

The Evolution of Embedding Metadata in Blockchain Transactions

Tooba Faisal
University College London
London, UK
tooba.hashmi@gmail.com

Nicolas Courtois
University College London
London, UK
n.courtois@ucl.ac.uk

Antoaneta Serguieva
nChain, and LSE Systemic Risk
London, UK
antoaneta@ncrypt.com

Abstract—The use of blockchains is growing every day, and their utility has greatly expanded from sending and receiving crypto-coins to smart-contracts and decentralized autonomous organizations. Modern blockchains underpin a variety of applications: from designing a global identity to improving satellite connectivity. In our research we look at the ability of blockchains to store metadata in an increasing volume of transactions and with evolving focus of utilization. We further show that basic approaches to improving blockchain privacy also rely on embedding metadata. This paper identifies and classifies real-life blockchain transactions embedding metadata of a number of major protocols running essentially over the bitcoin blockchain. The empirical analysis here presents the evolution of metadata utilization in the recent years, and the discussion suggests steps towards preventing criminal use. Metadata are relevant to any blockchain, and our analysis considers primarily bitcoin as a case study. The paper concludes that simultaneously with both expanding legitimate utilization of embedded metadata and expanding blockchain functionality, the applied research on improving anonymity and security must also attempt to protect against blockchain abuse.

Index Terms—bitcoin, bitcoin cash, blockchain, cryptographic key management, embedded metadata, anonymity, privacy, ransomware, multisig.

I. INTRODUCTION

The use of blockchains is expanding from transferring crypto-coins to implementing smart-contracts that service a variety of domains. IBM and Sovrin are designing and implementing a global digital identity layer enabled by blockchain: decentralized, point-to-point exchange of information about people, organizations, or things. [1] EtherSat is developing a protocol for satellite connectivity utilizing blockchain: a decentralized global area network that maximizes efficiency of existing ground-station infrastructure. [2] nChain is creating a blockchain tokenization layer to enable interactivity and interoperability among smart contracts underlying various services [3] [4]. These are only few examples of how the technology is expanding. Throughout the initial and the expansion stages, the ability of a blockchain to store metadata has been exploited in an increasing volume of transactions and with evolving focus of utilization.

A. Embedding Metadata

Reviewing historically, and based on the bitcoin blockchain primarily, this ability at first involved creating an Unspent Transaction Output (UTXO) that could never be spent. That was used with a focus on permanently and securely storing

information (such as notary data) not directly related to the current transaction. The destination bitcoin address in the locking script of such unspendable UTXO in a Pay-To-Public-Key (P2PK) and Pay-To-Public-Key-Hash (P2PKH) transactions was used as a freeform 20-byte field to store metadata, and the the transaction was recorded on the blockchain. [5] Then, concerns were raised that the unspendable outputs could never be removed from the UTXO database, causing the database to increase forever. In response, Bitcoin Core version 0.9 introduced the RETURN operator, explicitly creating such outputs as provably unspendable and excluded from the UTXO set. [6] Simultaneously, the allowance for metadata increased from 20 to 80 bytes. Thus, legitimate non-payment data could be stored on the blockchain without increasing the UTXO database. However, concerns were raised that non-payment data stored in OP_RETURN outputs could allow meta-protocols to run permission-free with criminal intent. That led to a large proportion of miners not processing OP_RETURN transactions, and a corresponding proportion of metadata not being recorded on the blockchain. The more recent trend is storing information in the redeem script of pay-to-script-hash (P2SH) transactions. P2SH were standardized with Bitcoin Improvement Proposal (BIP): 16, as a powerful new type of transactions that simplifies the use of complex script. [7] The hash of the redeem script is restricted to 20 bytes, but the script itself and the size of metadata are not restricted. These transactions are spendable, and the full script must be revealed when spending a P2SH UTXO. At this stage, the utilization of embedded metadata is focused on variety of applications such as tokenization, blockchain-enforced smart contracts, and related access to secure databases. Some concerns about ransomware remain.

Among the innovative uses of P2SH-embedded metadata are the tokenization of assets (tangible, intangible, divisible, and non-divisible), the blockchain-enforcement of smart contracts, the blockchain-recorded progress through the complex conditionality structure of a smart contract, the efficient blockchain-registered exchange of various tokenized entities underlying smart contracts, and the blockchain-recorded access links and access privileges to off-chain databases. [8] Such databases can be Distributed Hash Table (DHT) databases that store smart-contract templates, or conditions for the exchange of and the characteristics of entities underlying smart contracts,

or software programs implementing intelligent agents capable of controlling various types of smart contracts. The described utilization of embedded-metadata serves as middleware that supports the development and execution of any specific smart contract.

In this paper, we identify millions of recorded blockchain transactions embedding metadata and classify them according to meta-protocols they support. That allowed us to observe, from one of its many perspectives, the broader and important question about technology adoption. We further analyze empirically the evolution of embedding metadata, and suggest security steps towards preventing criminal use. Metadata are relevant to any blockchain, and our analysis is based on bitcoin primarily. Bitcoin has long historical data and the largest market-share, but is considered inert to innovation in more recent years.¹ Our conclusion is that simultaneously with both expanding legitimate utilization of embedded metadata and expanding blockchain functionality, the applied research on improving anonymity and security must continue with protecting against blockchain abuse.

