controlled by the same owner.

To increase the number of addresses or acquire a set of addresses controlled by the same scammer, some authors use "multiplier" techniques such as the multi-input heuristic [4,6,19,31].

In the multi-input heuristic, the assumption is that the same person owns all the addresses on the input side of any transactions [4,6,19,31]. Usually, these kinds of transactions occur when user *X* wishes to transfer a specific Bitcoin value to user *Y*. However, none of the user *X* addresses has sufficient funds to complete the transaction. In such cases, user *X* can construct a multi-input transaction redeemable by user *Y*, which also avoids paying multiple transaction fees [6]. Under that heuristic, if a single address is known on the input side, all the other addresses on that same input side are deemed to belong to the same actor, thus increasing the number of addresses attributed to that actor.

However, the multi-input heuristic was deemed to be error-prone and difficult to apply, especially with the introduction of CoinJoin, in which multiple senders combine their payments in a single joint transaction [36,37].

In this section, we use a restrictive version of the multi-input heuristic to infer more scam addresses related to BGS and assess the possible upper limit of damage caused by the BGS operators. To reduce the possibility of errors in the inferred addresses, we add two constraints. First, we only consider a transaction if there is a single address on the transaction output side. Second, we only consider transactions for which the number of confirmed scam addresses on the input side reaches a certain threshold. To identify the optimal threshold that yields the lowest false positive rate, we applied the heuristic to four different datasets. Each dataset contains 500 addresses chosen as follows:

- BGS dataset. It is a set of confirmed scam addresses that we found with our model. Some of the addresses are related in that they were extracted from the same scam domain, and some are completely unrelated and were extracted from different domains.
- Clean dataset #1. It is a set of random Bitcoin addresses that we have constructed by randomly selecting the addresses from more than 350K unique Bitcoin addresses extracted from 40 consecutive blockchain blocks.
- Clean dataset #2. It is a set of tightly connected addresses in which all the addresses co-spent together in a single transaction[30].
- Clean dataset #3. In this dataset, we chose the addresses to resemble the BGS dataset. Some of the addresses are related in that they were

[25] https://www.virustotal.com/gui/.
[26] https://www.avast.com/en-ca/index#pc.
[27] https://www.avg.com/en-ca/homepage#pc.
[28] https://www.bitdefender.com/.
[29] https://www.kaspersky.ca/.

[30] i.e., all the addresses appeared in the input side of the same transaction.

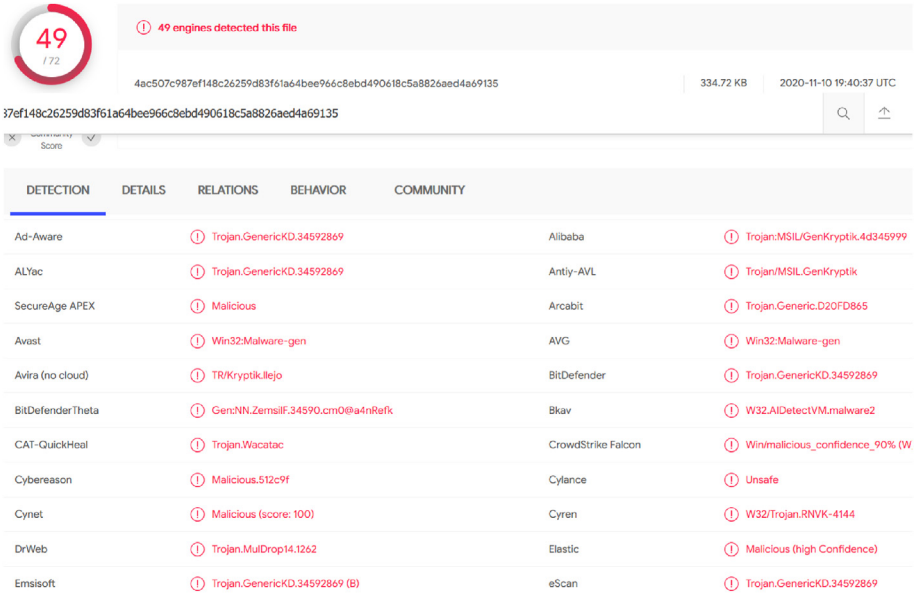


Fig. 13. An example of virus total scan results.

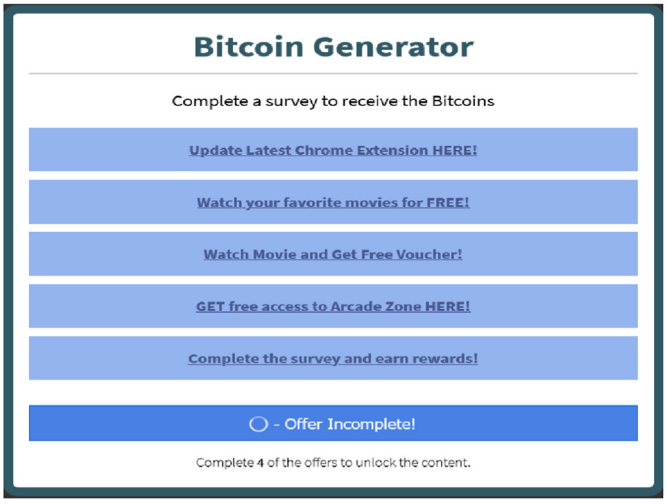


Fig. 14. An example of the scam content locker.

