The Hacker's Code: Finding Bitcoin Thieves Through the Similarity and Status Claims Between Users

Chaitanya Katikala, CIPP
shaq.katikala@gmail.com

John Phillips
jophilli@stanford.edu

Andy Mai
andymai@stanford.edu

1. Introduction

Bitcoin (BTC), a form of digital currency started in 2009, functions through a distributed network that catalogues transaction activity throughout the network, resulting in a nearly frictionless medium of exchange. Similar to traditional currencies, bitcoins are still vulnerable to theft through Bitcoin wallet compromise. We apply the social network analysis tools of similarity and status against the problem of bitcoin thief re-identification through passive analysis on the BTC transaction history block-chain. We supplement previous research in the field with a two-part method and algorithm to improve thief re-identification by probabilistically associating thieves to public facing and associate addresses as well as social proof signaling between complicit or status aware actors. This information is used to build a list of suspect users, ranked by their status in the bitcoin thief network, which improves upon established association methodologies.

2. Prior Work

Bitcoin, a form of digital currency started in 2009 and, functions through a distributed network of public and private key exchanges that represent currency transactions. Thefts occur through security holes in intermediary holders or from direct attacks on computers storing "wallets," or a digital collection of private keys that connect a user to their public transactions. These thefts involve either some form of hacking of online storage sites or fraud of Bitcoin users. Our review of the available literature did not discover any research that viewed BTC users in term of degree of positive/negative links between each other.

Research by Reid and Harrison[1] demonstrated that while many believe that their identities remain secret when conducting business with bitcoins, matching published transactions with IP addresses collected from intermediaries could reveal the identity of a bitcoin user. The 2011 study revealed that 60% of users are exposed through this method. While they were able to reveal a significant amount of detail about the bitcoin thief, and provide Degree, In-Degree and Out-Degree for both user and transaction network structures, their application was retroactive and anecdotal. The Reid and Harrison paper does not attempt to discover new BTC financial fraud.

Androulaki, et al.[2], also worked to dismantle the aura of anonymity surrounding bitcoins. This study looked at two heuristics that are followed by later studies. First, Multi-input Transactions heuristic merges transactions that receive multiple inputs from different user, treating the sending users as one user. The second heuristic looked at "shadow" addresses, which incorporate the change address property of bitcoins.[3] The study then tested these heuristics in a simulated university setting and concluded that while the first heuristic cannot be easily evaded, the second heuristic can be evaded

[1] Reid, Fergal, and Martin Harrigan. "An analysis of anonymity in the bitcoin system." Security and Privacy in Social Networks. Springer New York, 2013. 197-223.

[2] Androulaki, Elli, et al. "Evaluating User Privacy in Bitcoin." IACR Cryptology ePrint Archive 2012 (2012): 596.

[3] "In the case when a Bitcoin transaction has two output addresses, aR_n, aR_o, such that aR_n is a new address (i.e., an address that has never appeared in pubLog before), and aR_o corresponds to an old address (an address that has appeared previously in pubLog), we can safely assume that aR_n constitutes a shadow address for a_i."

with some effort. The study concludes, “The privacy of users in Bitcoin can be compromised, even if users manually create new addresses in order to enhance their privacy in the system.” Under the simulation, the profiles of 40% of users could be revealed, even when they are all privacy-aware users. This is particularly interesting when considering that the Reid and Harrison study found that 60% of users can be revealed through methods that can be circumvented by sophisticated users. Androulaki, et al., however, is limited in that the sampled dataset was a simulation, making assumptions about how real-world users use multi-output transactions that fail to hold true. In addition, the study points out that the use of mixers (BTC Banks, BTC Anonymizers, etc.) is a real world easy solution to increase the privacy of bitcoin clients.

Meiklejohn et al.’s study essentially applies the heuristics of the previous study (Androulaki, et al.) to actual bitcoin data instead of a simulated dataset. Meiklejohn et al focused heavily on perfecting the change address heuristic (i.e., the 2nd heuristic of the previous study) by closely analyzing the idioms of use. They found the change address heuristic to be effective in some small experiments, but were able to conclude that that bitcoin thieves can circumvent this test. When examining Bitcoin thieves, the researchers were able to apply the change address heuristic and assumptions about bitcoin “peeling” to reveal some thieves’ identities. However, the ability to reveal these users varied depending on the thieves’ sophistication – i.e., whether they used complex layering and mixing to mask their identities. This supports our criticism of the dangers of applying conclusions from the Androulaki’s simulated dataset of “privacy-aware” users to real world applications. Meiklejohn’s study posed its own defects as well. The study selected known thieves retroactively and applied their method only to “a list of major Bitcoin thefts” that had public transactions, further increasing the likelihood of thieves that would go undetected by the change address heuristic.

