# A Quantitative Analysis of the Impact of Arbitrary Blockchain Content on Bitcoin

Roman Matzutt[1], Jens Hiller[1], Martin Henze[1], Jan Henrik Ziegeldorf[1], Dirk Müllmann[2], Oliver Hohlfeld[1], and Klaus Wehrle[1]

[1] Communication and Distributed Systems, RWTH Aachen University, Germany,
{matzutt,hiller,henze,ziegeldorf,hohlfeld,wehrle}@comsys.rwth-aachen.de
[2] Data Protection Research Institute, Goethe University, Frankfurt/Main,
muellmann@jur.uni-frankfurt.de

**Abstract.** Blockchains primarily enable credible accounting of digital events, e.g., money transfers in cryptocurrencies. However, beyond this original purpose, blockchains also irrevocably record *arbitrary* data, ranging from short messages to pictures. This does not come without risk for users as each participant has to locally replicate the complete blockchain, particularly including potentially harmful content. We provide the first systematic analysis of the benefits and threats of arbitrary blockchain content. Our analysis shows that certain content, e.g., illegal pornography, can render the mere possession of a blockchain illegal. Based on these insights, we conduct a thorough quantitative and qualitative analysis of unintended content on Bitcoin's blockchain. Although most data originates from benign extensions to Bitcoin's protocol, our analysis reveals more than 1600 files on the blockchain, over 99 % of which are texts or images. Among these files there is clearly objectionable content such as links to child pornography, which is distributed to all Bitcoin participants. With our analysis, we thus highlight the importance for future blockchain designs to address the possibility of unintended data insertion and protect blockchain users accordingly.

## 1 Introduction

Bitcoin [45] was the first completely distributed digital currency and remains the most popular and widely accepted of its kind with a market price of ∼4750 USD per bitcoin as of August 31st, 2017 [14]. The enabler and key innovation of Bitcoin is the *blockchain*, a public append-only and tamper-proof log of all transactions ever issued. These properties establish trust in an otherwise trustless, completely distributed environment, enabling a wide range of new applications, up to distributed general-purpose data management systems [69] and purely digital data-sharing markets [41]. In this work, we focus on the arbitrary, non-financial data on Bitcoin's famous blockchain, which primarily stores financial transactions. This non-financial data fuels, e.g., digital notary services [50], secure releases of cryptographic commitments [16], or non-equivocation schemes [62].

However, since all Bitcoin participants maintain a *complete local copy* of the blockchain (e.g., to ensure correctness of blockchain updates and to bootstrap

new users), these desired and vital features put all users at risk when *objectionable content* is irrevocably stored on the blockchain. This risk potential is exemplified by the (mis)use of Bitcoin's blockchain as an anonymous and irrevocable content store [40,56,35]. In this paper, we systematically analyse non-financial content on Bitcoin's blockchain. While most of this content is harmless, there is also content to be considered objectionable in many jurisdictions, e.g., the depiction of nudity of a young woman or hundreds of links to child pornography. As a result, it could become illegal (or even already is today) to possess the blockchain, which is required to participate in Bitcoin. Hence, objectionable content can jeopardize the currently popular multi-billion dollar blockchain systems.

These observations raise the question whether or not unintended content is ultimately beneficial or destructive for blockchain-based systems. To address this question, we provide the first *comprehensive* and *systematic* study of unintended content on Bitcoin's blockchain. We first *survey and explain* methods to store arbitrary, non-financial content on Bitcoin's blockchain and discuss potential benefits as well as threats, most notably w.r.t. content considered illegal in different jurisdictions. Subsequently and in contrast to related work [56,40,12], we *quantify and discuss* unintended blockchain content w.r.t. the wide range of insertion methods. We believe that objectionable blockchain content is a pressuring issue despite potential benefits and hope to stimulate research to mitigate the resulting risks for novel as well as existing systems such as Bitcoin.

This paper is organized as follows. We survey methods to insert arbitrary data into Bitcoin's blockchain in Section 2 and discuss their benefits and risks in Section 3. In Section 4, we systematically analyze non-financial content in Bitcoin's blockchain and assess resulting consequences. We discuss related work in Section 5 and conclude this paper in Section 6.

## 2 Data Insertion Methods for Bitcoin

Beyond intended recording of financial transactions, Bitcoin's blockchain also allows for injection of *non-financial* data, either short messages via special transaction types or even complete files by encoding arbitrary data as standard transactions. We first briefly introduce Bitcoin transactions and subsequently survey methods available to store arbitrary content on the blockchain via transactions.

Bitcoin *transactions* transfer funds between a payer (sender) and a payee (receiver), who are identified by public-private key pairs. Payers announce their transactions to the *Bitcoin network*. The *miners* then publish these transactions in new *blocks* using their computational power in exchange for a *fee*. These fees vary, but averaged at 215 satoshi per Byte during August 2017 [4] (1 satoshi = $10^{-8}$ bitcoin). Each transaction consists of several *input scripts*, which unlock funds of previous transactions, and of several *output scripts*, which specify who receives these funds. To unlock funds, input scripts contain a signature for the previous transaction generated by the owner of the funds. To prevent malicious scripts from causing excessive transaction verification overheads, Bitcoin uses transaction script *templates* and expects peers to discard non-compliant scripts.

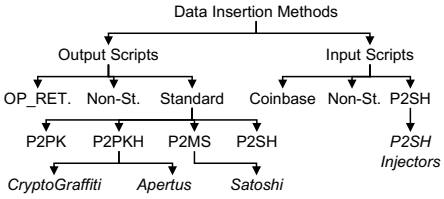


Fig. 1: Bitcoin data insertion methods (italics show content insertion services)

| Method | Payload | Costs/B | Eff. |
|---|---|---|---|
| OP_RET. | 80 B | 3.18–173.55 ct | poor |
| Coinbase | 96 B | — | poor |
| Non-St. Out. | 99 044 B | 1.03–198.05 ct | poor |
| Non-St. In. | | | med. |
| P2PK | 85 345 B | 1.24–207.79 ct | high |
| P2PKH | 58 720 B | 1.87–197.58 ct | high |
| P2MS | 92 625 B | 1.11–234.33 ct | high |
| P2SH Out. | 62 400 B | 1.77–195.54 ct | high |
| P2SH In. | 99 018 B | 1.03–225.61 ct | high |

Table 1: Payload, costs, and efficiency of low-level data insertion methods

Figure 1 shows the insertion methods for non-financial data we identified in Bitcoin. We distinguish *low-level data insertion methods* inserting small data chunks and *content insertion services*, which systematically utilize the low-level methods to insert larger chunks of data. In the following, we refer to non-financial blockchain data as *content* if it has a self-contained structure, e.g., a file or readable text, or as *data* otherwise, e.g., fragments inserted via a low-level method.