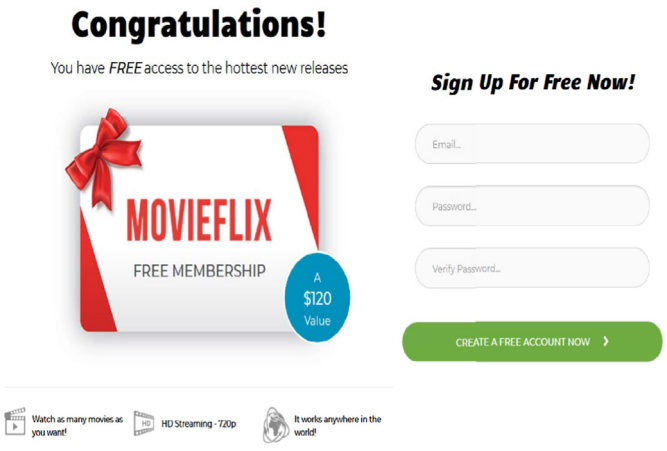


Fig. 15. An example of the scam offers.

extracted from the input side of a single transaction, and some are completely unrelated and were extracted from different transactions.

We show the results in Table 6. We applied the algorithm in eight rounds while increasing the threshold each time. For example, in the last row of the table, we consider a transaction if at least eight addresses from the dataset appear on the input side. As can be seen in the table, for dataset #1, the heuristic did not infer related addresses when the threshold was more than one. For dataset #2, the number of inferred addresses is relatively high, even with a threshold as high as eight. In the BGS dataset and dataset #3, the number of inferred addresses decreased with the increasing threshold. Based on these results, we used eight addresses as our threshold.

By applying the heuristic to our complete dataset, we inferred 2,371 more scam addresses. Overall, these addresses received 12,657.798 Bitcoins (approximately 36 million USD), with an average of 0.115633 BTC per transaction. The average value of received Bitcoins per transaction is higher than the average value of the BGS addresses.

We further checked the addresses of transactions and found that 37 of them have received Bitcoins from BGS addresses. We think that these addresses may be owned by the scammers but used in legitimate transactions or used in different scams. As a result, our analysis in Section 4.1 might underestimate the actual number of scam addresses and the damage inflicted on the victims.

6.2. *Connecting campaigns together*

In this section, we attempt to cluster the BGS websites and their related addresses into campaigns operated by the same scammer. For this purpose, we use a variety of features and identifiers. We use website-related features that were deemed suitable for illicit website clustering in previous work [34] and address-related features that we extract from the blockchain transaction history. Some of the website features are used individually (for example, Refs. [38,39]), and some are combined (for example, Ref. [34]) with group websites. To the best of our knowledge, this is the first time all these features have been applied together to infer campaigns operated by the same scammer.

1. Level 1 **groups per domain**: this is a direct grouping feature in which we consider all the addresses within the same website to be controlled by the same scammer.
2. Level 2 **address reuse**: our analyses in Section 4.5 have shown that some attackers use the same address in different BGS domains to carry out their attacks (we have 266 addresses that have been detected in

Table 6
The result of applying the multi-input heuristic on different datasets.

#of co-spent addresses	BGS_Data	Dataset #1	Dataset #2	Dataset #3
1	17,230	7,151	445,490	14,662
2	9,617	0	378,345	2,939
3	7,472	0	327,591	1,162
4	3,922	0	288,680	550
5	2,648	0	255,898	447
6	869	0	230,878	273
7	138	0	208,600	39
8	28	0	189,568	0

more than one BGS domain). At this level, we merge two clusters when they have some common addresses.

3. Level 3 **analytic/tracking ID**: in some of the scam instances, we found the signature of online advertisements and statistics websites. When using such services, identifiers have to be embedded in the DOM of the sites so that the service can track that particular site. In some cases, people reuse the same identifier across different sites, either on purpose to aggregate the results or simply by mistake. Separated sites having the same identifier suggest that they belong to and are operated by the same entity [26,27]. Some of these identifiers relate to third-party analytic services such as the sites histats.com and statcounter.com. However, it does not mean that either histats.com or statcounter.com have any part in the scam, merely that scammers tend to use these sites for their analytics. Other identifiers commonly found in the DOM of scam instances are related to the sites that provide scam templates and offers at the end of the scam process. Other researchers [34,40] have shown that a Google Analytics ID can be used to cluster separate illicit websites into campaigns.

These identifiers often require a user account ID to be placed within the DOM of scam instances. Finding matching identifying account IDs in the DOM of seemingly unrelated websites suggests that they belong to the same individual. At this level, we merge two clusters if they have domains that contain the same identifier ID.

4. Level 4 **IP address**: the same IP address can serve the content of numerous domain names. Being hosted on the same IP address has been used as a feature to link illicit websites to the same scammer [34]. At this level, we merge together clusters if they have domains hosted on the same IP.

In some cases, many websites could be hosted on public hosting services or share the content distribution network IPs, introducing some errors in our clustering. We have manually checked some of the IPs that caused a merge between our clusters and found that the domains hosted on these IPs share similar template and domain names, indicating that they are connected. For example, the domains bitgenx.online and bitcoingenerator2020.club are hosted on the same IP and have similar templates.

5. Level 5 **fund transfer between scam instances**: at this level, we merge two clusters, A and B, if addresses from A appeared on the input side and addresses from B appeared on the output side of the same transaction.

We provide an overview of each clustering level's outcome in Fig. 16. In the figure, we present the number of clusters at each level and the two clusters with the highest value in USD. For each of the two clusters, we show the number of domains, the number of cryptocurrency addresses, the incoming value in USD, what caused the merge between the clusters from previous levels, and a sample of the domains that caused the merge and what type of connections exist between them. The cluster number in the class represents a numeric value to distinguish between the different clusters.

Using our clustering method, we could connect different scam attacks and link them to the same scammer. Our results show that a small group of scammers controls the majority of the received funds. The top two clusters[31] have received around 5 million USD, which is more than half of the total funds received by the scam addresses.