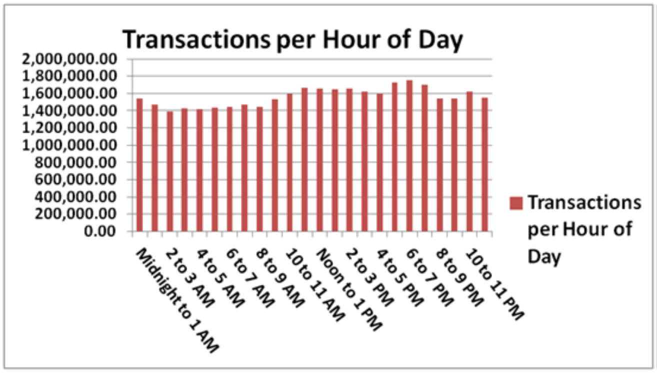
We build upon the previous studies by taking a different approach the problem. The previous studies used the transaction amounts and addresses to identify bitcoin users. However, the studies were retroactive or theoretical in nature, requiring one to already have found a bitcoin theft to flag a transaction as suspect. We attempt to assist in identifying transactions as suspect for Bitcoin fraud by two methods: through the similarity of bitcoin users and the claims to status after a heist is made. Under the similarity analysis, we group together users and accounts that may be correlated by the time, date, and transaction amounts. This is because hackers may attempt to circumvent detection by using multiple wallets or accounts but hackers groups may work together and thus be online at the same time, and share similarity of times and dates of Bitcoin transactions. Second, we look at coded messages embedded in transactions sent by users. Large heists are often difficult to convert to USD and smaller thefts lend themselves to motivations other monetary gain - we believe that other motivation is fame in the hacker community and users will send coded messages signaling their accomplishments in transactions amounts (e.g., 1337 in leetspeak).

3. Preparing Data

The entire bitcoin transaction history is publicly available in the BTC block-chain. Despite this, the block-chain is of significant size that performing research on it can become expensive. We use the preprocessed BTC dataset available from previous research conducted at the University of Illinois Urbana-Champaign (UIUC). This dataset is approximately 1.5GB large and comprises all BTC transactions from the network’s inception through to April 7th of 2013. In order to find specific data on Bitcoin thefts and fraudulent bitcoin transactions, we researched online forums and new articles that discussed such malicious Bitcoin events. However, despite this research we were only able to find specific identifying information, such as the transaction ID, on a portion of the overall volume of the known malicious bitcoin events. We speculate that some Bitcoin users are hesitant to provide such information in public forums. Nevertheless, our research yielded a list of more than forty individual malicious events, which we believe to be a sufficient level to show the validity of our analysis. In this paper, user nodes are referred to using their UIUC dataset reference numbers and not their actual Bitcoin addresses. Bitcoin addresses can be recovered using the UIUC dataset.

4.1 Method for Similarity Analysis

Although Bitcoin transactions can occur without physical proximity, some transactions are likely to occur in physical proximity to other users in order to coordinate the trade of Bitcoins for goods, cash, or consumables. In addition, there's a strong likelihood that peer groups of Bitcoin users will know more details about each other than is available publicly. While some users may experience less variation in transaction characteristics, some individuals along with brick and mortar service providers, and businesses adhere to specific transaction characteristics due to schedule and availability constraints. Thus we developed a probabilistic association model of similarity based on multiple methods of similarity detection such as the hour of day of the transaction as well as the day of the month/week and transaction amount size (both in BTC and USD) which seeks to associated thieves on with second degree contact who share a similarity profile with the thief and therefore may indeed be the thief or be affiliated with the thief.