### 2.1 Low-level Data Insertion Methods

We first survey the efficiency of the low-level data insertion methods w.r.t. to *insertable payload* and *costs* per transaction (Table 1). To this end, we first explain our comparison methodology, before we detail i) intended data insertion methods (OP_RETURN and coinbase), ii) utilization of non-standard transactions, and iii) manipulation of standard transactions to insert arbitrary data.

**Comparison Methodology.** We measure the *payload per transaction (PpT)*, i.e., the number of non-financial Bytes that can be added to a single standard-sized transaction ($\leq 100\,000$ B). *Costs* are given as the minimum and maximum costs per Byte (CpB) for the longest data chunk a transaction can hold, and for inserting 1 B. Costs are inflicted by paying transaction fees and possibly *burning* currency (at least 546 satoshi per output script), i.e., making it unspendable. For our cost analysis we assume Bitcoin's market price of 4748.25 USD as of August 31st, 2017 [14] and the average fees of 215 satoshi per Byte as of August 2017 [4]. Note that high variation of market price and fees results in frequent changes of presented absolute costs per Byte. Finally, we rate the overall *efficiency* of an approach w.r.t. insertion of arbitrary-length content. Intuitively, a method is efficient if it allows for easy insertion of large payloads at low costs.

**OP_RETURN.** This special transaction template allows attaching one small data chunk to a transaction and thus provides a *controlled channel* to annotate transactions without negative side effects. E.g., in typical implementations peers increase performance by caching spendable transaction outputs and OP_RETURN outputs can safely be excluded from this cache. However, data chunk sizes are limited to 80 B per transaction.

**Coinbase.** In Bitcoin, each block contains exactly one coinbase transaction, which introduces new currency into the system to incentivize miners to dedi-

cate their computational power to maintain the blockchain. The input script of coinbase transactions is up to 100 B long and consists of a variable-length field encoding the new block's position in the blockchain [9]. Stating a larger size than the overall script length allows placing arbitrary data in the resulting gap. This method is inefficient as only active miners can insert only small data chunks.

**Non-standard Transactions.** Transactions can deviate from the approved transaction templates [48] via their output scripts as well as input scripts. In theory, such transactions can carry arbitrarily encoded data chunks. Transactions using non-standard *output* scripts can carry up to 96.72 KiB at comparably low costs. However, they are inefficient as miners ignore them with high probability. Yet, non-standard output scripts occasionally enter the blockchain if miners insufficiently check them (cf. Section 4.2). Contrarily, non-standard *input* scripts are only required to match their respective output script. Hence, input scripts can be altered to carry arbitrary data if their semantics are not changed, e.g., by using dead conditional branches. This makes non-standard input scripts slightly better suited for large-scale content insertion than non-standard output scripts.

**Standard Financial Transactions.** Even *standard financial transactions* can be (mis)used to insert data using mutable values of output scripts. There are four approved templates for standard financial transactions: Pay to public-key (P2PK) and pay to public-key hash (P2PKH) transactions send currency to a dedicated receiver, identified by an address derived from her private key, which is required to spend any funds received [48]. Similarly, multi-signature (P2MS) transactions require $m$ out of $n$ private keys to authorize payments. Pay to script hash (P2SH) transactions refer to a *script* instead of keys to enable complex spending conditions [48], e.g., to replace P2MS [10]. The respective public keys (P2PK, P2MS) and script hash values (P2PKH, P2SH) can be replaced with arbitrary data as Bitcoin peers can not verify their correctness before they are referenced by a subsequent input script. While this method can store large amounts of content, it involves significant costs: In addition to transaction fees, the user must burn bitcoins as she replaces valid receiver identifiers with arbitrary data (i.e., invalid receiver identities), making the output unspendable. Using multiple outputs enables PpTs ranging from 57.34 KiB (P2PKH) to 96.70 KiB (P2SH inputs) at CpBs from 1.03 ct to 1.87 ct. As they behave similarly w.r.t. data insertion, we collectively refer to all standard financial transactions as P2X in the following. P2SH scripts also allow for efficient data insertion into input scripts as P2SH input scripts are published with their redeem script. Due to miners' verification of P2SH transactions, transaction are not discarded if the redeem script is not template-compliant (but the overall P2SH transaction is).

We now survey different services that systematically leverage the discussed data insertion methods to add larger amounts of content to the blockchain.

### 2.2 Content Insertion Services

Content insertion services rely on the low-level data insertion methods to add content, i.e., files such as documents or images, to the blockchain. We identify four conceptually different content insertion services and present their protocols.

**CryptoGraffiti.** This web-based service [30] reads and writes messages and files from and to Bitcoin's blockchain. It adds content via multiple P2PKH output scripts within a single transaction, storing up to 60 KiB of content. To retrieve previously added content, CryptoGraffiti scans for transactions that either consist of at least 90 % printable characters or contain an image file.
**Satoshi Uploader.** The Satoshi Uploader [56] inserts content using a single transaction with multiple P2X outputs. The inserted data is stored together with a length field and a CRC32 checksum to ease decoding of the content.
**P2SH Injectors.** Several services [35] insert content via slightly varying P2SH input scripts. They store chunks of a file in P2SH input scripts. To ensure file integrity, the P2SH redeem scripts contain and verify hash values of each chunk.
**Apertus.** This service [29] allows *fragmenting* content over multiple transactions using an arbitrary number of P2PKH output scripts. Subsequently, these fragments are referenced in an *archive* stored on the blockchain, which is used to retrieve and reassemble the fragments. The chosen encoding optionally allows augmenting content with a comment, file name, or digital signature.

To conclude, Bitcoin offers various options to insert arbitrary, non-financial data. These options range from small-scale data insertion methods exclusive to active miners to services that allow any user to store files of arbitrary length. This wide spectrum of options for data insertion raises the question which benefits and risks arise from storing content on Bitcoin's blockchain.

## 3 Benefits and Risks of Arbitrary Blockchain Content

Bitcoin's design includes several methods to insert arbitrary, non-financial data into its blockchain in both intended and unintended ways. In this section, we discuss potential benefits of engraving arbitrary data into Bitcoin's blockchain as well as risks of (mis)using these channels for content insertion.

