2018 IEEE 32nd International Conference on Advanced Information Networking and Applications

Security of Distributed Ledger Solutions Based on Blockchain Technologies

Marek R. Ogiela, Michał Majcher
AGH University of Science and Technology
Faculty of Electrical Engineering, Automatics,
Computer Science and Biomedical Engineering
30 Mickiewicza Ave., 30-059 Krakow, Poland
e-mail: mogiela@agh.edu.pl, micmajcher@gmail.com

***Abstract*—Distributed Ledger technology and its most notable implementation, the Blockchain, is disrupting today's industry in extremely fast pace with a potential to change the world. The security posture of Blockchain remains one of a key topics in today's industry and distributed services. On and on, we can embrace the attempts to implement the Blockchain technology in sensitive areas of our daily life like finance [1], insurance services [2], health care [3] etc. It is therefore crucial raise awareness of its limitations, possible improvements, as well as embedded compensations. In this paper, we provide a holistic view on the security aspects of the Blockchain technology. We identify the most notable security threats applicable in the above context and reveal technology-specific challenges, that need to be taken into account. Our analysis lists the security features already embedded in the Blockchain and sample uses in nowadays industry. Our results lead to several observations, recommendations, and open points that could be considered in ongoing development of the technology.**

Keywords—distributed ledger; cryptographic protocols; blockchain; security threats

I. Introduction

On the highest possible level, when thinking Distributed Ledger, we are thinking about a purpose-specific distributed network. A network, that shares a locally maintained (through a consensus protocol) database between all its nodes.

As of now, the most notable implementation of the Distributed Ledger is known as Blockchain - a concept firstly mentioned in 1991 by Haber [4] and investigated deeply (as well as pushed to the mainstream) with the famous Bitcoin whitepaper [5]. Essentially, Blockchain is a list (a chain) of cryptographically linked and secured blocks, maintained through a distributed network of node participants. This model creates for us a distributed database, locally stored and independently maintained by anyone who actively participates in the Blockchain network. Some possible structures of distributed networks are presented in Fig. 1.

Theoretically, these blocks could store any kind of data (e.g. in Bitcoin's scenario, it is information about the transactions that occurred in the system) and per design anyone being part of the network can create new blocks (records) in the Blockchain. The blocks are created asynchronously and with no central authority engagement.

In the standard case, considered in this paper, the new block is created after certain amount of valid transaction requests is broadcasted and evaluated within the Blockchain network. Technically we could say that Blockchain transaction is simply a request to change the Ledger state (since Blockchain is pretty much a state transition system, as mentioned in [1]). Such request is then evaluated by the nodes that received it. If the transaction complies with the rules defined in the Blockchain protocol, it is added to the new block being crafted by that node. After required amount of valid transactions is evaluated and added into the new block, the node who managed to create the new block as first, broadcasts his creation to rest of the network.

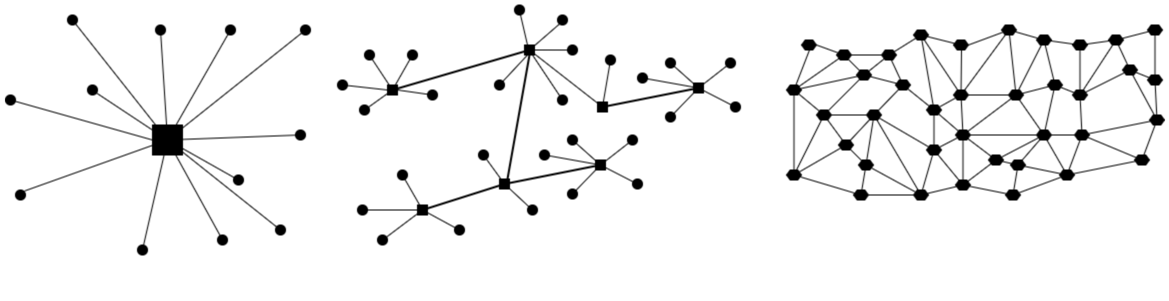

Figure 1. Overview of (a) Centralized, (b) Decentralized, and (c) Distributed networks.

1550-445X/18/$31.00 ©2018 IEEE
DOI 10.1109/AINA.2018.00156

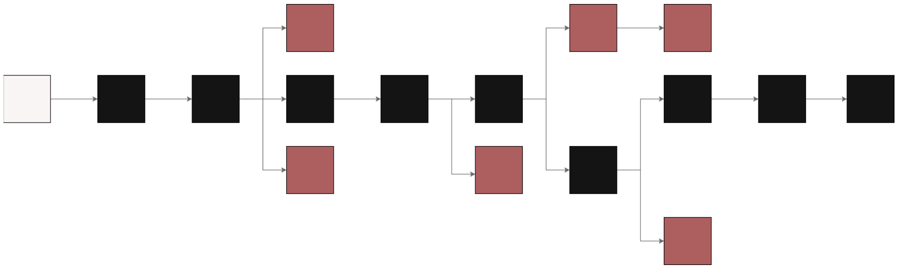

Figure 2. A simple Blockchain diagram, with the white square representing the root block of the Blockchain, the black squares creating the main chain, and the red squares representing the occasional forks.