B. Ecosystems

Bitcoin is the first and has long been the most popular cryptocurrency and blockchain. All β -transactions are recorded in the immutable, append only, blockchain data-structure, where the key features and elements include: blocks, transactions stored in the blocks, and inputs, outputs, lock-time included in each transaction according to the transactions' format. The unspent outputs UTXO are monitored by the miners in validating transactions, and each input and output contain scripts – locking, unlocking, redeem scripts. [9] Depending on the type of transaction – P2PK, P2PKH, multisig, P2SH – the scripts may contain public keys, hashed public keys, multiple public keys, signatures based on private keys, multiple signatures, metadata, hashed metadata, and OP codes. The entire transaction is hashed using SHA-256 and this hash typically serves as a globally unique Transaction ID (TXID). [10]

The bitcoin script language *Script* is a Forth-like stack-based execution language. Each transaction is processed by every bitcoin validating node and the node's validation software executes, independently for each of the transaction's inputs, the unlocking script in the input alongside a corresponding UTXO's locking script. A transaction is valid if the cryptographic puzzles in all UTXO referenced by its inputs are solved by the inputs' scripts, i.e. all spending conditions are satisfied. Then these UTXO are removed from the UTXO database but remain permanently recorded on the blockchain. [5] *Script* is a stateless and predictable language, and Bitcoin Core currently includes 174 active *Script* opcodes (and 15 disabled) including 14 reserved opcodes, of the following types: push-value, flow-control, stack-ops, splice-ops, bit-logic, arithmetic, crypto, locktime, template-matching, and reserved-words. [11] [12]

¹Bitcoin has however broken grounds for crypto-currencies, and new currencies, such as bitcoin cash BCH, are actively pursuing innovation.

The *Script* in the more actively innovating bitcoin-cash BCH-blockchain is introducing further opcodes, by re-designing and re-testing functionalities previously intended (to an extent) by now disabled bitcoin-script opcodes, as well as by introducing new functionalities. [13] The bitcoin-cash network is undergoing a protocol upgrade in May 2018, supporting on-chain scalability, new transaction signatures, and a new difficulty adjustment algorithm. [14] The blocksize limit is adaptable, with an increased default of 8MB, and quite larger sizes are being tested on the bitcoin-cash testnet. [15] A new SigHash reusable signing mechanism ensures replay protection under a chain split, an improved hardware-wallet security, and elimination of the quadratic hashing problem. It provides for users creating transactions with a fork-specific ID, which are invalid on forks lacking support for the mechanism. [16] A new difficulty adjustment algorithm allows miners to migrate from the bitcoin chain as desired and provides protection against hashrate fluctuations. [17] Multiple independent teams develop bitcoin-cash software, assisted by peer-review workgroups, in contrast to the single-group development of Bitcoin Core. The development of bitcoin cash is decentralized and the ecosystem is dynamic. The focus is on protocol developments and on building Software Development Kits (SDK) that provide for the implementation and support of smart contracts and applications. [18]

II. IMPROVING PRIVACY AND SECURITY

Extending blockchain functionality and legitimate utilization of embedded metadata demands effective protection against blockchain abuse. The effective protection is supported by active applied research on anonymity and security.

A. Privacy

Privacy is a key desirable feature of all public and some private blockchains. Adoption and usage of bitcoin demonstrates early developments in distributed P2P payment systems anonymity engineering, and the privacy levels offered by current bitcoin pseudo-anonymous ledger is not very strong [19] [20]. Improving this is a major and difficult problem. It is not obvious how to reconcile ledger transparency and the desire for better privacy, and there is no easy quick-fix solution. Some early solutions involve using one public key only once. Using many different keys per user immediately raises the question of key management. In order to avoid the necessity for regular backups of fresh private keys generated at random, deterministic key derivation functions have been introduced. Hierarchical Deterministic (HD) wallets [21] use Elliptic Curve mathematics in order to calculate the public keys without revealing the private keys. HD wallets also allow users to derive various keys in a deterministic way from a single human readable seed. Using several keys there at the same time, however, like joining payments made to several keys belonging to the same user, compromises privacy. [22]

In addition to HD wallets, there exist several other methods improving bitcoin anonymity.

- *Mixes*: Mixing services or tumblers can be used to improve the anonymity of users by taking their coins and exchanging them with coins of other users, while hiding their identities. These services charge commissions between 1-3 %, and also need to be trusted not to steal users' coins. [19].
- *CoinSwap*: This is a similar concept to Mixing. If Alice wants to pay Charlie, she can send her coins to Bob instead of Charlie, and then Charlie can send a fresh unrelated coin to Bob. In order to resolve the theft problem, a central authority can manage these swaps. If any of the three misbehaves, the swap may be resolved by using hash-locked transactions that are linkable in the public-ledger. [20]
- *Fair Exchange*: This method allows the users to hide their identities by exchanging coins. Ideally, the fair exchange requires that either both parties involved in the transition receive each others items or none do. [23]
- *CoinJoin*: In CoinJoin users collaborate and create transactions where inputs of several users are mixed together. The transaction is not valid and will not be accepted by the network until all the signatures are provided [24].

Further methods that require attention include stealth addresses and dark wallets:

a) *Stealth Address*: One way to break the linkability in blockchain is to ask a recipient for two destination addresses, and then make two transactions and broadcast them into the network few seconds apart [22]. This concept has been further improved leading towards Stealth Address (SA) techniques, which are forms of non-interactive key exchange protecting privacy of users receiving payments. The origins of these techniques could be tracked to [25] [26]. Instead of a public key, the payee advertises a long unique static identifier, which is used by the payer to generate one time address to send money. A stealth address do not appear in the \mathbb{B} -blockchain; instead, random ephemeral public keys are generated and used. SA addresses use the Diffie-Hellman key-exchange mechanism, which allows the sender and receiver to exchange information and jointly generate some ephemeral public keys. Only lower-level derived keys will appear on the blockchain.

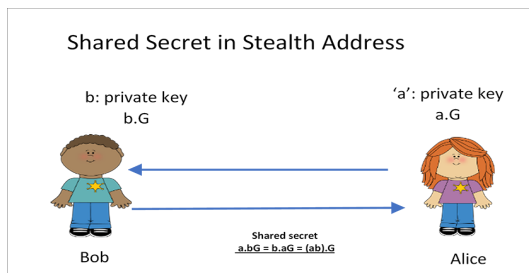


Fig. 1: DH key exchange in Stealth Address technique

For example, Bob advertises his (multiple-use) public key on his web page. There exist several variants of SA addressees. A basic method that uses permanent public/private identities of two participants, Alice and Bob, is adapted from [27] next. We

consider a basic Diffie-Hellman key exchange, as presented in Fig. 1. Let G be the generator point on Elliptic curve, and let 'a' and 'b' be the private keys of the sender (Alice) and the receiver (Bob), correspondingly. Alice and Bob use here their permanent identities that will typically appear on the blockchain (this will change in more advanced SA methods). Alice and Bob publish $a.G$ and $b.G$, correspondingly, and keep their private keys confidential. Alice computes $a.(b.G)$, and Bob equivalently computes $b.(a.G)$. Their shared secret is $S = a.(b.G) = b.(a.G) = a.b.G$, which no one else can compute. Next, Alice sends money to the *transfer address* E , and Bob detects the transaction and spends its UTXO. For example:

$$E = H(S).G$$

looks like a random \mathbb{B} -address. However, Bob knows the corresponding private key as he knows the common secret S , and can scan the bitcoin blockchain for E to appear.