### 3.1 Benefits of Arbitrary Blockchain Content

Besides the manipulation of standard financial transactions, Bitcoin offers coinbase and OP_RETURN transactions as explicit channels to irrevocably insert small chunks of non-financial data into its blockchain (cf. Section 2). As we discuss in the following, each insertion method has distinguishing benefits:
**OP_RETURN.** Augmenting transactions with short pieces of arbitrary data is beneficial for a wide area of applications [40,12,62]. Different services use OP_RETURN to link non-financial assets, e.g., vouchers, to Bitcoin's blockchain [40,12], to attest the existence of digital documents at a certain point of time as a digital notary service [58,50,12], to realize distributed digital rights management [70,12], or to create non-equivocation logs [62,8].
**Coinbase.** Coinbase transactions differ from OP_RETURN as only miners, who dedicate significant computational resources to maintain the blockchain, can use them to add extra chunks of data to their newly mined blocks. Beyond advertisements or short text messages [40], coinbase transactions can aid the

mining process. Adding random bytes to the coinbase transactions allows miners to increase entropy when repeatedly testing random nonces to solve the proof-of-work puzzle [48]. Furthermore, adding identifiable *voting flags* to transactions enables miners to vote on proposed features, e.g., the adoption of P2SH [10].
**Large-scale Data Insertion.** Engraving large amounts of data into the blockchain creates a long-term non-manipulable file storage. This enables, e.g., the archiving of historical data or censorship-resistant publication, which helps protecting whistleblowers or critical journalists [66]. However, their content is replicated to all users, who do not have a choice to reject storing it.

Hence, non-financial data on the blockchain enables new applications that leverage Bitcoin's security guarantees. In the following, we discuss threats of forcing honest users to download copies of all blockchain content.

### 3.2 Risks of Arbitrary Blockchain Content

Despite potential benefits of data in the blockchain, insertion of objectionable content can put all participants of the Bitcoin network at risk [43,11,40], as such unwanted content is unchangeable and locally replicated by each peer of the Bitcoin network as benign data. To underpin this threat, we first derive an extensive catalog of content that poses high risks if possessed by individuals and subsequently argue that objectionable blockchain content is able to harm honest users. In the following, we identify five categories of objectionable content:
**Copyright Violations.** With the advent of file-sharing networks, pirated data has become a huge challenge for copyright holders. To tackle this problem, copyright holders predominantly target users that actively distribute pirated data. E.g., German law firms sue users who distribute copyright-protected content via file-sharing networks for fines on behalf of the copyright holders [28]. In recent years, prosecutors also convicted downloaders of pirated data. For instance, France temporarily suspended users' Internet access and subsequently switched to issuing high fines [36]. As users distribute their blockchain copy to new peers, copyright-protected material on the blockchain can thus provoke legal disputes about copyright infringement.
**Malware.** Another threat is to download malware [20,42], which could potentially be spread via blockchains [31]. Malware has serious consequences as it can destroy sensitive documents, make devices inoperable, or cause financial losses [34]. Furthermore, blockchain malware can irritate users as it causes antivirus software to deny access to important blockchain files. E.g., Microsoft's antivirus software detected a non-functional virus signature from 1987 on the blockchain, which had to be fixed manually [68].
**Privacy Violations.** By disclosing sensitive personal data, individuals can harm their own privacy and that of others. This threat peaks when individuals deliberately violate the privacy of others, e.g., by blackmailing victims under the threat of disclosing sensitive data about them on the blockchain. Real-world manifestations of these threats are well-known, e.g., non-consensually releasing private nude photos or videos [54] or fully disclosing an individual's identity to the public with malicious intents [21]. Jurisdictions such as the whole European

Union begin to actively prosecute the unauthorized disclosure *and forwarding* of private information in social networks to counter this novel threat [5].

**Politically Sensitive Content.** Governments have concerns regarding the leakage of classified information such as state secrets or information that otherwise harms national security, e.g., propaganda. Although whistleblowers reveal nuisances such as corruption, they force all blockchain users to keep a copy of leaked material. Depending on the jurisdiction, the intentional disclosure or the mere possession of such content may be illegal. While, e.g., the US government usually tends to prosecute *intentional* theft or disclosure of state secrets [63], in China the mere possession of state secrets can result in longtime prison sentences [49]. Furthermore, China's definition of state secrets is vague [49] and covers, e.g., "activities for safeguarding state security" [60]. Such vague allegations w.r.t. state secrets have been applied to critical news in the past [18,24].

**Illegal and Condemned Content.** Some categories of content are virtually universally condemned and prosecuted. Most notably, possession of *child pornography* is illegal at least in the 112 countries [64] that ratified an optional protocol to the Convention on the Rights of the Child [65]. *Religious content* such as certain symbols, prayers, or sacred texts can be objectionable in extremely religious countries that forbid other religions and under oppressive regimes that forbid religion in general. As an example, possession of items associated with an objected religion, e.g., Bibles in Islamist countries, or blasphemy have proven risky and were sometimes even punished by death [13,38].

In conclusion, a wide range of objectionable content can cause direct harm if possessed by users. In contrast to systems such as social media platforms, file-sharing networks, or online storage systems, such content can be stored on blockchains anonymously and irrevocably. Since all blockchain data is downloaded and persistently stored by users, they are liable for any objectionable content added to the blockchain by others. Consequently, it would be illegal to participate in a blockchain-based systems as soon as it contains illegal content.

While this risk has previously been acknowledged [43], definitive answers require court rulings yet to come. However, considering legal texts we anticipate a high potential for illegal blockchain content to jeopardize blockchain-based system such as Bitcoin in the future. Our belief stems from the fact that, w.r.t. child pornography as an extreme case of illegal content, legal texts from countries such as the USA [47], England [3], Ireland [32] deem all data illegal that can be converted into a visual representation of illegal content. As we stated in Section 2, it is easily possible to locate and reassemble such content on the blockchain. Hence, even though convertibility usually covers creating a visual representation by, e.g., decoding an image file, we expect that the term can be interpreted to include blockchain data in the future. For instance, this is already covered implicitly by German law, as a person is culpable for possession of illegal content if she *knowingly possesses* an *accessible document* holding said content [2]. It is critical here that German law perceives the hard disk holding the blockchain as an document [1] and that users can easily reassemble any illegal content within the blockchain. Furthermore, users can be assumed to *knowingly* maintain control

over such illegal content w.r.t. German law if sufficient media coverage causes the content's existence to become public knowledge among Bitcoin users [61], as has been attempted by Interpol [31]. We thus believe that legislators will speak law w.r.t. non-financial blockchain content and that this has the potential to jeopardize systems such as Bitcoin if they hold illegal content.

## 4 Blockchain Content Landscape

To understand the landscape of non-financial blockchain data and assess its potentials and risks, we thoroughly analyze Bitcoin's blockchain as it is the most widely used blockchain today. Especially, we are interested in i) the *degree of utilization* of data and content insertion methods, ii) the *temporal evolution* of data insertion, and iii) the *types* of content on Bitcoin's blockchain, especially w.r.t. *objectionable* content. In the following, we first outline our measurement methodology before we present an overview and the evolution of non-financial data on Bitcoin's blockchain. Finally, we analyze files stored on the blockchain to derive if any objectionable content is already present on the blockchain.