To view the relations between domains and addresses, we build what we call the domain/addresses connection graph. Specifically, to connect the nodes, we use the features that caused the merge at different levels. The domain/addresses connection graph of one of the top two clusters from level 5 is shown in Fig. 17. The red nodes represent the domains, the black nodes represent the addresses, and the edges represent the connections created during the clustering process. In the graph, an edge is created between a domain and an address if the address is found in the domain DOM. The green edges connect the domains that have the same analytic/tracking ID (generated from level 3). The red edges connect the domains hosted on the same IP (generated from level 4). The blue edges connect the addresses that transferred funds to each other (generated from level 5). The edge size correlates to the number of transfers between the addresses. An interactive domain/addresses connection graph of the top two clusters can be accessed at https://ebadawi.github.io/level5_1/ and https://ebadawi.github.io/level5_2/, respectively.

7. Investigating BGS in other languages

In our analysis, we trained our classifier on pages with English text only. Thus, we focused our research on pages with English text. In this section, we use a text-independent classifier to investigate whether we can find any evidence of significant BGS attacks in other languages.

For this purpose, we have expanded the 696 search queries generated in Section 2.2 to include non-English queries. We used the Google translator to translate the 696 queries into different languages, which are: English, Hindi, Spanish, French, Ukrainian, Russian, Chinese, and Swahili. We have targeted the 5 most spoken languages[32] and the languages spoken in the top 5 countries with the highest cryptocurrency adoption index[33]. We have used our model to perform the extended search queries for 2 days, during which we collected 14,825 pages identified as non-English pages.

To detect the presence of BGS instances in these pages, we have identified four non-language dependent features from the BGS instances to train a classifier:

- **The presence of cryptocurrency addresses:** this feature checks the existence of cryptocurrency addresses within the HTML page content. We look for the pattern of 16 cryptocurrencies that we have observed during the first part of our analysis.
- **Domain name:** this feature checks for the existence of terms related to cryptocurrency or the scams in the domain name. For example, we have observed the terms “btc”, “bitcoin”, “generate”, and “invest” in many scam domains.
- **The presence of input fields:** BGS instances usually contain an input field to accept the victim's address to deposit the alleged generated coins. For this feature, we simply look for the tags related to the buttons. We include the tag <input>.
- **The presence of buttons:** BGS instances usually contain a button to initiate the false generation process. For this feature, we simply look for the tags related to the buttons. We include the tag <button>, the tag <input> when the type is “button”, and any other tag with “class” or “id” related to buttons.

We used these features to train five machine learning algorithms from the Scikit-learn Python library [29]: KNN, Neural Networks (NN),

[31] We suspect that one or two groups of scammers control these clusters.
[32] https://www.visualcapitalist.com/100-most-spoken-languages/.
[33] https://markets.chainalysis.com/#geography.

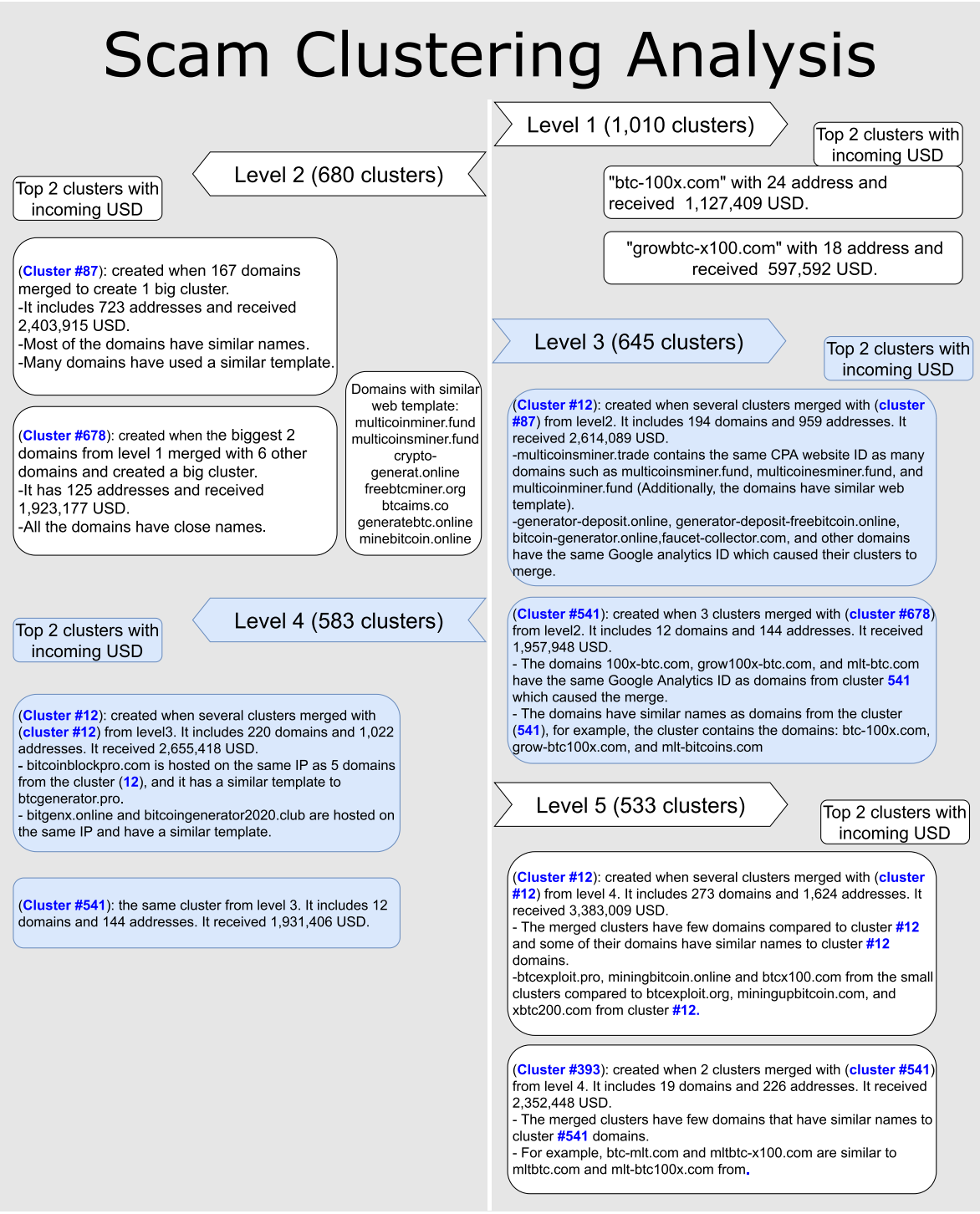


Fig. 16. BGS (Bitcoin generator scam) addresses clustering analysis.