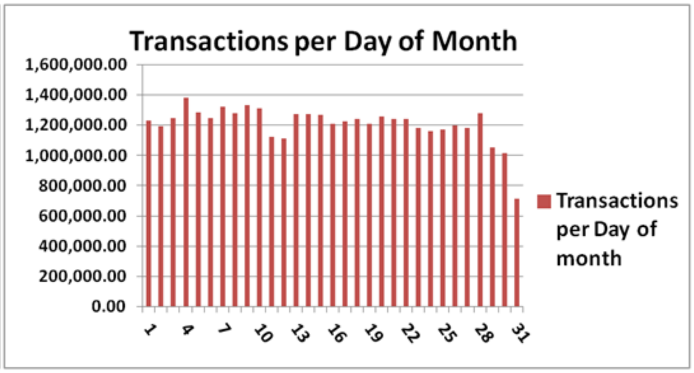


Chart 1: BTC Transactions per Hour of Day (UTC) **Chart 2. Transactions per Day of Month**

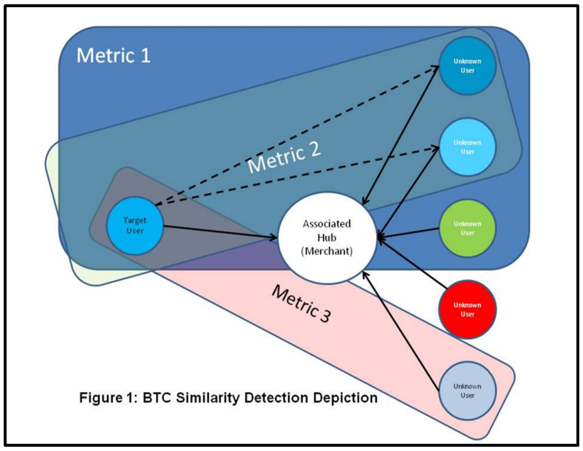


Figure 1: BTC Similarity Detection Depiction

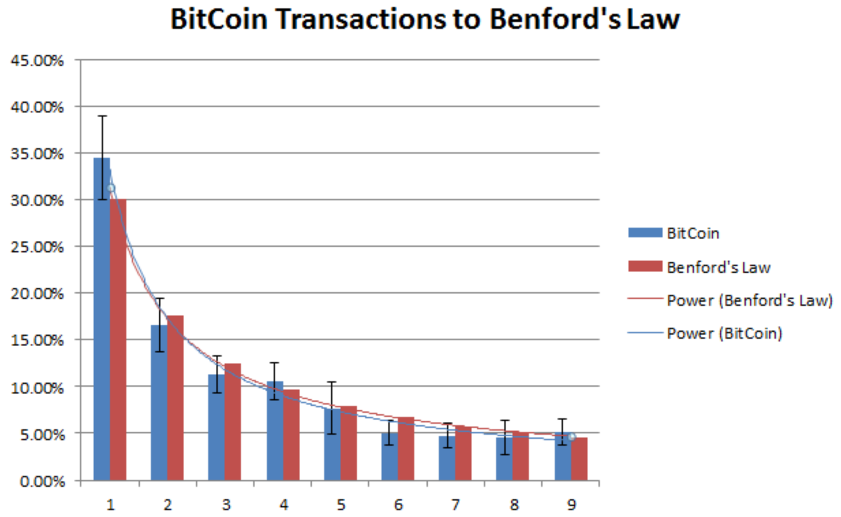
An initial examination of these similarity measures found that users within the bitcoin network do exhibit non-random characteristics of detection and thereby avail themselves to similarity based algorithmic analysis. Viewing the entire Bitcoin transaction history we detect a modest increase in activity during the mid-morning hours UTC and another increase in the early evening UTC. Given that these times correlate with the morning and afternoon hours for the US, which during the early years of BTC's existence was its predominant usage base, this is unsurprising. Additionally, a small damping effect appears to be observed in the mid part of each month.