This solution is still not quite secure, as Alice also knows the private key and may spend the UTXO before Bob. An improved asymmetric stealth address uses a stronger spending key e . [8] For example,

$$E = H(S).G + b.G \quad \text{and} \quad e = H(S) + b$$

Then, the sender Alice can no longer spend the transaction output, and can only compute the public key E . This method is still not ideal, as it is static and deterministic. In order to mitigate this, the sender can replace her permanent identity a by a random number r , and publish $r.G$ by typically using OP_RETURN in the very next output. In this case, one-time destination key and address are generated. This is not yet the best SA technique, and can be improved further by using 2 public keys, $b.G$ and $v.G$, where $v.G$ is a view key². Knowledge of the private part of the view key allows to build read-only wallets that can see transactions (undo anonymity) but cannot spend. Even a more robust stealth address has been proposed that protects against private key compromise, due to thefts, bad random attacks or Spectre/Meltdown type vulnerabilities [27]. Potential further development can use metadata based on processing various combinations of different partial biometric features, when generating signatures and keys. [28] Improved approaches to generating hierarchical asymmetric ephemeral keys and addresses have also been proposed and implemented in Nakasendo, an SDK supporting bitcoin cash applications, as well as applications for any blockchain based on elliptic-curve cryptography [18].

b) *Dark Wallet*: In order to enhance anonymity further, light-weight wallets that use both stealth-address and CoinJoin techniques have been created and termed Dark Wallets (DW). [29] Stealth addresses are discussed in detail in the previous section. A brief reminder on CoinJoin tells that a transaction of one user is combined with that of a random other user, who is making a payment at around the same time. Dark wallet is currently in its alpha testing state [30], as a Chrome

²View keys are also used in Monero and other CryptoNote-based currencies, and were first described in [25].

extension enabled in developer mode. During this study, it has been working on and off for brief periods of time.

B. Security

Innovative technologies are subject to abuse, as a series of incidents have demonstrated, including the recent Facebook data abuse by Cambridge Analytica affecting 87 million users. [31] Blockchain has also been abused, as the ransomware attack on UK NHS showed last year, affecting many people. [32] The adoption of bitcoin in ransomware crime is a major event of recent years [33]. Very recently, the first incident has been reported, as well, of a ransomware accepting bitcoin-cash payments. [34] In order to protect blockchain expansion into services benefiting the society, it is necessary to address the issue of its abuse. Innovation in terms of security and prevention from ransomware must be an integral part in the development of smart contracts and blockchain-based services. Next, we briefly introduce the key ransomware types and existing defenses. In later sections of this paper, we review them in relation to bitcoin and bitcoin cash, and suggest some solutions.

a) *Overall Rise of Ransomware*: Ransomware is a class of malware aiming to force users to pay a ransom in order to regain full access to their system [35]. This terminology covers a wide range of malicious software programs, including CryptoLocker, Locky, Cryptowall, KeyRanger, SamSam, TeslaCrypt, TorrentLocker and others [36] [19] The history of ransomware goes back to 2004, and the early software included screen lockers that were easy to remove or circumvent. Their level of sophistication, however, has been improving since then. Since 2013, a more harmful type of software has been developed. Though the programs are still called "lockers", they are not just lockers but quietly search for specific files and encrypt them, and then ask for ransom in order to decrypt. Only in the last few years, since bitcoin raised in popularity, ransomware has been combined with Bitcoin-payments. [33] [32]

- *CryptoLocker*: This is a well-known ransomware, since Sep. 2013. CryptoLocker v3 uses Advanced Encryption Standard (AES)-128 in Cipher Block Chaining (CBC) mode [36] and RivestShamirAdleman (RSA)-2048 for encryption of a header [37]. AES-128 is a symmetric key algorithm with 128-bit keys, and RSA-2048 is an asymmetric encryption algorithm using 2048-bit keys. This combination makes it most likely impossible to decrypt, without paying the ransom.
- *TorrentLocker Etc.* These are different strains of ransomware that have used AES differently: particularly in Counter (CTR) and CBC modes. [38]
- *TeslaCrypt*: This type of malware has been active since 2015, and provides customer support for the victims. It uses Elliptic Curve cryptography (ECC), an advanced key-derivation scheme, and has an ECC master private key that is later made public.
- *Locky*: Locky is a more recent and more sophisticated ransomware, since 2016. It uses Domain Generation Algorithm (DGA) to prevent blacklisting of domain names,

as well as custom encrypted communications. Locky also uses strong (RSA-2048 + AES-128) file encryption, and targets and encrypts over 160 different file types, including virtual disks, source codes and databases [39]. Locky has spread in two countries in particular, the United States and France, and uses The-Onion-Router (TOR) hidden servers. [36]

Further advanced ransomware techniques are discussed in Section IV-B. A recent study by IBM reports 6,000% of overall increase in ransomware in 2016 compared to 2015, and finds that 70% of business victim paid the hackers. [40]

b) *Ransomware Defenses*: There exist a number of OS-level countermeasures to avoid infection. Such measures include white-listing executables in user data directories [41], avoiding mapping backup drives [38], and disrupting the malware when using the Microsoft Crypto API [36].