Naturally, all nodes in the network maintain and create their own instances of Blockchain in an independent way. Moreover, a natural characteristic of any network is large number of concurrent and sometimes also contradictory transactions over it. As a result, in not so rare scenario we can encounter two or more transactions being broadcasted simultaneously in the Blockchain network. By the rule of thumb, the unanimously approved instance of the Blockchain Ledger is mandated for its proper operation. This introduces a conflict scenario, where different next blocks are simultaneously broadcasted in the network and there. Such scenario is known as a fork event, a division of the network into two or more groups each maintaining its own version of the ledger. In order to recover the network and bring back full interaction between all nodes, a consensus must be established. That way, establishing back one commonly approved version of the Blockchain. A simple Blockchain diagram is presenten in Fig. 2.

All of above led to some specific design requirements that could allow for unanimous ratification of the ledger - a distributed consensus mechanism. So far, in Blockchain's systems we can distinguish two main types of the distributed consensus mechanisms that are used:

- Proof-of-Work (PoW), a concept invented by Cynthia Dwork and Moni Naor [6] and based on Client Puzzle Protocol [7]. In short, a PoW protocol requires a client to solve a relatively hard (from computational perspective) task, that on the other hand is easy to verify for the recipient (e.g. service provider). As a result, the client demonstrates that he performed certain computational work and in return receives his "award". This in turn makes specific sets of attacks (such as Denial of Service, Spam, or Double-spending attacks) relatively unprofitable from the financial perspective,
- Proof-of-Stake (PoS), a concept that varies depending on implementation and firstly introduced in PPCoin by Sunny King and Scott Nadal [8]. In PoS systems, instead of proving that certain computational work has been done, the client must demonstrate ownership of certain amount of digital property (which in most cases is money or more generally, crypto-tokens).

Apart from the consensus mechanism, we usually recognize four other components building the Blockchain network. These are:

- *the ledger* itself;
- *the nodes* (i.e. networked computing machines) that built the Blockchain network;
- *the protocol* and its rules defining the behavior that must be followed by the nodes (when to accept the new block, what structure should have the block, how to communicate with other participants of the network, etc.)
- *the cryptography* that links it all together and provides necessary security layer.

All of these build what we know by the name of Blockchain. At this point it is also worth to mention the concept of public and private (or permissionless and permissionned) Blockchains. Anyone can join the network the public Blockchain network, whereas in the private Blockchain is conditional and the participation is subject to some predefined rules created by the community or developers.

From security standpoint, we should be interested mostly in the protocol and cryptography bullets. Both of this define the security of Blockchain and enable reliable processing in the network. The protocol in distributed ledger network simply defines the logic behind nodes behavior. These are the rules defining common structures and message types used and allowed in the given Blockchain network. It also embeds the cryptographic standards accepted by the network. And this cryptography is what is supposed to make the Blockchain provably secure and reliable, as well as maintainable. It safe to state, that any Blockchain implementation combines some set of predefined crypto-primitives for ensuring these three properties. These primitives are usually:

- Hashing functions that are used to calculate hashes of important data sets in the network. Hashing is used on bulks of transactions, as well as whole blocks of the Blockchain. Consequently, we have hashes used as pointers to previous blocks in the Blockchain and hashes representing the current and all past states of transactions in the Ledger.
- Merkle trees [9][10][11], a hash trees where each leaf node is the hash of a data block and each non-leaf node is a hash of the labels of its child nodes. This data structures are used to compress and consolidate the whole Blockchain. Since storing information about all past transactions would be highly ineffective, usually only some part of the Blockchain (consisting only the latest blocks) is maintained by the

nodes with the rest of the history being store in the form of Merkle tree (or even its part and root node).

- Public-key cryptography, that is used by the nodes to digitally sign their transactions (via private key), as well as e.g. point out the receiver of their transaction (via receiver's public key). Essentially, the private/public key pair defines the node in the Blockchain network, it is the way how the node can communicate with other participants, how it can be distinguished from others, and how it can prove it identity (via digital signature).