SVC, RF, and NB. To evaluate the classifiers, we used 10-fold cross-validation on the labeled dataset that we prepared in Section 2.1. We evaluated the performance using the AUC: the greater the AUC was, the better the performance. The results are shown in Fig. 18, we can see that all the classifiers performed well, with RF having the greater area.

We then used the RF classifier to classify the set of 14,825 non-English pages we had gathered. Of these pages, 14,770 were classified as clean, while 55 pages were classified as BGS instances. We manually inspected the pages classified as BGS instances and verified that nine of them were scam pages. Six of these pages are English pages that were wrongly identified as non-English by our language detector[34]. The other three pages are BGS instances with non-English text. We also inspected 50 randomly selected pages that were classified as clean and verified they were classified correctly.

Our feature classifier is not perfect and, in particular, is not as

[34] We have used our text classifier on these pages and they were identified as scams.

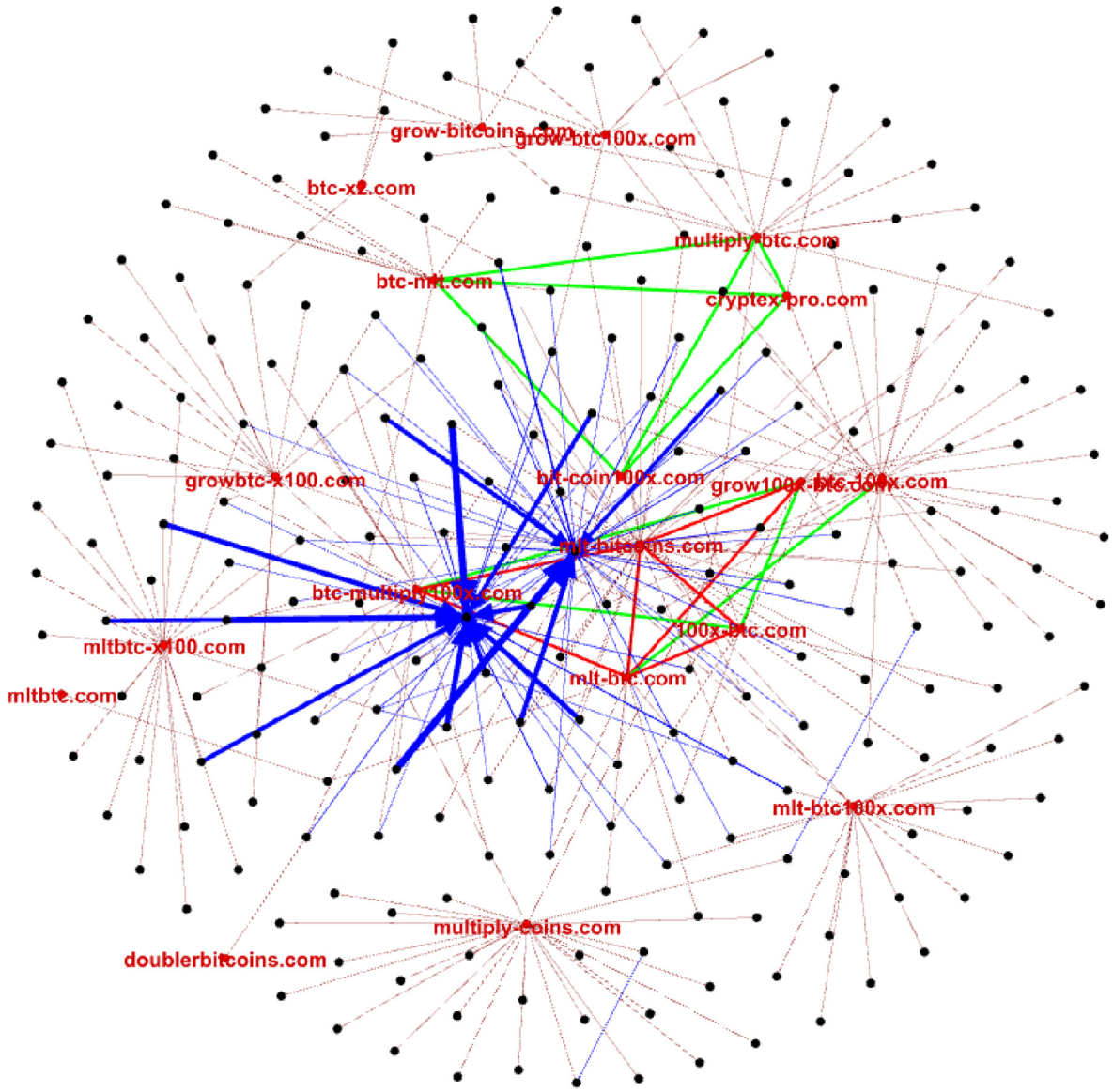


Fig. 17. Cluster #393 BGS (Bitcoin generator scam) domains and addresses connection graphs (generated by Gephi using layout Fruchterman Reingold).

effective as our text classifier. However, this experiment seems to conclusively indicate that the BGS is carried out mostly in English[35]. Therefore, we are confident that our English-only study is, in fact, very representative of the attacks as a whole, at least at the time of writing.

8. Related work

The current state of the art for Bitcoin scam detection usually relies on extracting features from the Bitcoin transaction history to train a classification model [4–7]. The classification model is trained on features such as the ratio of received/sent transactions to all transactions, the frequency of transactions, or the address lifetime. To collect these addresses, the authors manually searched on Bitcoin discussion forums such as bitcointalk.org [7], or they used semi-automated web crawls of the same forums, followed by manual inspection and address collection [4–6].