### 4.1 Methodology

We detect data-holding transactions recorded on Bitcoin's blockchain based on our study of data insertion methods and content insertion services (cf. Section 2). We distinguish detectors for data insertion methods and detectors for content insertion services. To reduce false positives, e.g., due to public-key hash values that resemble text, we exclude all standard transaction outputs that include already-spent funds from analysis. This is sensible as data-holding transactions replace public keys or hashes such that spending requires computing corresponding private keys or pre-images, which is assumed to be infeasible. Contrarily, even though we thoroughly analyzed possible insertion methods, there is still a chance that we do not exhaustively detect all non-financial data. Nevertheless, our content type analysis establishes a solid lower bound as we only consider readable files retrieved from Bitcoin's blockchain. In the following, we explain the key characteristics of the two classes of our blockchain content detectors.

**Low-level Insertion Method Detectors.** The first class of detectors is tailored to match individual transactions that are likely to contain non-financial data (cf. Section 2.1). These detectors detect manipulated financial transactions as well as OP_RETURN, non-standard, and coinbase transactions.

Our text detector scans for P2X output scripts for mutable values containing $\geq 90\,\%$ printable ASCII characters (to avoid false positives). The detector returns the concatenation of all output scripts of the same transaction that contain text.

Finally, we consider all coinbase and OP_RETURN transactions as well as non-standard output scripts. We detect coinbase transactions based on the length field mismatch described in Section 2.1. OP_RETURN scripts are detectable as they always begin with an `OP_RETURN` operation. Non-standard output scripts comprise all output scripts which are not template-conform.

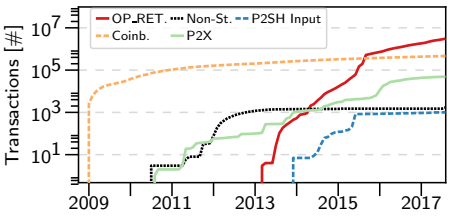


Fig. 2: Cumulative numbers of detected transactions per data insertion method

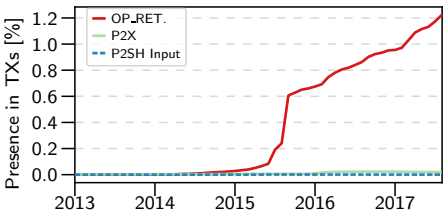


Fig. 3: Ratio of transactions that utilize data insertion methods

**Service Detectors.** We implemented detectors specific to the content insertion services we identified in Section 2.2. These service-specific detectors enable us to *detect and extract* files based on the services' protocols. These detectors also track the data insertion method used in service-created transactions.

The CryptoGraffiti detector matches transactions with an output that sends a tip to a public-key hash controlled by its provider. For such a transaction, we concatenate all mutable values of output scripts that spend fewer than 10 000 satoshi and store them in a file. This threshold is used to ignore non-manipulated output scripts, e.g., the service provider spending their earnings.

To detect a Satoshi Uploader transaction, we concatenate all of its mutable values that spend the same small amount of bitcoins. If we find the first eight bytes to contain a valid combination of length and CRC32 checksum for the transaction's payload, we store the payload as an individual file.

We detect P2SH Injector content based on redeem scripts containing more than one hash operation (standard transactions use at most one). We then extract the concatenation of the second inputs of all redeem scripts (the first one contains a signature) of a transaction as one file.

Finally, the Apertus detector recursively scans the blockchain for Apertus archives, i.e., Apertus-encoded lists of previous transaction identifiers. Once a referred Apertus payload does not constitute another archive, we retrieve its payload file and optional comment by parsing the Apertus protocol.

**Suspicious Transaction Detector.** To account for less wide-spread insertion services, we finally analyze standard transactions that likely carry non-financial data but are not detected otherwise. We only consider transactions with at least 50 *suspicious outputs*, i.e., roughly 1 KiB of content. We consider a set of outputs suspicious if all outputs i) spend the same small amount ($< 10\,000$ satoshi) and ii) are unspent. This detector trades off detection rate against false-positive rate. Due to overlaps with service detectors, we exclude matches of this detector from our quantitative analysis, but discuss individual findings in Section 4.3.

### 4.2 Utilization of Data Insertion Methods

Data and content insertion in Bitcoin has evolved over time, transitioning from single miners exploiting coinbase transactions to sophisticated services that enable the insertion of whole files into the blockchain. We study this evolution in

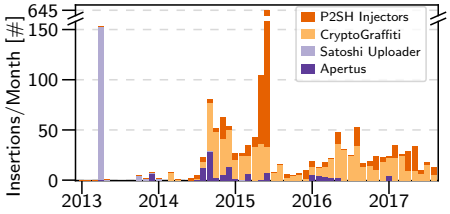


Fig. 4: Number of files inserted via content insertion services per month

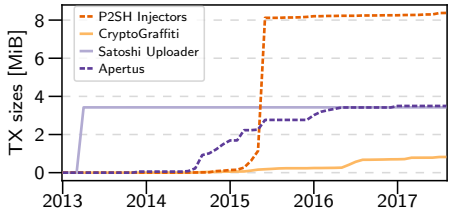


Fig. 5: Cumulative sizes of transactions from content insertion services

terms of used data insertion methods as well as content insertion services and quantify the amount of blockchain data using our developed detectors. Our key insights are that OP_RETURN constitutes a well-accepted success story while content insertion services are currently only infrequently utilized. However, the introduction of OP_RETURN did not shut down other insertion methods, e.g., P2X manipulation, which enable single users to insert objectionable content.

Our measurements are based on Bitcoin's complete blockchain as of August 31st, 2017, containing 482 870 blocks and 250 845 217 transactions with a total disk size of 122.64 GiB. We first analyze the popularity of different data insertion methods and subsequently turn towards the utilization of content insertion services to assess how non-financial data enters the blockchain.

**Data Insertion Methods.** As described in Section 2.1, OP_RETURN and coinbase transactions constitute *intended* data insertion methods, whereas P2X and non-standard P2SH inputs manipulate legitimate transaction templates to contain arbitrary data. Figure 2 shows the cumulative number of transactions containing non-financial data on a logarithmic scale. In total, our detectors found 3 535 855 transactions carrying a total payload of 118.53 MiB, i.e., only 1.4 % of Bitcoin transactions contain non-financial data. However, we strive to further understand the characteristics of non-financial blockchain content as even a single instance of objectionable content can potentially jeopardize the overall system.