- *Data Backups – False Good Solution*: It may seem that all ransomware is harmless if the data is backed-up on a regular basis and the back-up drives are encrypted. However, the problems go far beyond, and just restoring files or partitions from backups is not the best strategy. This destroys forensic evidence about how the malware propagates and how it operates, and makes the fight against malicious software more difficult. This also leaves our systems wide open and in the same state as before infection: they can be later re-infected by malware through the same channels. For example, a main infection channel for CryptoLocker was the Gameover Zeus botnet, which had existed earlier. [41]
- *Propagation of Ransomware*: Computer security experts must be able to monitor and analyze the infections. It is very useful to know how the ransom gets here in the first place. The problem is that there exist extensive and offensive expertise and experience, which have emerged over years of contrived action against the anti-virus industry. Different types of malware infection propagation and social engineering techniques are exploited to help criminals diffuse their unsolicited encryption payloads.

III. USE AND ABUSE OF BLOCKCHAIN TRANSACTIONS: EMPIRICAL ANALYSIS

A. Methodology and Results

The length of OP_RETURN script is currently 80 bytes, where the first two bytes always are hex 6a, followed by two bytes indicating the length in hex of the metadata-record that starts from byte number 5. With its protocol upgrade from version 1.0 to 1.1, on May 15, 2018, the OP_RETURN relay size will increase to 223 bytes, only on the bitcoin-cash blockchain. [14] Information about the two blockchains has been updated, corrected and analyzed in this study, by the contributing authors, based primarily on the OpReturnTool from [42].

The APIs of blockchain.info and coinsecrets.org are queried by this software. Blockchain.info is used only to get the latest block number. Then coinsecrets.org, a dedicated API

for OP_RETURN transactions, is queried to extract their Time stamp, Transaction ID, hash and ASCII code. The incoming data are recorded into a text file and exported to Excel [43]. Several methods were used, in order to identify the evolution of stealth-address techniques. We have performed experimental transactions, and further analysis on the observed patterns, to identify potential DW transactions and patterns. Among those that could not be related to a known protocol, there are transactions potentially related to criminal activities.

To identify the transactions from another wallet implementing stealth address, such as SX [44], its documentation has been consulted and observed patterns inside transactions are matched with our dataset. Overall, 22 protocols are identified and the rest of the OP_RETURN transactions are marked as unattributed. The prefix and ASCII for all the protocols are analyzed, and based on the analysis further three protocols have been identified: YEJ, BITCC, and Counterparty. However, their share inside the dataset is quite slim ($\approx 0\%$), and only Counterparty has been included for further analysis.

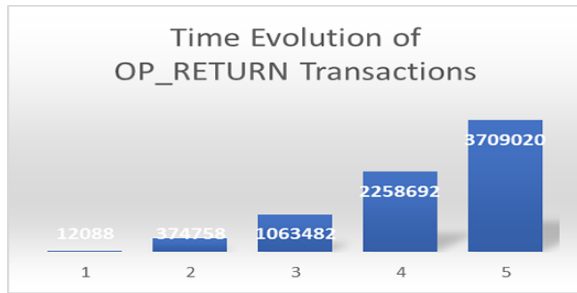


Fig. 2: Time evolution of OP_RETURN transactions

The first OP_RETURN transaction, identified in our study, appears on Mar. 29, 2013. This corrects [20], where the first such transaction is identified as appearing a year later. Thus, our dataset consists of data since Mar. 29, 2013 (block #228596) to Jul. 6, 2017 (block #474451). In 2013, only 430 OP_RETURN transactions are found, and all of them are in the unattributed category. From 2014 to 2017, we observe a significant increase in the volume of such transactions, reaching over 2 million per year in 2017. (see Fig. 2) A detailed examination shows that $\approx 51\%$ of these transactions correspond to the 22 known protocols, explained briefly in Table I. About 49% of the transactions remain unattributed.

The experimental dark-wallet transactions are identified in the unattributed section of the dataset, and have the prefix 6a-26-06, where 6a is the opcode for OP_RETURN, 26 is hex of the length of metadata that follows, and 06 distinguishes dark-wallet transactions from other protocols. The whole dataset is scanned and ≈ 2762 transactions are found with this pre-fix. The time evolution of transactions is illustrated with Fig. 3.

B. Analysis and Discussion

Linkability of transactions affects their anonymity, and several techniques have been adapted in the \mathbb{B} -systems to address that issue. Approaches such as CoinJoin, Fair Exchange and

TABLE I: Protocols Using op_return Opcode

Protocol	Contribution(%)	Usage
Unattributed	49%	Not identified
Blockstore	8.5%	Key value store
Factom	4.14%	Notary/Doc
Omni Layer	10.3%	Assets
Blocksign	0.06%	Notary/Doc
Colu	10.11%	Assets
Stampery	2.60%	Notary/Doc
Eternity wall	0.16%	Any Messages
Bitproof	0.03%	Notary/Doc
Open Assets	8.09%	Assets
Ascribe	2%	Digital Arts
Monegraph	2.7%	Digital Arts
Coinspark	1.1%	Assets
Proof of Existence	0.22%	Notary/Doc
Original My	<0.01%	Notary/Doc
Open Provenance	<0.01%	Proof of ownership
Remembr	<0.01%	Notary/Doc
Crypto copyright	<0.01%	Notary/Doc
LaPreuve	<0.01%	Notary/Doc
ProveBit	<0.01%	Notary/Doc
Blockchain Notary	<0.01%	Notary/Doc
Counterparty	<0.01%	Assets
Stampd	<0.01%	Notary/Doc

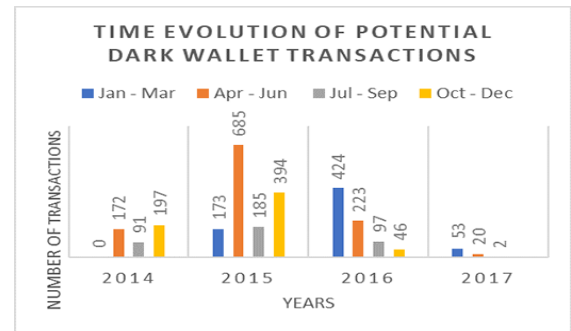


Fig. 3: Time evolution of potential DW transactions

CoinSwap, typically need a third-party involvement to achieve anonymity, and the honesty of the third party is not guaranteed. The latest stealth-address techniques seem currently effective. There, the receiver advertises its static, unique identifier and the sender generates a one-time key. There is no apparent way that a blockchain observer can relate transactions to the same payee. Stealth-address approaches are introduced to bitcoin and bitcoin cash, and have been used in monero and vertcoin. Stealth-address, dark-wallet and SX transactions appear as unable-to-decode by block explorers.