Toyoda et al. [19] used a binary classifier to identify the Bitcoin addresses of Ponzi schemes. The authors trained an RF classifier on features extracted from the transaction history of 1,500 addresses that they

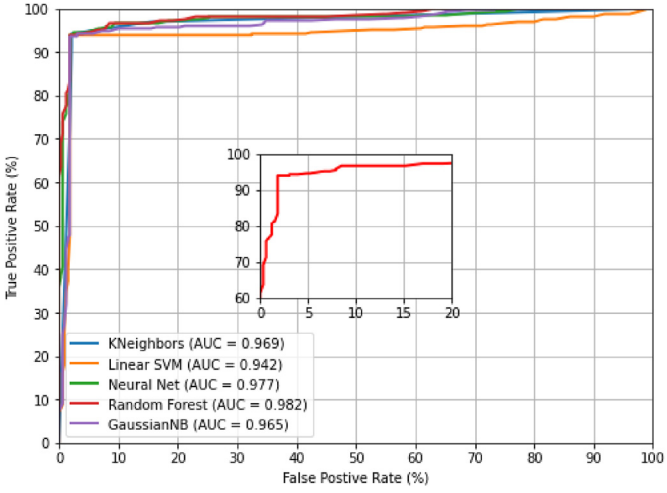


Fig. 18. Feature classifiers area under the curve (AUC).

[35] More precisely, the BGS instances that can be found through our search queries are almost exclusively English.

manually scraped from bitcointalk.org and blockchain.info. The classifier is able to identify HYIP addresses with a true positive rate of 83% and a false positive rate of 4.4%. In Ref. [4], the authors automated the address collection process by using a crawler to collect possible posts that advertise HYIP schemes. They then manually inspected the returned posts and extracted more than 2,000 addresses of Ponzi schemes. The newly discovered addresses increased the classifier's true positive rate to 95% while maintaining a false positive rate of less than 5%. Furthermore, they reported that limiting the number of transactions taken into account per address to only 100 transactions reduces the feature computation time while maintaining good detection accuracy.

In a follow-up work [6], the authors introduced more features and proposed a multi-class Bitcoin-enabled service identification to classify Bitcoin addresses into one of seven major "services", such as gambling, exchange, scam, and mixer. Furthermore, the authors extracted the features based on the address or the owner. In the owner situation, the authors used the multi-input heuristic clustering algorithm to separate the addresses into groups owned by the same person before extracting the features. The owner-based classification achieved a higher accuracy of 72% compared to 70% for the address-based classification.

Bartoletti et al. [7] implemented a binary classifier using a random forest classifier to detect Ponzi addresses based on features extracted from the address transaction histories. They trained the classifier on 32 addresses of HYIP operators that they manually collected from old posts on Reddit and bitcointalk.org, along with 6,400 non-HYIP addresses. The authors reported that the best detection accuracy was achieved using a "cost-sensitive" approach that gives a higher penalty for misclassifying an HYIP address than for misclassifying a legitimate address. Additionally, the authors collected 1,211 addresses controlled by the same scammers using the multi-input heuristic clustering algorithm. Their analysis showed that more than 50% of the scammers use more than one Bitcoin address. Overall, the scammers received 10 million dollars at the addresses they controlled.

In Ref. [41], the authors formulated the problem of detecting illicit accounts in the blockchain as a node classification task in the transaction graph. They proposed "SIGTRAN", a graph-based method that uses the transaction records from blockchain to generate a graph representation. They then extracted different features to analyze the nodes' activity signatures and used them to distinguish illicit nodes.

Unlike the previous approaches, we do not rely on previously reported campaigns in our analysis. Instead, we proactively crawl the web, searching for new, unreported instances. Furthermore, our detection phase does not depend on the address transaction history, which allows us to find transaction-less addresses.

In Ref. [34], Phillips and Wilder used the DBSCAN clustering algorithm [42] to cluster different isolated scam websites into smaller distinct types of attacks based on information extracted from the scam websites' HTML content. They then used the same algorithm to identify scam campaigns controlled by the same scammer based on features extracted from the websites' registration and ownership details. The authors used the Google Analytics ID, domain registrant details, domain registrar, and IP address in the campaign clustering.

In our multi-level clustering, we have used features that we extracted from the scam websites and the cryptocurrency address transaction history. We have also used analytics/tracking IDs that belong to different websites, such as histats.com and statcounter.com.

Other researchers focused on studying and analyzing Ethereum-based Ponzi schemes using the smart contract code [16,43]. Bartoletti et al. [16] studied and manually analyzed 191 smart contracts retrieved from etherscan.io to identify Ponzi schemes using their source code. In Ref. [43], the authors introduced SADPonzi, which is a semantically-aware model that includes a heuristic-guided symbolic execution technique to detect smart Ponzi schemes on the blockchain using the contract's runtime opcodes.

Finally, some authors surveyed and provided a general overview of the literature studying the cryptocurrency scam industry. Bartoletti et al. [44] used different public resources to build a dataset of thousands of cryptocurrency scams and analyzed them to create a scam taxonomy based on seven different features that capture different types of illegal activities. They also provided a large scam dataset that they compiled during their study. In Ref. [45], Trozze et al. conducted a systematic literature review and discussed the different types of cryptocurrency fraud activities that currently exist or are expected to exist in the future. The authors built their study based on published academic research on cryptocurrency fraud, gray literature, and expert opinions. Other researchers [46] classified the threat models that target the Internet of Things network's blockchain protocols into five main categories: cryptanalytic attacks, identity-based attacks, service-based attacks, manipulation-based attacks, and reputation-based attacks.

Similarly, in Ref. [30], we conducted a systematic literature review in which we explored and aggregated the state-of-the-art threats that have emerged with cryptocurrencies and the defensive mechanisms that have been proposed. We also discussed the threat types, scales, and efficiency of the defensive mechanisms in providing early detection and prevention. We also listed the resources used to collect datasets and identify the publicly available ones.