The vast majority of extracted transactions are OP_RETURN (86.8 % of all matches) and coinbase (13.13 %) transactions. Combined, they constitute 95.90 MiB (80.91 % of all extracted data). Out of all blocks, 96.15 % have content-holding coinbase transactions. While only 0.26 % of these contain $\geq$ 90 % printable text, 33.49 % of them contain $\geq$ 15 consecutive printable ASCII characters (mostly surrounded by data without obvious structure). Of these short messages, 14.39 % contain voting flags for new features (cf. Section 3.1). Apart from this, miners often advertise themselves or leave short messages, e.g., prayer verses.

OP_RETURN transactions were introduced in 2013 to offer a benign way to augment single transactions with non-financial data. This feature is widely used, as shown by Figure 3. Among all methods, OP_RETURN is the only one to be present with a rising tendency, with currently 1.2 % of all transactions containing OP_RETURN outputs. These transactions predominantly manage off-blockchain assets or originate from notary services [12]. While P2X transactions are contin-

uously being manipulated, they make up only 0.02 % of all transactions; P2SH inputs are virtually irrelevant. Hence, short non-financial data chunks are well-accepted, viable extensions to the Bitcoin system (cf. Section 3.1).

P2X transactions are asymmetric w.r.t. the number and sizes of data-carrying transactions. Although constituting only 1.6 % of all detector hits, they make up 9.08 % of non-financial data (10.76 MiB). This again highlights the high content-insertion efficiency of P2X transactions (cf. Section 2.1).

Finally, we discuss non-standard transactions and non-standard P2SH input scripts. In total, we found 1703 transactions containing non-standard outputs. The three first non-standard transactions (July 2010) repeatedly used the `OP_CHECKSIG` operation. We dedicate this to an attempted DoS attack that targets to cause high verification times. Furthermore, we found 23 P2PKH transactions from October 2011 that contained `OP_0` instead of a hash value. The steady increase of non-standard transactions in 2012 is due to scripts that consist of 32 seemingly random bytes. Contrarily, P2SH input scripts sporadically carry non-standard redeem scripts and are then often used to insert larger data chunks (as they are used by P2SH Injectors). This is due to P2SH scripts not being checked for template conformity. We found 888 such transactions holding 8.37 MiB of data. Although peers should reject such transactions [48], they still often manage to enter the blockchain. Non-standard P2SH scripts even carry a substantial amount of data (7.07 % of the total data originate from P2SH Injectors).

**Content Insertion Services.** We now investigate to which extent content insertion services are used to store content on Bitcoin's blockchain. Figure 4 shows utilization patterns for each service and Figure 5 shows the cumulative size of non-financial data inserted via the respective service. Notably, only few users are likely responsible for the majority of service-inserted content.

In total, content insertion services account for 16.12 MiB of non-financial data. More than a half of this content (8.37 MiB) originates from P2SH Injectors. The remainder was mostly inserted using Apertus (21.70 % of service-inserted data) and Satoshi Uploader (21.24 %). Finally, CryptoGraffiti accounts for 0.82 MiB (5.10 %) of content related to content insertion services. In the following, we study how the individual services have been used over time.

Our key observation is that both CryptoGraffiti and P2SH Injectors are infrequently but steadily used; since 2016 we recognize on average 23.65 data items being added per month using these services. Contrarily, Apertus has been used only 26 times since 2016, while the Satoshi Uploader has not been used at all. In fact, the Satoshi Uploader was effectively used only during a brief period: 92.73 % of all transactions emerged in April 2013. During this time, the service was used to upload four archives, six backup text files, and a PDF file.

Although Apertus and the Satoshi Uploader have been used only infrequently, together they constitute 64.32 % of all P2X data we detected. This stems from the utilization of those services to engrave files into the blockchain, e.g., archives or documents (Satoshi Uploader), or images (Apertus). Similarly, P2SH Injectors are used to backup conversations regarding development of the Bitcoin client, especially online chat logs, forum threads, and emails, with a significant peak

| File Type | Via Service? yes | no | Overall Portion | File Type | Via Service? yes | no | Overall Portion |
|---|---|---|---|---|---|---|---|
| Text | 1353 | 54 | 87.07 % | Archive | 4 | 0 | 0.25 % |
| Images | 144 | 2 | 9.03 % | Audio | 2 | 0 | 0.12 % |
| HTML | 45 | 0 | 2.78 % | PDF | 2 | 0 | 0.12 % |
| Source Code | 7 | 3 | 0.62 % | **Total** | 1557 | 59 | 100.0 % |

Table 2: Distribution of blockchain file types according to our content-insertion-service and suspicious-transactions detectors.

utilization between May and June 2015 (76.46 % of P2SH Injector matches). Especially Apertus is well-suited for this task as files are spread over multiple transactions. Based on the median, the average Apertus file has a size of 17.15 KiB and is spread over 10 transactions, including all overheads. The largest Apertus file is 310.72 KiB large (including overheads), i.e., three times the size of a standard transaction, and is spread over 96 transactions. The most heavily fragmented Apertus file is even spread over 664 transactions. Contrarily, 95.7 % of CryptoGraffiti matches are short text messages with a median length of 80 Byte.

In conclusion, content insertion services are only infrequently used with varying intentions and large portions of content was uploaded in bursts, indicating that only few users are likely responsible for the majority of service-inserted blockchain content. While CryptoGraffiti is mostly used to insert short text messages that also fit into one OP_RETURN transaction, other services are predominantly used to store, e.g., images or documents. As such files can constitute objectionable content, we further investigate them in the following.

### 4.3 Investigating Blockchain Files

After quantifying basic content insertion in Bitcoin, we now focus on readable files that are extractable from the blockchain. We refer to *files* as findings of our content-insertion-service or suspicious-transaction detectors that are viewable using appropriate standard software. We reassemble fragmented files only if this is unambiguously possible, e.g., via an Apertus archive. Out of the 22.63 MiB of blockchain data not originating from coinbase or OP_RETURN transactions, we can extract and analyze 1557 files with meaningful content. In addition to these, we could extract 59 files using our suspicious-transaction detector (92.25 % text). Table 2 summarizes the different file types of the analyzed files. The vast majority are text-based files and images (99.34 %).

In the following, we discuss our findings with respect to objectionable content. We manually evaluated all readable files with respect to the problematic categories we identified in Section 3.2. This analysis reveals that content from all those categories already exists in Bitcoin's blockchain today. For each of these categories, we discuss the most severe examples. To protect the safety and privacy of individuals, we omit personal identifiable information and refrain from providing exact information on the location of critical content in the blockchain.
**Copyright Violations.** We found seven files that publish (intellectual) property and showcase Bitcoin's potential to aid copyright violations. Engraved are the

text of a book, a copy of the original Bitcoin paper [45,56], and two short textual white papers. Furthermore, we found two leaked cryptographic keys: one RSA private key and a firmware secret key. Finally, the blockchain contains a so-called illegal prime, encoding software to break the copy protection of DVDs [56].