We have identified some DW and SX transactions among transactions unattributed to known protocols, in the OP_RETURN database we have extracted from the blockchain. It is noted that a smaller number of SX-related transactions are identified, as that protocol is less user-friendly than DW. A large number of transactions remain unattributed, and part of the issue is that meta-protocols are not required to coordinate and register unique identifiers. Therefore, many legitimate protocols don't use distinctive pre-fixes [45] that help decode/classify OP_RETURN transactions they produce,

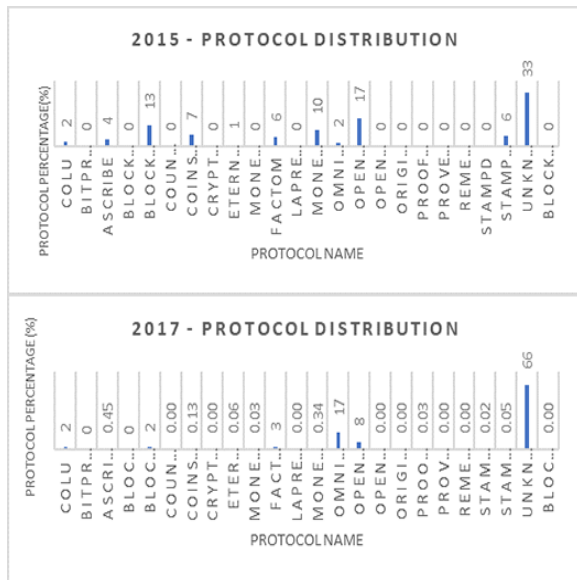


Fig. 4: The share of unattributed protocols is increasing.

and those transactions remain unattributed. This reason is even more valid now and in foreseeable future, as the anticipated proliferation of blockchain technology is through smart contracts that benefit users rather than ransomware them, and brings positive rather than destructive effect on society. Smart contracts are implemented and executed through embedding metadata in OP_RETURN and P2SH transactions, and therefore the number of transaction with embedded metadata will continue to rise. It is also valid to anticipate that criminals will exploit the new functionalities. We address both these issues in Section IV next.

IV. RANSOMWARE, KEY MANAGEMENT AND METADATA

A. Ransom Payments

Ransomware has always existed, but it has been associated with substantial risks to receive ransom payments without detection. With the increased popularity of Bitcoin in the last few years, criminals have started abusing the technology, in order to avoid detection. [46]. Receiving one or multiple ransom payments in ₿ allows very good initial anonymity. [33]. A new unique ₿ -address is created to receive payment from each victim. As long as the coins are not yet spent, there is no way to track who has received the ransom, and it can be spent in the future. [35] Once spending starts, the linkability of transactions is weakening anonymity and some transactions could be traced. [47] However, criminals can use various proxies, and carefully move and mix money in arbitrary ways for a long time, in order to diminish their chances of being traced.

Many companies and individuals will and do pay ransomware to get their data back, though advised the contrary by the authorities. Some individuals bought ₿ for the first time when became victims of ransomware, and some companies buy ₿ in advance to be able to pay in case of an attack.

[46] This affects the image of the technology, as illustrated by Google Trends during the attack on NHS last May. Fig. 5 presents the online interest in bitcoin and ransomware for the first two quarters of last year, and identifies they both spiked in May. [48] The image affects the development and adoption of innovative and legitimate blockchain-based services. The

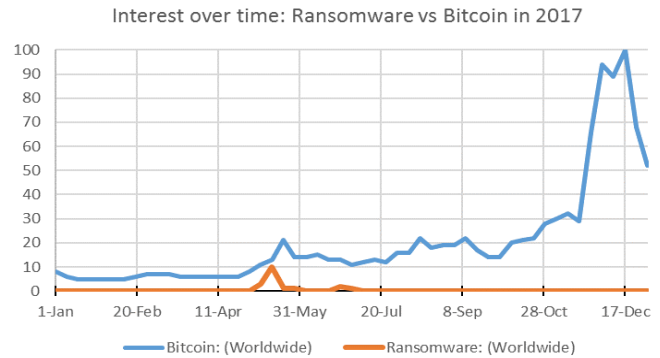


Fig. 5: Google search interests: Ransomware vs Bitcoin January–December 2017

technology is still moving forward, however, as Fig. 5, because the positive potential has been recognized. The target now is decoupling and disassociation from ransomware.

B. Public Key Generation and Diversification

A variant of the Curve-Tor-Bitcoin (CTB)-Locker, targeting web servers, generates a unique ₿ -address for every infection. Once the ransom is received, hackers produce a transaction using OP_RETURN and embedding a decryption key inside. [49] Other ransomware, such as the Locky payment system, rely on the anonymity features of TOR. It uses TOR hidden servers that remain operational years after the infection. The server software, presumably operating without human intervention, has automatically adjusted to ₿ -price over the years: initially asking for $\text{₿}2$, and recently for much smaller amounts. A Locky decryptor tool, made available to the victims on the same TOR website when they connect again after paying, is received on the blockchain [39]. One of the onion servers used with Locky is twbers4hmi6dx65f.onion, and we have observed that after payment they contain malware such as Variant.Zusy.185950. The Locky payments also seem to be automatically aggregated into larger pots of $\text{₿}50$ or 100 . An address associated with Locky ransom is 1Cjqt4C17sXYrWkrRyPr73RjrjZu1fuHVMV, though it is not clear if the address belongs to an exchange, a mixing service, or is still criminally controlled. If we look from this address backwards, the blockchain allows identifying victims of Locky who have paid ransom in standardized amounts of $\text{₿}1$ or 0.25 .