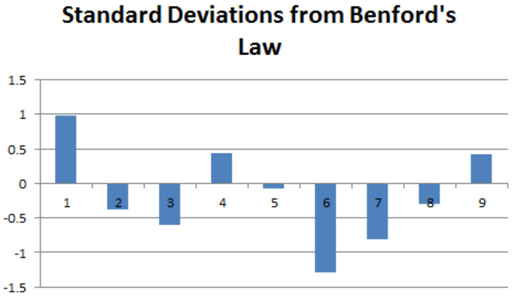
In constructing a similarity based detection algorithm we observed for each target user/node a number of discrete transaction metrics including 1) Size of transaction in BTC 2) Size of Transaction in Dollars (estimated with weighted daily averages) 3) Day of Week 4) Day of Month 5) Time of day and 6) Hour of day. We then compared each transaction from each first-degree connection node (to the thief node) and compared that transaction to every other inbound transaction to the first-degree connection node. We checked these other inbound transactions for similarity with the thief node transaction for similarity across each of the six characteristics, saving each node that triggered a given number of similarity measures against the thief's transaction. By doing this we establish high probability positive associations between separate transactor nodes and the target node that may have affiliation or be candidates for address consolidation.

4.2 Method for Status Analysis

Finding Coded Messages using Benford's Law

Benford’s Law, also known as the First Digit Law, states that the distribution of first digits in transactions amount in certain datasets will follow a power distribution as illustrated below. This property holds in financial transactions when the transactions amounts are determined by volatile pricing or market forces, as opposed to marketing (e.g., $0.99) or otherwise manually determined prices (e.g., $5.00). As Bitcoin transaction amounts are heavily determined by market forces (exchange rates, etc.) as opposed to manually determined prices, we expect the data to follow Benford’s Law. To improve the accuracy, we removed transactions that only had one non-zero digit (e.g., 0.004) from our dataset.





We found a close fit of Bitcoin transaction data the Benford’s curve, with all digits being less than 1.5 standard deviations away from Benford’s Law. Knowing the data fit the Benford’s Law curve closely, this provided a good baseline from which to test for frequently occurring deviant transaction amounts. For each user, we calculated their standard error from Benford’s Law from their received transactions. We focused on received transactions because an individual user is less likely to send a large number of status claims than a receiver is likely to receive a large number of them. To handle small samples size (users with only a few incoming transactions), we set a minimum of 6 received transactions and created a penalty multiplier for users with less than 30 incoming transactions of (1/ (30-N)), where N is the number of transactions. We then excluded users with less than 1.5 standard deviations from Benford’s Law in order to focus on the deviant transactions.

Our analysis counted the number of times every numerical phrase with 3 or more digits appears in the transaction amounts. For example, a transaction of 23.83 would add 1 count of 238, 383, and 2383. Although this analysis was expensive, it generated insightful files containing the most common phrases appearing in the data. We manually searched through the list to discover patterns of numbers that exceeded phrases we might expect through Benford’s Law, such as 100, 00002, etc. By grouping together similar phrases from low numbers of digits to high numbers of digits (e.g., 337, 1337, and 31337), we were able to construct a list of new suspected coded messages in the Bitcoin network. The search was very successful in revealing disproportionately occurring phrases in the data - the results are described below. We only searched through the top 50 3-digit, 4-digit, and 5-digit phrases and the longer phrases they are derived from. Future research may incorporate a full extraction of all coded messages.

Developing a Suspicion Score

The next step was to give rank the users by a combination of their deviation from Benford’s Law and the number of disproportionate status claims received. This would give strong weight to a user that received many irregular transactions and for which we identified that many of them are in fact coded messages. We used an approximation of Benford’s Law for multiple digits for determining the expected value for coded phrases to appear in a user’s received transactions, as shown below. If the number of phrases exceeded the expected amount, then our function only returned the number of claims that exceeded the expected amount.

$$E(Coded\ Phrases\ Received\ by\ User) = \sum_{\mathrm{i}=1}^{\mathrm{N}} \sum_{i=1}^{X} P_s \log_{10}(1 + \frac{1}{x})$$

X=number of coded phrases
P_s=positions that a phrase can appear in a number string (length of string +1 - length of coded phrase)
x=coded phrase
N=number of transactions received by a user

We needed what we call a suspicion score to give exponentially increasing weight as a user received higher numbers of transactions exceeding the expected amount. We saved only suspicion scores greater than 1, which limited users that received no coded messages to those with a standard deviation of *e*:

$$Suspicion\ Score = \log B(H+1)^{10(\frac{H}{T})}$$

H=Coded messages received by user (above expected value)
T=Total Transactions received by user
B=Average STD from Benford's Law for user's received transactions