**Malware.** We could not find actual malware in Bitcoin's blockchain. However, an individual non-standard transaction contains a non-malicious cross-site scripting detector. A security researcher inserted this small piece of code which, if interpreted by an online blockchain parser, notifies the author about the vulnerability. Such malicious code could become a threat for users as most websites offering an online blockchain parser also offer online Bitcoin accounts.

**Privacy Violations.** Users store memorable private moments on the blockchain. We extracted six wedding-related images and one image showing a group of people, labeled with their online pseudonyms. Furthermore, 609 transactions contain online public chat logs, emails, and forum posts discussing Bitcoin, including topics such as money laundering. Storing *private* chat logs on the blockchain can, e.g., leak single user's private information irrevocably. Moreover, third parties can release information without knowledge nor consent of affected users. Most notably, we found at least two instances of *doxing*, i.e., the complete disclosure of another individual's personal information. This data includes phone numbers, addresses, bank accounts, passwords, and multiple online identities. Recently, jurisdictions such as the European Union began to punish such serious privacy violations, including the distribution of doxing data [5]. Again, carrying out such assaults via blockchains fortifies the problem due to their immutability.

**Politically Sensitive Content.** The blockchain has been used by whistleblowers as a censorship-resistant permanent storage for leaked information. We found backups of the WikiLeaks Cablegate data [37] as well as an online news article concerning pro-democracy demonstrations in Hong Kong in 2014 [25]. As stated in Section 3.2, restrictive governments are known to prosecute the possession of such content. For example, state-critical media coverage has already put individuals in China [18] or Turkey [24] at the risk of prosecution.

**Illegal and Condemned Content.** Bitcoin's blockchain contains at least eight files with sexual content. While five files only show, describe, or link to mildly pornographic content, we consider the remaining three instances objectionable for almost all jurisdictions: Two of them are backups of link lists to child pornography, containing 274 links to websites, 142 of which refer to Tor hidden services. The remaining instance is an image depicting mild nudity of a young woman. In an online forum this image is claimed to show child pornography, albeit this claim cannot be verified (due to ethical concerns we refrain from providing a citation). Notably, two of the explicit images were only detected by our suspicious-transaction detector, i.e., they were not inserted via known services.

While largely harmless, potentially objectionable blockchain content is infrequently inserted, e.g., links to alleged child pornography or privacy violations. We thus believe that future blockchain designs must proactively cope with objectionable content. Peers can, e.g., filter incoming transactions or revert content-holding transactions [11,51], but this must be scalable and transparent.

## 5 Related Work

Previous work related to ours comprises i) mitigating the distribution of objectionable content in file-sharing peer-to-peer networks, ii) studies on Bitcoin's blockchain, iii) reports on Bitcoin's susceptibility for content insertion, and iv) approaches to retrospectively remove blockchain content.

The trade-off between enabling open systems for data distribution and risking that unwanted or even illegal content is being shared is already known from peer-to-peer networks. Peer-to-peer-based file-sharing protocols typically limit the spreading of objectionable *public* content by tracking the reputation of users offering files [6,26,55,73] or assigning a reputation to files themselves [19,67]. This way, users can reject objectionable content or content from untrustworthy sources. Contrarily, distributed content stores usually resort to encrypt *private* files before outsourcing them to other peers [17,7]. By storing only encrypted files, users can plausibly deny possessing any content of others and can thus obliviously store it on their hard disk. Unfortunately, these protection mechanisms are not applicable to blockchains, as content cannot be deleted once it has been added to the blockchain and the utilization of encryption cannot be enforced reliably.

Bitcoin's blockchain was analyzed w.r.t. different aspects by numerous studies. In a first step, multiple research groups [53,33,71,72,39] studied the currency flows in Bitcoin, e.g., to perform wealth analyses. From a different line of research, several approaches focused on user privacy and investigated the identities used in Bitcoin [52,46,44,59,23]. These works analyzed to which extent users can be de-anonymized by clustering identities [52,46,44,59,23] and augmenting these clusters with side-channel information [52,44,59,23]. Finally, the blockchain was analyzed w.r.t. the use cases of OP_RETURN transactions [12]. While this work is very close to ours, we provide a first comprehensive study of the complete landscape of non-financial data on Bitcoin's blockchain.

The seriousness of objectionable content stored on public blockchains has been motivated by multiple works [56,57,43,11,40,51]. These works, however, focus on reporting individual incidents or consist of preliminary analyses of the distribution and general utilization of content insertion. To the best of our knowledge, this paper gives the first comprehensive analysis of this problem space, including a categorization of objectionable content and a survey of potential risks for users if such content enters the blockchain. In contrast to previously considered attacks on Bitcoin's ecosystem [22,27], illegal content can be inserted instantly at comparably low costs and can put all participants at risk.

The utilization of chameleon hash functions [15] to chain blocks recently opened up a potential approach to mitigate unwanted or illegal blockchain content [11]. Here, a single blockchain maintainer or a small group of maintainers can retrospectively revert single transactions, e.g., due to illegal content. To overcome arising trust issues, $\mu$chain [51] leverages the consensus approach of traditional blockchains to vote on alterations of the blockchain history. As these approaches tackle unwanted content for newly designed blockchains, we seek to motivate a discussion on countermeasures also for *existing* systems, e.g., Bitcoin.

## 6 Conclusion

The possibility to store non-financial data on cryptocurrency blockchains is both beneficial and threating for its users. Although controlled channels to insert non-financial data at small rates opens up a field of new applications such as digital notary services, rights management, or non-equivocation systems, objectionable or even illegal content has the potential to jeopardize a whole cryptocurrency. Although court rulings do not yet exist, legislative texts from countries such as Germany, the UK, or the USA suggest that illegal content such as child pornography can make the blockchain illegal to possess for all users.

As we have shown in this paper, a plethora of fundamentally different methods to store non-financial–potentially objectionable–content on the blockchain exists in Bitcoin. As of now, this can affect at least 112 countries in which possessing content such as child pornography is illegal. This especially endangers the multi-billion dollar markets powering cryptocurrencies such as Bitcoin.

To assess this problem's severity, we comprehensively analyzed the *quantity* and *quality* of non-financial blockchain data in Bitcoin today. Our quantitative analysis shows that 1.4 % of the roughly 251 million transactions in Bitcoin's blockchain carry arbitrary data. We could retrieve over 1600 files, with new content infrequently being added. Despite a majority of arguably harmless content, we also identify different categories of objectionable content. The harmful potential of *single instances* of objectionable blockchain content is already showcased by findings such as links to illegal pornography or serious privacy violations.