C. Future Risks and Mitigation

The combination of all topics we study above present a major risk: using stealth addresses, CoinJoin, TOR, and possibly smart contracts, in order to further automate and streamline the process of ransom threat and payment. We

are not there yet but in future, criminals can hide their identity behind the increasing number of active blockchain participants, while shuffling money around. Dark wallet has

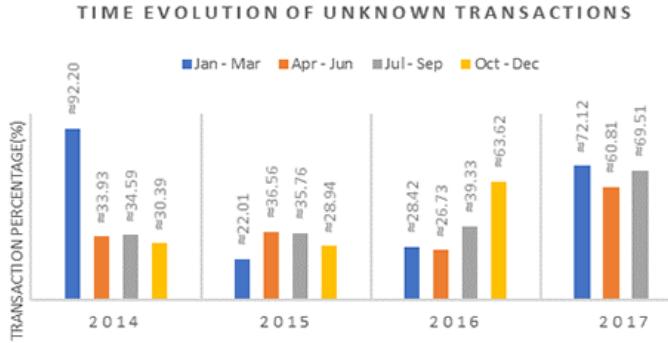


Fig. 6: Share of unattributed protocols.

been a prominent β -wallet [30] focusing on anonymity. It implements stealth-address with CoinJoin techniques. Yet DW deployment has raised concerns about its potential abuse. [29] This study has analyzed DW transactions and found that they cannot be associated with known protocols. The number of DW transactions is still low: the wallet is in its alpha state and has not recently been operating properly. [30] It can be expected, however, that a more dangerous DW-successor will be inevitably created. [26] We propose that companies using blockchain to implement complex protocols, should provide means of audit capable of distinguishing legitimate traffic they produce from potentially criminal activity.

Though bitcoin and bitcoin cash are studied here, all blockchains are imminently subject to abuse, and fall within the wider range of FinTech and SocTech technology targeted with criminal intent. Bitcoin has existed longer and had the largest market-share among blockchains. On the one hand, it has traversed through challenges to bring visibility to the technology and recognition of its potential, and opportunities to and drive for emerging competition. On the other hand, this position has made bitcoin most targeted by criminal activity: the wider and more active is a blockchain network, the better it can be both used and abused. Empirical history has also shown that the more a cryptocurrency is used, the more valuable it is. Further blockchains are getting momentum and raising their market-share, and are imminently attracting criminal intent. The other case-study in this paper, bitcoin cash, is representative of recently set up but high-momentum blockchains, and has risen to 4th market-share. Only one ransomware has been reported using bitcoin cash, seven months after its launch. This contributes to the observation that relative maturity is a precondition for targeted abuse of a blockchain.

With this paper, we raise the requirement that security and robustness against ransomware and other abuse must be of equal priority in innovation as are functionalities directly impacting market-share. Another factor is the rate of innovation itself. For example, blockchain cash is supported by several rather than one development team, and releasing a new protocol this May that introduces new op-codes, increases

blocksize, and provides a new algorithm stepping towards blockchains' interoperability. The need for innovation has been a main reason for the bitcoin cash fork. The development teams have started releasing SDKs to support the blockchain community in implementing higher-level apps, as well. Multiple teams and continuous innovation contribute to resistance to abuse. We have to note that a large part of identified transactions without attributed protocols are associated with legitimate meta-protocols, but there is no registry of meta-protocols to serve as a reference for identification. The use of P2SH transactions and the development of smart contracts are trends for a foreseeable future, and will expand significantly the range of meta-protocols. Therefore, we suggest that an off-chain DHT registry of meta-protocols is set up, and a corresponding unique indicator/identifier for each type of protocol is embedded in transactions that use/implement it. The DHT repository will have secure selective access, so that the meta-protocols are not compromised but allow corresponding level of audit. We anticipate that with the proliferation of smart contracts, the legitimate types of meta-protocols will greatly outnumber malicious ones. However, the system will still be vulnerable to large-scale outlier cyber-attacks, if decisive, expert, priority solutions are not developed, maintained and updated. If anticipation is that criminals will use smart contracts, then monitoring smart contracts can be developed to identify, protect against, and prevent their activity.

V. CONCLUSION

This paper addresses blockchain adoption from the perspective of the evolution and utilization of metadata, and therefore, from the perspective of the evolution of protocols that are implemented and executed through embedding additional information in blockchains. We identify, analyze and discuss reasons for the role of metadata, including empirical analysis; and suggest approaches towards protecting the expanding blockchain functionality against criminal abuse and ransomware. Challenges, in terms of exploiting/improving anonymity and prioritizing/raising security, are clearly stated, and intended and unintended consequences are addressed. We review key characteristics of ransomware and of expert approaches protecting against it. The paper suggests that blockchains should provide some level of transparency of what they are used for. We need to improve the auditability of blockchain transactions and smart-contract protocols. This can be facilitated for example by DHT databases of protocols, with secure and selective authorization or audit access. Another major lesson learned from our research on blockchain technology adoption and on metadata usage in blockchains, is to see that new technology must innovate and change all the time in order to near its full potential and that protocols and usage of blockchains must evolve with time. We need to continue to improve privacy of blockchains, while mitigating and monitoring their problematic or maybe criminal usage.