Sample of Results for Top Suspicion Scores

User ID	Suspicion Score	Phrases Exceeding Expected Amount	Total Number of Transactions	STD from Benford's Law
304467	44.8744207451	159	187	5.59499994618
162286	41.1325972619	4084	8587	4.88353935813
489726	39.4922194061	2913	6137	5.08910613525
1393855	38.9955970536	255	375	3.62694403902
481128	38.6100255898	442	733	6.46056480133

Creation of Status Network

The suspicion scores only provided a ranking of users *receiving* coded messages. We are interested in the users that sent those coded messages since we expect some of them to be hackers or thieves laying claim to status. We included any user that sent at least one transaction to someone on the top 1,000 of the suspicion score list and created an edge that represented a coded term. Interestingly, there was only a 10% difference in the number of included users included in the Top 1000 List and the full list (~63,000), but the former avoided including many noisy nodes. Future research may consider also analyzing infrequent deviant transactions by including edges from users whose outgoing transactions deviate from Benford's Law to users with high suspicion scores.

5.1 Results for Similarity Analysis

For our battery of thief transactions, the table below shows for varying number of similarities which were set as a trigger levels (out of the six described) for a varying tightness of focus around the given measure, what percent of the original list of second degree contacts remain as candidates for possible consolidation/affiliation with the thief after performing our analysis. The figures listed are the averages of our similarity algorithm run against all of the thief transactions that we were able to find. For example, if a trigger similarity number is set to 3 and the tightness of similarity required for a trigger is 0.5 standard deviations, then for all thief events, the number of second degree users which had at least three measures of similarity fall within an absolute distance of .5 standard deviations from the given first degree node (for both the thief to first degree node and subsequent first degree to second degree node) per similarity type was -on average- 45.39% of the starting list of second degree nodes.. Thus we count a lower number averages a more focused set of potentially affiliated nodes, given the -assumably- high costs of pursuing a large number of second degree connections vs. a small number of list candidates.

The gradient effect of the reduction in the second degree list to the consolidated list as we increase the number of similarities and the tightness of the events gives us good reason to conclude that our results match our goal of providing a reduced set of probable nodes for affiliation and/or consolidation with the given thief address. That said, finding ways to validate the consolidated lists continues to prove challenging as it would require the ability to definitively evaluate the consolidated list against a given thief's actual similarity to the users it transacts with, which requires additional assets and capabilities to explore.

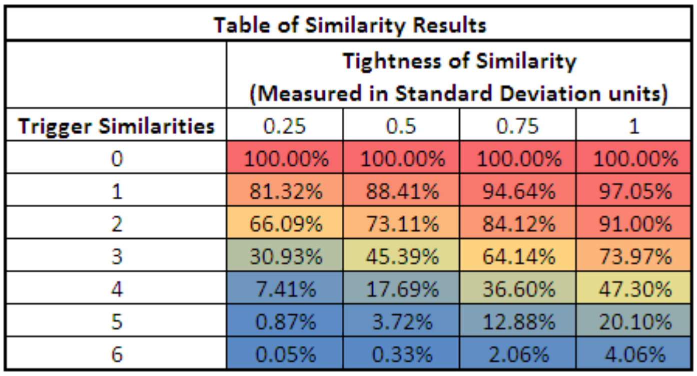


Table of Similarity Results

	Tightness of Similarity (Measured in Standard Deviation units)			
Trigger Similarities	0.25	0.5	0.75	1
0	100.00%	100.00%	100.00%	100.00%
1	81.32%	88.41%	94.64%	97.05%
2	66.09%	73.11%	84.12%	91.00%
3	30.93%	45.39%	64.14%	73.97%
4	7.41%	17.69%	36.60%	47.30%
5	0.87%	3.72%	12.88%	20.10%
6	0.05%	0.33%	2.06%	4.06%

5.2 Results for Status Analysis

Our Benford's Law analysis test was able to detect a 31 terms that appear more frequently than terms expected to appear frequently by Benford's Law. Here is a summary of those terms:

Coded Messages

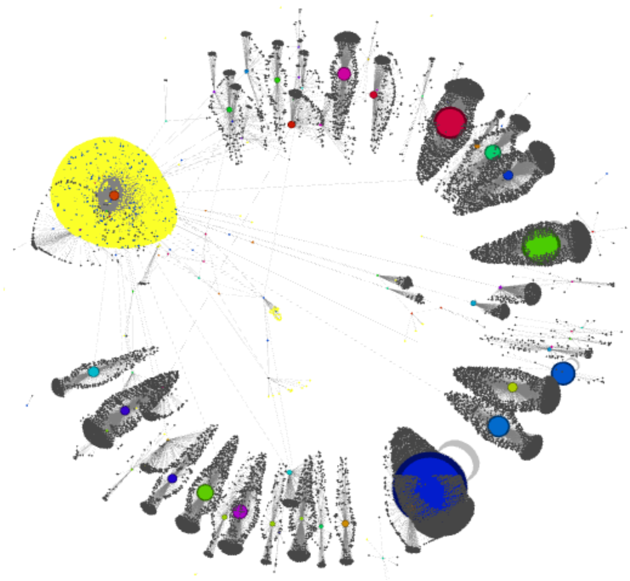
Category	**# of Phrases**	**Number of Times Phrase(s) is Used**
Node 25 Phrases (explained below)	11	92,958[4]
LEET (Elite)	9	980
Suspected Hacker Group 1	2	40[5]
Suspected Hacker Group 2	2	893
Suspected Hacker Group 3 (possibly Anonymous)	2	108
Phrase that possibly means "target"	3	204
Unknown	2	21

Graph Analysis

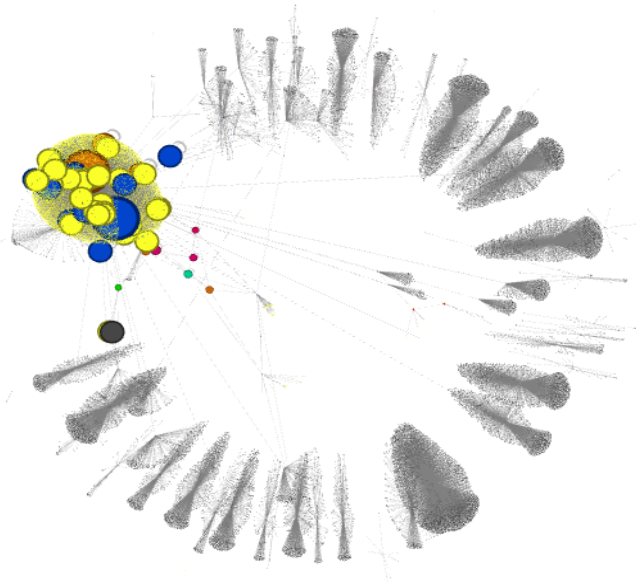
Network of Coded Messages

[4] Used 4,106 times by users other than Node25

[5] This was used more frequently than the terms relating to other hacker groups, but was not sent to nodes with high suspicion scores.