## Acknowledgements

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under funding reference number 16KIS0443. The responsibility for the content of this publication lies with the authors.

## References

1. German Criminal Code, Section 11 (2013)
2. German Criminal Code, Sections 184b and 184c (2013)
3. Protection of Children Act, Chapter 37, Section 7 (2015)
4. Bitcoin Transaction Fees. `https://bitcoinfees.info` (2016) Accessed 09/23/2017.
5. General Data Protection Regulation, Section 24 (2016)
6. Aberer, K., Despotovic, Z.: Managing Trust in a Peer-2-Peer Information System. In: ACM CIKM. (2001) pp. 310–317
7. Adya, A., Bolosky, W.J., Castro, M., Cermak, G., Chaiken, R., Douceur, J.R., Howell, J., Lorch, J.R., Theimer, M., Wattenhofer, R.P.: FARSITE: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment. SIGOPS Oper. Syst. Rev. **36**(SI) (2002) pp. 1–14
8. Ali, M., Shea, R., Nelson, J., Freedman, M.J.: Blockstack: A New Decentralized Internet. (2017) Accessed 09/23/2017.

9. Andresen, G.: Block v2 (Height in Coinbase). https://github.com/bitcoin/bips/blob/master/bip-0034.mediawiki (2012) Accessed 09/23/2017.
10. Andresen, G.: Pay to Script Hash. https://github.com/bitcoin/bips/blob/master/bip-0016.mediawiki (2012) Accessed 09/23/2017.
11. Ateniese, G., Magri, B., Venturi, D., Andrade, E.: Redactable Blockchain – or – Rewriting History in Bitcoin and Friends. In: IEEE EuroS&P. (2017) pp. 111–126
12. Bartoletti, M., Pompianu, L.: An analysis of Bitcoin OP_RETURN metadata. In: FC Bitcoin Workshop. (2017)
13. Bellinger, J., Hussain, M.: Freedom of Speech: The Great Divide and the Common Ground between the United States and the Rest of the World. Islamic Law and International Human Rights Law: Searching for Common Ground? (2012) pp. 168–180
14. Blockchain.info: Bitcoin Charts. https://blockchain.info/charts (2011) Accessed 09/23/2017.
15. Camenisch, J., Derler, D., Krenn, S., Pöhls, H.C., Samelin, K., Slamanig, D.: Chameleon-Hashes with Ephemeral Trapdoors. In: PKC '17. (2017) pp. 152–182
16. Clark, J., Essex, A.: CommitCoin: Carbon Dating Commitments with Bitcoin. In: FC. (2012) pp. 390–398
17. Clarke, I., Sandberg, O., Wiley, B., Hong, T.W.: Freenet: A Distributed Anonymous Information Storage and Retrieval System. In: Designing Privacy Enhancing Technologies: Workshop on Design Issues in Anonymity and Unobservability. (2001) pp. 46–66
18. Committee to Protect Journalists: Chinese journalist accused of illegally acquiring state secrets. https://cpj.org/x/660d (2015) Accessed 09/23/2017.
19. Damiani, E., di Vimercati, D.C., Paraboschi, S., Samarati, P., Violante, F.: A Reputation-based Approach for Choosing Reliable Resources in Peer-to-peer Networks. In: ACM CCS. (2002) pp. 207–216
20. Dell Security: Annual Threat Report. (2016) Accessed 09/23/2017.
21. Douglas, D.M.: Doxing: a conceptual analysis. Ethics and Information Technology **18**(3) (2016) pp. 199–210
22. Eyal, I., Sirer, E.G.: Majority Is Not Enough: Bitcoin Mining Is Vulnerable. In: FC. (2014) pp. 436–454
23. Fleder, M., Kester, M., Sudeep, P.: Bitcoin Transaction Graph Analysis. (2015)
24. Freedom House: Turkey Freedom of the Press Report. https://freedomhouse.org/report/freedom-press/2016/turkey (2016) Accessed 09/23/2017.
25. Gracie, C.: Hong Kong stages huge National Day democracy protests. http://www.bbc.com/news/world-asia-china-29430229 (2014) Accessed 09/23/2017.
26. Gupta, M., Judge, P., Ammar, M.: A Reputation System for Peer-to-peer Networks. In: ACM NOSSDAV. (2003) pp. 144–152
27. Heilman, E., Kendler, A., Zohar, A., Goldberg, S.: Eclipse Attacks on Bitcoin's Peer-to-Peer Network. In: USENIX Security. (2015) pp. 129–144
28. Herald Union: Copyright infringement by illegal file sharing in Germany. http://www.herald-union.com/copyright-infringement-by-illegal-file-sharing-in-germany (2015) Accessed 09/23/2017.
29. HugPuddle: Apertus – Archive data on your favorite blockchains. http://apertus.io (2013) Accessed 09/23/2017.
30. "Hyena": Cryptograffiti.info. http://cryptograffiti.info Accessed 09/23/2017.
31. Interpol: INTERPOL cyber research identifies malware threat to virtual currencies. https://www.interpol.int/News-and-media/News/2015/N2015-033 (2015) Accessed 09/23/2017.