Although recent papers have provided important insights into different types of cybercrimes that use cryptocurrencies as a payment medium, we are not aware of any study focusing on what we have called the "Bitcoin Generator Scam" before this one.

9. Limitations and future work

One of the limitations of our work is that we used the BGS instances we already found to look for more instances. As a result, we may miss some of the instances if they have very different characteristics. If some of these instances were found, new search queries could be generated to include them in a future study.

We are also relying on search engines to find new instances, which implies that new instances have to first be indexed by these search engines, or the pages referencing these instances have to be indexed. It is reasonable to assume that scammers will go out of their way to increase the exposure of their attacks, since the attack has to be found by the victim to be useful to the scammer. It is, however, possible that attackers will use different media to advertise the attacks (social media, spamming campaigns, messaging, etc.). An instance that is exclusively promoted outside of the surface web (indexed by search engines) will not be detected by our current method.

Note, however, that scammers do not have unlimited freedom in the techniques they can use, as their basic success factor is the number of victims who reach the scam sites. Evading our model, e.g., by using social media and not relying on victims actively finding their sites, will make it harder for scammers to spread their scams widely, thus reducing their overall profit. This encourages scammers to widely publish their attacks, especially when there are no consequences to having the scam be reachable through search engines, such as being taken down or being blacklisted. Nevertheless, in our future work, we will expand our search to include social media scraping, such as using Twitter's streaming API, which has proved to be efficient in detecting spam campaigns [47,48].

Finally, we used a text-based classifier to implement our classification model, which can be evaded relatively easily. In our future work, we would like to enhance our feature set to be less dependent on the text that is being presented to the user. We will build a more accurate classification model by adding some non-text-based features.

The limitations outlined above, if they indeed apply, mean that we are underestimating the extent of the attacks and the damage inflicted.

10. Conclusion

In this paper, we investigated what we call the "Bitcoin Generator Scam". In BGS, the scammers promise to generate free Bitcoin using dubious methods, such as owning a high-speed mining device or the ability to hack the blockchain. The attack is being advertised through

web pages and targets victims who are looking for an easy profit using cryptocurrency. We created a system that automatically searches the internet for scam pages, monitors their behavior, and collects the cryptocurrency addresses used by the scammer.

Identifying scam addresses by analyzing blockchain history is typically difficult, error-prone, and only works on addresses with a good transaction history. However, our system proactively looks for the source of the scam, which enables us to detect transaction-less addresses or addresses with a low number of transactions. Finally, we also innovated with the source of information we use. In addition to using traditional search engines, we showed that services such as the Internet Archive, urlscan.io, and CuteStat.com can be used to significantly increase the number of instances found.

Our data collection work spanned 16 months. In that time, we uncovered 8,714 cryptocurrency addresses extracted from 1,010 unique domains. These addresses have received 8,762,177 USD, with an average of 49.24 dollars per transaction. We also used several features that we extracted from the scam websites and the address transaction history to link scam instances and create groups of scams controlled by the same scammer. Our system has been integrated as one of the "feeds" to the Anti-Phishing Working Group Cryptocurrency eCrime Exchange database.

Finally, we believe that our main contribution is our automated approach, which can be used to detect different scams that have a significant web presence. By actively looking for the source of the scam instances, we were able to discover 8,714 addresses directly advertised by the scam. This is a much greater number of addresses than is usually found in state-of-the-art research, where typically the scam instances are manually collected, and the bulk of the addresses come from clustering techniques such as the multi-input heuristic algorithm [18].

All the data used in our study are freely available at https://ssrg.eecs.uottawa.ca/bgsextended/.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Fundings

This work was supported in part by Canada's Natural Sciences and Engineering Research Council (grant number "CRDPJ 539938-19") and IBM Centre for Advanced Studies (CAS) Canada (grant number "1059").

Appendix A. doublebitcoin.win Addresses Graph Representation

In this section, we investigate whether the five Bitcoin addresses related to the website doublebitcoin.win are related or not. For this purpose, we have built what we call the addresses/transactions connection graph. Specifically, we use the address transaction history to connect the five addresses. We have used the multi-input heuristic twice to infer more addresses related to the scam. The first time, it was applied to the five addresses and then inferred 144 new addresses (level 1). The second time, it was applied to the 144 addresses and inferred 554 addresses (level 2).

Our graph is shown in Fig. A.1. The nodes represent Bitcoin addresses, and the edges represent coin flows. We used different node colors to distinguish between the addresses as follows:

- **Red nodes** represent the initial five addresses.
- **Blue nodes** represent the addresses identified in the multi-input heuristic first level.
- **Green nodes** represent the addresses identified in the multi-input heuristic second level.
- **Black nodes** represent other addresses that appeared in the transaction history.

As shown in the graph, the five addresses, and those identified using the multi-input heuristic, are connected through a different series of transactions, which suggests they are related to the same scammer.

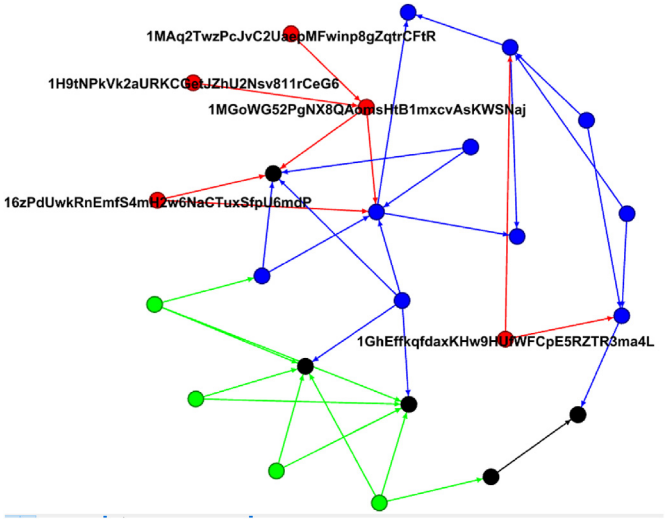


Fig. A.1. A graph representation of the doublebitcoin.win Bitcoin generator scam domain addresses.