REFERENCES

- [1] Sovrin Foundation, "SovrinTM: A protocol and token for self sovereign identity and decentralized trust," Jan. 2018, Sovrin Founda-

- tion White Papers, [Online]. Available: <https://sovrin.org/wp-content/uploads/Sovrin-Protocol-and-Token-White-Paper.pdf> [Accessed: Apr. 15, 2018].
- [2] A. Cohen, L. Duncan and A. Edwards, "EtherSat protocol: A blockchain approach to efficient satellite connectivity," Proceedings of 2017 International Joint Conference on Neural Networks, Jul. 2018, forthcoming, IEEE Press.
- [3] nChain Holdings (C.S. Wright and S. Savanah), "A Method and system for the Efficient Transfer of Entities on a Blockchain". Patent PCT/IB2017/050859, Feb. 23, 2016, [Online]. Available: <http://patentscope.wipo.int/search/en/result.jsf> [Accessed: Apr. 15, 2018].
- [4] nChain Holdings (C.S. Wright and S. Savanah), "Registry and Automated Management Method for Blockchain-Enforced Smart Contracts". Patent PCT/IB2017/050865, Feb. 23, 2016, [Online] Available: <http://patentscope.wipo.int/search/en/result.jsf> [Accessed: Apr. 15, 2018].
- [5] A.M. Antonopoulos, "Mastering Bitcoin: Programming the Open Blockchain," 2nd Edition, 3rd Release, Mar. 23, 2018, O'Reilly Media.
- [6] Bitcoin Projecte, "Bitcoin Core version 0.9.0 released," Bitcoin Core, Mar. 19, 2014, [Online]. Available: <http://bitcoin.org/en/release/v0.9.0> [Accessed: Apr. 15, 2018].
- [7] G. Andresen, "Pay to script hash," Bitcoin Improvement Proposals, BIP: 16, Jan. 3, 2012, [Online]. Available: <http://github.com/bitcoin/bips/blob/master/bip-0016.mediawiki> [Accessed: Apr. 15, 2018].
- [8] C.S. Wright and A. Serguieva, "Sustainable blockchain-enabled services: Smart contracts," Proceedings of the 2017 IEEE International Conference on Big Data, pp 4255–4264, Dec. 2017, IEEE Press.
- [9] Bitcoin Wiki, [Online]. Available: <http://en.bitcoin.it/wiki> [Accessed: April. 15, 2018].
- [10] S. Ammous, *The Bitcoin Standard: The Decentralized Alternative to Central Banking*, Mar. 23, 2018, Wiley.
- [11] Bitcoin Project, "script.h," Github Bitcoin, [Online]. Available: <http://github.com/bitcoin/bitcoin/blob/master/src/script/script.h> [Accessed: Apr. 15, 2018].
- [12] Bitcoin Wiki, "OpCodes," Bitcoin Wiki Script, [Online]. Available: <http://en.bitcoin.it/wiki/Script#OpCodes> [Accessed: Apr. 15, 2018].
- [13] BitcoinCashOrg, "Restore disabled script opcodes," Github BitcoinCashOrg, [Online]. Available: <http://github.com/bitcoincashorg/spec/blob/master/may-2018-reenabled-opcodes.md> [Accessed: Apr. 15, 2018].
- [14] BitcoinCashOrg, "New features," Bitcoin Cash Features, [Online]. Available: <http://www.bitcoincash.org> [Accessed: Apr. 15, 2018].
- [15] BitcoinCashOrg, "Upgrade Testing," [Online]. Available: http://docs.google.com/spreadsheets/d/1_uRyqNnMEHogUdCY6WhCMoyuozZsyMtVm2R4xAsIeI/edit#gid=299033565 [Accessed: Apr. 15, 2018].
- [16] BitcoinCashOrg, "BUIP-HF: Digest for replay protected signature verification across hard forks," Github BitcoinCashOrg Spec, [Online]. Available: <http://github.com/bitcoincashorg/spec/blob/master/replay-protected-sighash.md> [Accessed: Apr. 15, 2018].
- [17] BitcoinCashOrg, "Bitcoin cash hardfork technical details, version 1.3," Github BitcoinCashOrg Spec, Nov. 7, 2017 [Online]. Available: <http://github.com/bitcoincashorg/spec/blob/master/nov-13-hardfork-spec.md> [Accessed: Apr. 15, 2018].
- [18] nChain Media, "Nakasendo™ Software Development Kit," Apr. 16, 2018, [Online]. Available: <http://nchain.com/en/media/nchain-releases-nakasendo-software-development-kit> [Accessed: Apr. 20, 2018].
- [19] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J. A. Kroll and E. W.Felten, "Mixcoin: Anonymity for Bitcoin with accountable mixes," Proceedings of the 18th International Conference on Financial Cryptography and Data Security, pp 486–504, Mar. 2014, Springer.
- [20] M. Moser and R. Bohme, "Anonymous Alone? Measuring Bitcoin's Second-Generation Anonymization Techniques," Proceedings of the 2017 IEEE European Symposium on Security and Privacy Workshops, pp 32–41, Apr. 2017, IEEE Press.
- [21] P. Wuille, "Hierarchical deterministic wallets," Bitcoin Improvement Proposals, BIP: 32, Feb. 11, 2012, [Online]. Available: <http://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki> [Accessed: Apr. 15, 2018].
- [22] Anonymous, "Reclaiming financial privacy with HD wallets," Bitcoinism, Jul. 5, 2013, [Online]. Available: <http://bitcoinism.blogspot.co.uk/2013/07/reclaiming-financial-privacy-with-hd.html> [Accessed: Apr. 15, 2018].
- [23] I. Ray and I. Ray, "An anonymous fair exchange protocol," Proceedings of the 15th International Parallel and Distributed Processing Symposium, pp 1790–1797, IEEE Press, 2001.
- [24] gmaxwell, "CoinJoin: Bitcoin privacy for the real world," Bitcoin Forum, Aug. 22, 2013, [Online]. Available: <http://bitcointalk.org/index.php?topic=279249.0> [Accessed: Apr. 15, 2018].
- [25] N. van Saberhagen, "CryptoNote v 2.0," CryptoNote Technology White Paper, Oct. 17, 2013, [Online]. Available: <http://cryptonote.org/whitepaper.pdf> [Accessed: Apr. 15, 2018].
- [26] Anonymous, "Untraceable transactions which can contain a secure message are inevitable," Bitcoin Forum, Apr. 17, 2011, [Online]. Available: <http://bitcointalk.org/index.php?topic=5965.0> [Accessed: Apr. 15, 2018].
- [27] N.T. Courtois and R. Mercer, "Stealth address and key management techniques in blockchain systems," Proceedings of the 3rd International Conference on Information Systems Security and Privacy, pp 559–566, SciTePress, 2017.
- [28] Y. Kaga, M. Fujio, K. Naganuma, K. Takahashi, T. Murakami, T. Ohki and M. Nishigaki, "A secure and practical signature scheme for blockchain based on biometrics," in: J. Liu and P. Samarati (eds), *Information Security Practice and Experience ISPEC 2017*, pp 877–891. Lecture Notes in Computer Science book series, vol 10701. Springer, 2017.
- [29] A. Greenberg, "'Dark Wallet' is about to make Bitcoin money laundering easier than ever," Wired Business, Apr. 29, 2014, [Online]. Available: <http://www.wired.com/2014/04/dark-wallet> [Accessed: Apr. 15, 2018].
- [30] Darkwallet Team, "Darkwallet build status," Github DarkWallet, [Online]. Available: <http://github.com/darkwallet/darkwallet> [Accessed: Apr. 15, 2017].
- [31] N. Lomas, "How Facebook has reacted since the data misuse scandal broke," Techcrunch, Apr. 10, 2018, [Online]. Available: <http://techcrunch.com/2018/04/10/how-facebook-has-reacted-since-the-data-misuse-scandal-broke> [Accessed: Apr. 15, 2018].
- [32] K. Collins, "Crime in RealTime," May 12, 2017, [Online]. Available: <http://qz.com/982993/watch-as-these-bitcoin-wallets-receive-ransomw-are-payments-from-the-ongoing-cyberattack> [Accessed: Apr. 15, 2018].
- [33] R. Brewer, "Ransomware attacks: detection, prevention and cure," *Network Security*, vol 2016, no 9, pp 5-9, 2016.
- [34] L. Abrams, "Thanatos ransomware is first to Use Bitcoin Cash: Messes Up encryption," *Bleeping Computer: security News*, Feb. 26, 2018, [Online]. Available: <http://www.bleepingcomputer.com/news/security/thanatos-ransomware-is-first-to-use-bitcoin-cash-messes-up-encryption> [Accessed: Apr. 15, 2018].
- [35] D. Sgandurra, L. Munoz-Gonzalez, R. Mohsen and E.C. Lupu, "Automated dynamic analysis of ransomware: Benefits, limitations and use for detection," arXiv preprint, arXiv:1609.03020, 2016.
- [36] A. Palisse, H. Le Boudier, J.L. Lanet, C. Le Guernic, A. Legay, "Ransomware and the legacy Crypto API," Proceedings of in 11th International Conference on Risks and Security of Internet and Systems, pp.11-28, Springer, 2017.
- [37] J. Doevan, "CryptoLocker: How to remove? (Uninstall guide)," Jul. 18, 2017, [Online]. Available: <http://www.2-spyware.com/remove-cryptolocker.html> [Accessed: Apr. 15, 2018].
- [38] C. Puodzius, "How encryption molded crypto-ransomware," *WeLiveSecurity*, Sep. 13, 2016, [Online]. Available: <http://www.welivesecurity.com/2016/09/13/how-encryption-molded-crypto-ransomware> [Accessed: Apr. 15, 2018].
- [39] Threat Intelligence Team, "A Deep and Technical Look into the Ransomware called Locky," Avast blog, Mar. 10, 2016. [Online]. Available: <http://blog.avast.com/a-closer-look-at-the-locky-ransomware> [Accessed: Apr. 15, 2018].
- [40] H. Taylor, "Ransomware spiked 6000% in 2016 and most victims paid the hackers, IBM finds," CNBC: Tech News, Dec. 14, 2016, [Online]. Available: <http://www.cnbc.com/2016/12/13/ransomware-spiked-6000-in-2016-and-most-victims-paid-the-hackers-ibm-finds.html> [Accessed: Apr. 15, 2018].
- [41] B. Krebs, "How To Avoid CryptoLocker Ransomware," *Krebs on Security: In-depth Security News and Investigation*, Nov. 2013, [Online] Available: <https://krebsonsecurity.com/2013/11/how-to-avoid-cryptolocker-ransomware> [Accessed: Apr. 15, 2018].
- [42] M. Bartoletti and L. Pompiaru, "An analysis of Bitcoin OP_RETURN metadata," arXiv preprint, arXiv:1702.01024v2, Mar. 1, 2017. (OpReturnTool, [Online], Available: <http://github.com/BitcoinOpReturn/OpReturnTool> [Accessed: Apr. 15, 2018].)
- [43] T. Faisal, *OP_ReturnToCSV and OP_RETURN-protocols*, GitHub, [Online]. Available: www.GitHub.com/tabz11 [Accessed: Apr. 15, 2018].
- [44] A. Taaki, S. Mendez and V. Buterin, "The SX tutorial," 2013, [Online]. Available: <http://sx.dyne.org/stealth.html> [Accessed: Aug. 8, 2017].