32. Irish Office of the Attorney General: Child Trafficking and Pornography Act, Section 2. Irish Statue Book (1998) pp. 44–61
33. Kondor, D., Pósfai, M., Csabai, I., Vattay, G.: Do the Rich Get Richer? An Empirical Analysis of the Bitcoin Transaction Network. PLOS ONE **9**(2) (02 2014) pp. 1–10
34. Labs, F.S.: Ransomware: How to Predict, Prevent, Detect & Resond. Threat Response (2016) Accessed 09/23/2017.
35. Le Calvez, A.: Non-standard P2SH scripts. `https://medium.com/@alcio/non-standard-p2sh-scripts-508fa6292df5` (2015) Accessed 09/23/2017.
36. Lee, D.: France ends three-strikes internet piracy ban policy. `http://www.bbc.com/news/technology-23252515` (2013) Accessed 12/12/2017.
37. Lynch, L.: The Leak Heard Round the World? Cablegate in the Evolving Global Mediascape. In Brevini, B., Hintz, A., McCurdy, P., eds.: Beyond WikiLeaks: Implications for the Future of Communications, Journalism and Society. Palgrave Macmillan UK (2013) pp. 56–77
38. Lyons, K., Blight, G.: Where in the world is the worst place to be a Christian? (2015) Accessed 09/23/2017.
39. Maesa, D.D.F., Marino, A., Ricci, L.: Uncovering the Bitcoin Blockchain: An Analysis of the Full Users Graph. In: IEEE DSAA. (2016) pp. 537–546
40. Matzutt, R., Hohlfeld, O., Henze, M., Rawiel, R., Ziegeldorf, J.H., Wehrle, K.: POSTER: I Don't Want That Content! On the Risks of Exploiting Bitcoin's Blockchain as a Content Store. In: ACM CCS. (2016)
41. Matzutt, R., Müllmann, D., Zeissig, E.M., Horst, C., Kasugai, K., Lidynia, S., Wieninger, S., Ziegeldorf, J.H., Gudergan, G., Spiecker gen. Döhmann, I., Wehrle, K., Ziefle, M.: myneData: Towards a Trusted and User-controlled Ecosystem for Sharing Personal Data. In Eibl, M., Gaedke, M., eds.: INFORMATIK, Gesellschaft für Informatik, Bonn (2017) pp. 1073–1084
42. McAfee Labs: Threats Report (December 2016). (2016) Accessed 09/23/2017.
43. McReynolds, E., Lerner, A., Scott, W., Roesner, F., Kohno, T.: Cryptographic currencies from a tech-policy perspective: Policy issues and technical directions. In: Springer LNCS. Volume 8976. (2015) pp. 94–111
44. Meiklejohn, S., Pomarole, M., Jordan, G., Levchenko, K., McCoy, D., Voelker, G.M., Savage, S.: A Fistful of Bitcoins: Characterizing Payments Among Men with No Names. In: IMC. (2013) pp. 127–140
45. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System. (2008) `https://bitcoin.org/bitcoin.pdf`.
46. Ober, M., Katzenbeisser, S., Hamacher, K.: Structure and Anonymity of the Bitcoin Transaction Graph. Future Internet **5**(2) (2013) pp. 237–250
47. Office of the Law Revision Counsel of the United States House of Representatives: U.S. Code, Title 18, Chapter 110, § 2256 (2017)
48. Okupski, K.: Bitcoin Developer Reference. Technical report (2014)
49. Peerenboom, R.P.: Assessing Human Rights in China: Why the Double Standard. (2005) Accessed 09/23/2017.
50. PoEx Co., Ltd: Proof of Existence. `https://proofofexistence.com` (2015) Accessed 09/23/2017.
51. Puddu, I., Dmitrienko, A., Capkun, S.: $\mu$chain: How to forget without hard forks. IACR Cryptology ePrint Archive **2017/106** (2017) Accessed 09/23/2017.
52. Reid, F., Harrigan, M.: An Analysis of Anonymity in the Bitcoin System. In: Security and Privacy in Social Networks. (2013) pp. 197–223
53. Ron, D., Shamir, A.: Quantitative Analysis of the Full Bitcoin Transaction Graph. In: FC. (2013) pp. 6–24

54. Scheller, S.H.: A Picture Is Worth a Thousand Words: The Legal Implications of Revenge Porn. North Carolina Law Review **93**(2) (2015) pp. 551–595
55. Selcuk, A.A., Uzun, E., Pariente, M.R.: A Reputation-based Trust Management System for P2P Networks. In: IEEE CCGrid. (2004) pp. 251–258
56. Shirriff, K.: Hidden surprises in the Bitcoin blockchain and how they are stored: Nelson Mandela, Wikileaks, photos, and Python software. `http://www.righto.com/2014/02/ascii-bernanke-wikileaks-photographs.html` (2014) Accessed 09/23/2017.
57. Sleiman, M.D., Lauf, A.P., Yampolskiy, R.: Bitcoin message: Data insertion on a proof-of-work cryptocurrency system. In: ACM CW. (2015) pp. 332–336
58. Snow, P., Deery, B., Lu, J., Johnston, D., Kirby, P.: Factom: Business Processes Secured by Immutable Audit Trails on the Blockchain. `https://www.factom.com/devs/docs/guide/factom-white-paper-1-0` (2014) Accessed 09/23/2017.
59. Spagnuolo, M., Maggi, F., Zanero, S.: BitIodine: Extracting Intelligence from the Bitcoin Network. In: FC. (2014) pp. 457–468
60. Standing Committee of the National People's Congress: Law of the People's Republic of China on Guarding State Secrets. (1989) Accessed 09/23/2017.
61. Taylor, G.: Concepts of Intention in German Criminal Law. Oxford Journal of Legal Studies **24**(1) (2004) pp. 99–127
62. Tomescu, A., Devadas, S.: Catena: Efficient non-equivocation via bitcoin. In: IEEE S&P. (2017) pp. 393–409
63. Tucker, E.: A Look at Federal Cases on Handling Classified Information. `http://www.military.com/daily-news/2016/01/30/a-look-at-federal-cases-on-handling-classified-information.html` (2016) Accessed 09/23/2017.
64. United Nations: Appendix to the Optional protocols to the Convention on the Rights of the Child on the involvement of children in armed conflict and on the sale of children, child prostitution and child pornography (2000)
65. United Nations: Optional protocols to the Convention on the Rights of the Child on the involvement of children in armed conflict and on the sale of children, child prostitution and child pornography. **2171** (2000) pp. 247–254
66. Waldman, M., Rubin, A.D., Cranor, L.: Publius: A Robust, Tamper-Evident, Censorship-Resistant and Source-Anonymous Web Publishing System. In: USENIX Security. (2000) pp. 59–72
67. Walsh, K., Sirer, E.G.: Experience with an Object Reputation System for Peer-to-peer Filesharing. In: NSDI. (2006)
68. Wei, W.: Ancient 'STONED' Virus Signatures found in Bitcoin Blockchain. `https://thehackernews.com/2014/05/microsoft-security-essential-found.html` (2014) Accessed 09/23/2017.
69. Wood, G.: Ethereum: A Secure Decentralised Generalised Transaction Ledger. Ethereum Project Yellow Paper (2016) Accessed 09/23/2017.
70. Zeilinger, M.: Digital art as 'monetised graphics': Enforcing intellectual property on the blockchain. Philosophy & Technology (2016)
71. Ziegeldorf, J.H., Grossmann, F., Henze, M., Inden, N., Wehrle, K.: CoinParty: Secure Multi-Party Mixing of Bitcoins. In: ACM CODASPY. (2015) pp. 75–86
72. Ziegeldorf, J.H., Matzutt, R., Henze, M., Grossmann, F., Wehrle, K.: Secure and Anonymous Decentralized Bitcoin Mixing. FGCS **80** (3 2018) 448–466
73. Zimmermann, T., Rüth, J., Wirtz, H., Wehrle, K.: Maintaining Integrity and Reputation in Content Offloading. In: IEEE/IFIP WONS. (2016) pp. 1–8