References

[1] J. Kamps, B. Kleinberg, To the moon: defining and detecting cryptocurrency pump-and-dumps, Crime Sci. 7 (1) (2018) 18.

[2] CoinMarketCap, Cryptocurrency Market Capitalizations, 2021. Available online: https://coinmarketcap.com/.

[3] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system, 2008. Available online: https://bitcoin.org/bitcoin.pdf.

[4] K. Toyoda, P.T. Mathiopoulos, T. Ohtsuki, A novel methodology for hyip operators' bitcoin addresses identification, IEEE Access 7 (2019) 74835–74848.

[5] M. Vasek, T. Moore, Analyzing the bitcoin ponzi scheme ecosystem, in: A. Zohar, I. Eyal, V. Teague (Eds.), Financial Cryptography and Data Security, Springer, Heidelberg, Berlin, 2018, pp. 101–112.

[6] K. Toyoda, T. Ohtsuki, P.T. Mathiopoulos, Multi-class bitcoin-enabled service identification based on transaction history summarization, in: 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData); 30 Jul–3 Aug 2018; Halifax, NS, Canada, IEEE, Piscataway, NJ, USA, 2018, pp. 1153–1160.

[7] M. Bartoletti, B. Pes, S. Serusi, Data mining for detecting bitcoin Ponzi schemes, in: 2018 Crypto Valley Conference on Blockchain Technology (CVCBT); 20–22 Jun 2018; Zug, Switzerland, IEEE, Piscataway, NJ, USA, 2018, pp. 75–84.

[8] M. Vasek, T. Moore, There's no free lunch, even using bitcoin: tracking the popularity and profits of virtual currency scams, in: R. Böhme, T. Okamoto (Eds.), Financial Cryptography and Data Security, Springer, Heidelberg, Berlin, 2015, pp. 44–61.

[9] S. Bistarelli, M. Parroccini, F. Santini, Visualizing Bitcoin flows of ransomware: Wannacry one week later. Proceedings of the Second Italian Conference on Cyber Security; 6–9 Feb 2018; Milan, Italy, ITASEC, 2018.

[10] K. Liao, Z. Zhao, A. Doupé, et al., Behind closed doors: measurement and analysis of cryptolocker ransoms in bitcoin, in: 2016 APWG Symposium on Electronic Crime Research (eCrime); 1–3 Jun 2016; Toronto, ON, Canada, IEEE, Piscataway, NJ, USA, 2016, pp. 1–13.

[11] M. Spagnuolo, F. Maggi, S. Zanero, Bitiodine: extracting intelligence from the bitcoin network, in: N. Christin, R. Safavi-Naini (Eds.), Financial Cryptography and Data Security, Springer, Heidelberg, Berlin, 2014, pp. 457–468.

[12] C. Brenig, R. Accorsi, G. Müller, Economic analysis of cryptocurrency backed money laundering. Twenty-Third European Conference on Information Systems (ECIS); 26–29 May 2015; Münster, Germany, ECIS, 2015. Paper 20.

[13] M. Möser, R. Böhme, D. Breuker, An inquiry into money laundering tools in the bitcoin ecosystem, in: 2013 APWG eCrime Researchers Summit; 17–18 Sep 2013; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2013, pp. 1–14.

[14] Kristen Chapman, Riviera Beach Commissioners Vote to Pay Ransom to Hacker Who Shut Down City Computers, June 19th 2019. Available online: https://bit.ly/2TTulE0.

[15] E. Badawi, G.-V. Jourdan, G. Bochmann, et al., An automatic detection and analysis of the bitcoin generator scam, in: 2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW); 7–11 Sep 2020; Genoa, Italy, IEEE, Piscataway, NJ, USA, 2020, pp. 407–416.

[16] M. Bartoletti, S. Carta, T. Cimoli, et al., Dissecting Ponzi schemes on Ethereum: identification, analysis, and impact, Future Generat. Comput. Syst. 102 (2020) 259–277.

[17] T. Moore, J. Han, R. Clayton, The postmodern ponzi scheme: empirical analysis of high-yield investment programs, in: A.D. Keromytis (Ed.), Financial Cryptography and Data Security, Springer, Berlin, Heidelberg, 2012, pp. 41–56.

[18] F. Reid, M. Harrigan, An analysis of anonymity in the bitcoin system, in: Y. Altshuler, Y. Elovici, A. Cremers (Eds.), Security and Privacy in Social Networks, Springer, New York, NY, USA, 2013, pp. 197–223.

[19] K. Toyoda, T. Ohtsuki, P.T. Mathiopoulos, Identification of high yielding investment programs in bitcoin via transactions pattern analysis, in: GLOBECOM 2017—2017 IEEE Global Communications Conference; 4–8 Dec 2017; Singapore, IEEE, Piscataway, NJ, USA, 2017, pp. 1–6.

[20] H.S. Yin, R. Vatrapu, A first estimation of the proportion of cybercriminal entities in the bitcoin ecosystem using supervised machine learning, in: 2017 IEEE International Conference on Big Data (Big Data); 11–14 Dec 2017; Boston, MA, USA, IEEE, Piscataway, NJ, USA, 2017, pp. 3690–3699.

[21] M.A. Razali, S.M. Shariff, Cmblock: in-browser detection and prevention cryptojacking tool using blacklist and behavior-based detection method, in: Advances in Visual Informatics Conference, Springer, Cham, Switzerland, 2019, pp. 404–414.

[22] Jean-Luc, Liste d'escroqueries liées à bitcoin et aux cryptomonnaies - bitcoin.fr, Available online: http://bit.ly/2Pi5YN7, 2020.

[23] WaybackMachine, Wayback machine, Available online: https://web.archive.org/.