- [45] Bitcoin Wiki, "OP_RETURN," May 7, 2017, [Online]. Available: http://en.bitcoin.it/wiki/OP_RETURN [Accessed: Aug. 5, 2017].
- [46] P. Mccausland, "Companies stockpiling Bitcoin in anticipation of ransomware attacks," NBC: U.S. News, May 18., 2017, [Online]. Available: <http://www.nbcnews.com/storyline/hacking-of-america/companies-stockpiling-bitcoin-anticipation-ransomware-attacks-n761316> [Accessed: Apr. 15, 2018].
- [47] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker and S. Savage, "A Fistful of Bitcoins: Characterizing Payments Among Men with No Names," iInternet Measurement Conference, pp. 127-140, 2013.
- [48] Google, "Google Trends", [Online]. Available: <http://trends.google.com/trends/explore?q=%2Fm%2F05p0rrx,%2Fm%2F0657nv> [Accessed: Apr. 15, 2017].
- [49] L. Constantin, "Ransomware authors use the bitcoin blockchain to deliver encryption keys," PCWorld: Security, Apr. 14, 2016, [Online]. Available: <http://www.pcworld.com/article/3056607/ransomware-authors-use-the-bitcoin-blockchain-to-deliver-encryption-keys.html> [Accessed: Aug. 15, 2018].
- [50] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online].