Node Size Proportional to its PageRank Score

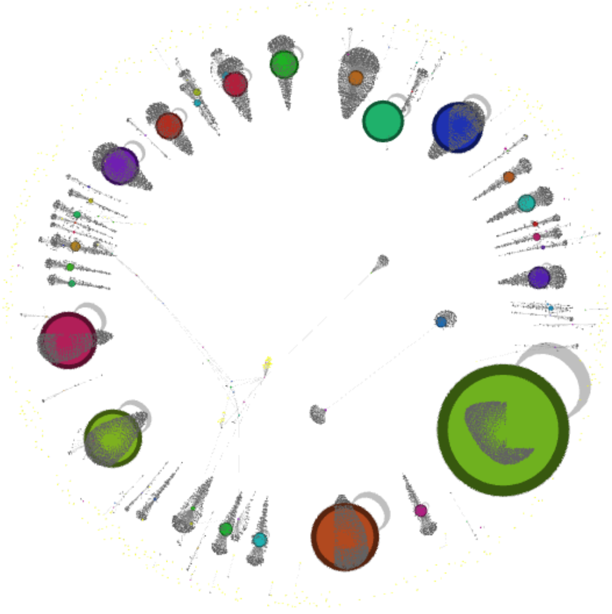


Node Size Proportional to its Clustering Coefficient

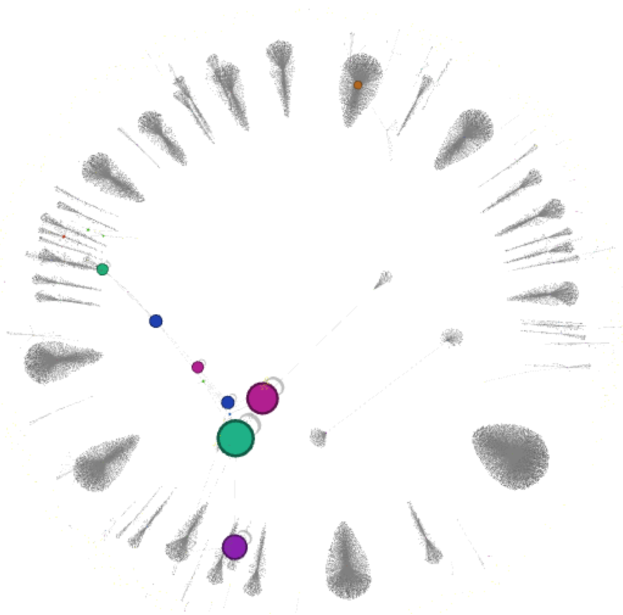
Because of the size of the graph, the visualization had to be zoomed out significantly. The black areas are large groups of nodes and edges consisting of hundreds or thousands of nodes. Black nodes and edges have no incoming edges. Mostly, we notice groups of such nodes in an orb shape all directed toward a central node, giving that central node a very large PageRank score and thus, size, in the left graph. In addition, nodes with 1 in degree are colored yellow. The yellow and red grouping of nodes at 10:00 on the graph is a central node (the red dot) sending coded messages out mostly to the yellow group of nodes around it. This group deserves attention because it deviates significantly from the rest of the graph, as shown in visualization of the Clustering Coefficient graph.

Unlike the other groups of nodes, node 25 (the central node in the yellow cluster) is *giving* a significant number of hacker signals to other users, not receiving them. It has an In-Degree of 42 and Out-Degree of 9002. Node 25 sends 11 different phrases very frequently, sometimes repeatedly the same users - the meaning of these phrases are unknown. Although others use these 11 terms as codes as well, Node 25 sends these phrases significantly more than others do. For example, node 25 used the phrase “118506” 12376 times while the rest of the users used the term 704 times (still a significant number for a 6 digit long phrase). This user also receives a significant amount of coded phrases, at least enough to receive a top 20 suspicion score. The clustering coefficient shows that the users affiliated with node 25 occasionally send codes amongst themselves, suggesting a community where users communicate through phrases. Node 25 also expands beyond its local community, as shown by the edges spanning the center of the graph. It and its network have many weak ties with the rest of the network, often pointing directly to or from nodes with high in-degrees. Thus, we believe that node 25 is a leader in a communication network of coded messages. An in-depth investigation in the nature of node 25 may alternatively reveal an odd payment arrangement for consumer transactions or some other strange characteristics. Once we remove node 25, we achieve a graph with a uniform pattern:

Network of Coded Messages after Removing Node 25



Node Size Weighted by PageRank Score



Node Size Weighted by Clustering Coefficient

The above diagrams show a graph with very high modularity, as node 25 was providing most of the graph's weak ties. Removal of just that one node significantly increases the modularity from 0.860 (71 communities) to 0.932 (362 communities). The largest node in the diagram above (the large green node at 4:00) is representative of user 493524, who received 8446 hacker terms *above the expected amount* out of 22103 total incoming transactions (38.2% of its transactions were suspicious). Like the nodes surrounding 493524, nearly all users in the graph have no incoming nodes and send their messages to only one user, as we see largely isolated groupings of nodes. This structure describes users that only send coded messages to their one favorite group. This may suggest allegiances, affiliations to a hacker group (represented by a user or node), locality, or other similarities. Applying the similarity analysis to the top 50 PageRanked users, we find a high level of similarity for each node. A mean of 97.73% (with a minimum of 90.5%) of nodes are two hops away are within 1 standard deviation of at least 3/6 triggers. This is compared to the results in 4.1, where for the same given standard deviation and trigger amount, the result included only 73.93% of nodes. This means that the nodes receiving a high number of coded messages are likely to share other characteristics with those users, perhaps implying geographic locality or cultural locality, based on similar purchase amounts. Locality can help explain the high modularity in this graph.