[24] B. Srinivasan, A. Kountouras, N. Miramirkhani, et al., Exposing search and advertisement abuse tactics and infrastructure of technical support scammers, in: WWW'18: Proceedings of the 2018 World Wide Web Conference; 23–27 Apr 2018; Lyon, France, ACM, New York, NY, USA, 2018, pp. 319–328.

[25] A. Kharraz, W. Robertson, E. Kirda, Surveylance: automatically detecting online survey scams, in: 2018 IEEE Symposium on Security and Privacy (SP); 20–24 May 2018; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2018, pp. 70–86.

[26] E. Badawi, G.-V. Jourdan, G. Bochmann, et al., Automatic detection and analysis of the "game hack" scam, J. Web Eng. 18 (8) (2020) 729–760.

[27] E. Badawi, G.-V. Jourdan, G. Bochmann, et al., The "game hack" scam, in: M. Bakaev, F. Frasincar, I.-Y. Ko (Eds.), Web Engineering, Springer, Cham, Switzerland, 2019, pp. 280–295.

[28] D. Jurafsky, J.H. Martin, Markov Assumption, 2014. Available online: stanford.io/29zsjAy.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, et al., Scikit-learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.

[30] E. Badawi, G.-V. Jourdan, Cryptocurrencies emerging threats and defensive mechanisms: a systematic literature review, IEEE Access 8 (2020) 200021–200037.

[31] K. Toyoda, T. Ohtsuki, P.T. Mathiopoulos, Time series analysis for bitcoin transactions: the case of pirate@ 40's hyip scheme, in: 2018 IEEE International Conference on Data Mining Workshops (ICDMW); 17–20 Nov 2018; Singapore, IEEE, Piscataway, NJ, USA, 2018, pp. 151–155.

[32] M. Conti, A. Gangwal, S. Ruj, On the economic significance of ransomware campaigns: a bitcoin transactions perspective, Comput. Secur. 79 (2018) 162–189.

[33] C.F. Torres, M. Steichen, R. State, The art of the scam: demystifying honeypots in ethereum smart contracts, in: 28th USENIX Security Symposium (USENIX Security 19); 14–16 Aug 2019; Santa Clara, CA, USA, USENIX Association, Berkeley, CA, USA, 2019, pp. 1591–1607.

[34] R. Phillips, H. Wilder, Tracing cryptocurrency scams: clustering replicated advance-fee and phishing websites, arXiv, 2020 preprint arXiv:2005.14440.

[35] S. Farrugia, J. Ellul, G. Azzopardi, Detection of illicit accounts over the Ethereum blockchain, Expert Syst. Appl. 150 (2020) 113318.

[36] A.A. Maksutov, M.S. Alexeev, N.O. Fedorova, et al., Detection of blockchain transactions used in blockchain mixer of coin join type, in: 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus); 28–31 Jan 2019; Saint Petersburg and Moscow, Russia, IEEE, Piscataway, NJ, USA, 2019, pp. 274–277.

[37] M. Möser, R. Böhme, Join me on a market for anonymity, in: 15th Workshop on the Economics of Information Security (WEIS); 13–14 Jun 2016; Berkeley, CA, USA, WEIS, 2016.

[38] T. Vissers, J. Spooren, P. Agten, et al., Exploring the ecosystem of malicious domain registrations in the. eu tld, in: M. Dacier, M. Bailey, M. Polychronakis (Eds.), Research in Attacks, Intrusions, and Defenses, Springer, Cham, Switzerland, 2017, pp. 472–493.

[39] C. Wei, A. Sprague, G. Warner, et al., Clustering spam domains and destination websites: digital forensics with data mining, J. Digit. Forensics, Secur. Law 5 (1) (2010) 2.

[40] O. Starov, Y. Zhou, X. Zhang, et al., Betrayed by your dashboard: discovering malicious campaigns via web analytics, in: WWW '18: Proceedings of the 2018 World Wide Web Conference; 23–27 Apr 2018; Lyon, France, ACM, New York, NY, USA, 2018, pp. 227–236.

[41] F. Poursafaei, R. Rabbany, Z. Zilic, Sigtran: signature vectors for detecting illicit activities in blockchain transaction networks, in: Advances in Knowledge Discovery and Data Mining, Springer, Cham, Switzerland, 2021, pp. 27–39.

[42] J. Ramos, Using tf-idf to determine word relevance in document queries, in: Proceedings of the First Instructional Conference on Machine Learning vol. 242, 2003, pp. 133–142.

[43] W. Chen, X. Li, Y. Sui, et al., Sadponzi: detecting and characterizing ponzi schemes in ethereum smart contracts, Proceedings of the ACM on Measurement and Analysis of Computing Systems 5 (2) (2021) 1–30.

[44] M. Bartoletti, S. Lande, A. Loddo, et al., Cryptocurrency scams: analysis and perspectives, IEEE Access 9 (2021) 148353–148373.

[45] A. Trozze, J. Kamps, E.A. Akartuna, et al., Cryptocurrencies and future financial crime, Crime Sci. 11 (1) (2022) 1–35.

[46] M.A. Ferrag, M. Derdour, M. Mukherjee, et al., Blockchain technologies for the internet of things: research issues and challenges, IEEE Internet Things J. 6 (2) (2018) 2188–2204.

[47] B. Abu-Salih, D.A. Qudah, M. Al-Hassan, et al., An intelligent system for multi-topic social spam detection in microblogging, arXiv, 2022 preprint arXiv:2201.05203.

[48] İ. Yurtseven, S. Bagriyanik, S. Ayvaz, A review of spam detection in social media, in: 2021 6th International Conference on Computer Science and Engineering (UBMK); 15–17 Sept 2021; Ankara, Turkey, IEEE, Piscataway, NJ, USA, 2021, pp. 383–388.

[49] Bitcoin wiki, Available online: https://en.bitcoin.it/wiki/From_address.