Ranking the Users

We used PageRank to determine the reputation of each node. PageRank captures the nature of the status claims - when one user of high status seeks status from another node, we expect the latter is of higher status than the first. When a reputable user begins sending much status to others, we want those claims to diluted. Thus, if we assume that coded messages are all hacker terms and those receiving the terms are hackers, we produce a list based on PageRank scores that ranks hackers or thieves by their reputation. For the filtered list of only the top 1000 suspicion scores and excluding node 25 from the analysis, we return the following ranking for the top 10 users: 493524, 491900, 489726, 162286, 507859, 518110, 491931, 491307, 126106, 89936. Additional tests will be able to determine if hubs for codes are likely affiliated with the same hacker group. If a node were to receive a disproportionate number of suspicious transactions and nearly all of those transactions are the same code, then we can group that node with others that affiliated with the same code.

6. Conclusion and Future Work

Evaluation

The similarity analysis returned results as expected and appears promising to lower the number of suspects when seeking affiliates with a known thief. While the analysis proves effective theoretically, empirical evaluation of this method against real world data requires in-depth knowledge about the groupings of nodes - data that is unavailable at this time.

Detailed information about bitcoin thieves online is scarce, but one source collected a list of known transaction IDs that relate to a known major theft. Because no further information is given regarding these transactions, it's unknown whether the source ID or the destination ID is a thief. However, we expect the destination user of a malicious transaction to be the recipient of the theft, and thus the thief or hacker. In the abbreviated list (nodes connecting to top 1000 suspicious receivers), no destination IDs were detected on only one source ID was found, but it had the 94th highest PageRank, making it the 99.68% highest PageRank. Full list was able to find 2 destination IDs and 5 source IDs (all of which were in the top 10% of the PageRank). Our results thus are thus inconclusive on their effectiveness in finding actual bitcoin thefts. It is not known positively known that any of the known thieves in our list have sent coded messages.

Difficulties:

Our major difficulty was the difficulty of testing our data against a strong control groups. While information about Bitcoin is available, the only detailed list we were able to find for known hackers contained only 40 actual transaction IDs but did not state the nature of those transactions. We thus were required to assume that destination IDs were the thief IDs and evaluate our results from there. Similarly, little data is available to test the actual effectiveness of our similarity measures. A close study of small groups of identified international users may be effective in proving the similarity algorithm shows geographical or other types of locality in the users.

Future Work:

The status test revealed the peculiar nature of node 25, suggesting a rather large communication network. Node 25 also significantly decreased modularity in the network. It's worth investigating Node 25 for the meaning of its highly used hacker terms, why Node 25 sends signals to other users, and why it's so well connected to the rest of the graph. Removing Node 25, nodes with high clustering or Betweenness centrality are also worth investing further, since they are providing the weak ties in the graph and possibly suggest a communication network.

Additional research in the area of similarity might expand the number of degrees of separation that our similarity searches use as initial screening pool from which to compare transactions from thieves. Separately, future research might attempt to formulate new similarity metrics, though no such metrics present themselves to us at this moment.

Conclusion

Our similarity algorithm yields results that are significantly smaller than the possible comparison sets of the second-degree connection (e.g., ~4% of users given 1 standard deviation and 6 similarity metrics). As a result, we believe this algorithm provides a potent tool for interested parties who want to identify bitcoin thieves.

We discovered Bitcoin transactions fit very closely to Benford's Law, and we were thus able to detect 31 coded messages in the Bitcoin network. Using these phrases, we detected a potential community of users communicating through coded phrases (i.e., node 25 and its neighbors). We discovered the graph is highly modular, especially if node 25 is excluded. Combining PageRank with both analyses, we are able to support a theory that localized segments in the coded messages network are highly similar, possibly suggesting geographic or cultural closeness for each segment. Although we discovered much about the nature of the Bitcoin network - that it's highly segmented and localized - our results are inconclusive about whether our methods are capable of detecting actual Bitcoin thefts, as our data for evaluation was limited. More data about Bitcoin thefts would be useful to test our results, such as an actual investigation to the most highly ranked nodes in the graph.