II. Security Threats for the Blockchain

As any type of technology, Blockchains are susceptible to a variety of different security threats, including all of widely recognized categories: adversarial, accidental, and environmental ones. Yet, unlike in most of technologies, the vast amount of these threats are extremely Blockchain-specific. One could also argue, that even the general and traditional security threats are given specific Blockchain flavor due to the distributed and ungoverned nature of this technology. We use the above *traditional* and *Blockchain-specific* differentiation (used in ENISA's Blockchain report [12]) to describe our threat proposals.

A. The Traditional Threats in the Blockchain Environment

1) Key management and cryptography

As mentioned above, Blockchain highly relates on the Public/Private key cryptography. Therefore, all standard threats related to this area are extremely important in its use case. Protection of node private key is a key for the node owner. In cryptocurrency systems, this private key is used to access and manage the owners assets. In other words, it is used to broadcast the transaction under particular node name, that way assuring its authenticity. By default, Blockchain protocol provides no security control over nodes private keys and this lies fully within the responsibility of the key owner. Therefore, it often happens keys are stolen or simply lost due to improper storage and processing (e.g. storing the private key in clear text file on unprotected yet networked server), making the wallet/node control vulnerable to theft.

Apart from the private key protection, the crypto primitives used are also utmost important. Both the public key generation and hashing algorithms used in the system highly impact its security posture. Depending on the Blockchain implementation the hashing and key generation methods may vary. Multiple ways of key generating and block hashing/signing can be available and accepted. The seed value used can be taken from different sources and the algorithm used for random number generation (RNG) can also differ. If unsafe or easy to crack methods are used in particular Blockchain implementation, the private keys of nodes could be possibly recovered (e.g. via finding two equal signatures in the system due to same random number being generated twice, as described in [13][14]) or the blocks could easy to re-forge and substitute (vide double spending attack described below). It is also important to remember that crypto algorithms and standards currently used to protect the Blockchains (such as SHA-256, Keccak/SHA-3, or ECDSA), that are right now considered secure and impossible to crack, can become flawed and vulnerable over the time. This is especially concerning given the ongoing development of quantum computing machines that highly increase the computing performance and possibilities to crack the existing cryptographic standards (as mentioned in [15]).

Lastly, due to decentralized, distributed, and definite nature of the Blockchain. Golden rule of the Blockchain says that the transaction that has occurred and has been already approved cannot ever be deleted from the Blockchain (whatever happened, happened). Additionally, usually there is no Certification Authority (CA) or any other "golden source of information" present in the Blockchain system (note: this is applicable mostly for the public Blockchains). As a result, the end user has no reliable means to properly verify the credibility of other user's public key. Yet, especially in the cryptocurrency systems, it is these end users who are responsible for assuring the correct public key (which in this cryptocurrency case is usually a wallet address, somehow correlated with the public key) is pointed out. This leaves them open to any attacks that would try to substitute the valid public key/address with a rogue one. For instance we could consider a scenario, where the end-user machine is infected with malware that changes the wallet address typed into the browser window, substituting it with the attacker's address. The victim, unaware of malware presence, may not spot the change and issue the transaction into the network. Due to the definite nature of the Blockchain, the transaction cannot be reversed, even when user spotted he has been deceived.

2) Unsafe & unattested code

Blockchain is fresh and extremely fast developing technology. Most of its code is actually open source and anyone in the world can contribute into development of a number of major projects (Bitcoin, Litecoin, Monero, or Ethereum all have their source code available on GitHub, just to mention few). On the one hand, many skilled engineers could have already reviewed the existing protocols and branches. On the other hand, due to the rapid development and novel, unfamiliar status of the technology, it is still highly possible that a bunch of unknown vulnerabilities exist in particular implementations of the Blockchain or even in the technology foundation itself (much less likely).

3) Distributed Denial of Service (DDoS) attacks

One of the most common and popular cyber-attack scenarios nowadays are denials of service attacks. Attacks where the attacker tries to push the network into an unavailability state. These attacks can be coming from a single source or can be distributed. They can also be carried on various levels, starting from lowest network layers up to the application layer. In Blockchain network this usually boils down into overloading the nodes participating in the network protocol. Usually, these is achieved through sending (a) a huge number of junk packets to the nodes, or (b) a huge number of microscopically small but valid transactions (i.e. relatively cheap transactions). In both cases the nodes would get stuck processing the rogue data and actual transactions in the network would not be impacted with hard delays or even would not be processed at all.

4) *Scalability issues*

Scalability is always and always will be an issue; with any technology. However, for the Blockchain the growth of its ledger and the designed high speed of transaction processing present a huge threat for the scalability of the technology. In nutshell, as long as we are required to store all the data related to our ledger (in distributed way, since that is how the Blockchain is designed), the growth may become unmanageable for worse equipped users (e.g. individuals from less developed technologically countries).For instance, according to charts from bitcoin.com, the size of Bitcoin Blockchain has exceeded 145 GiB and has grown 29 times (2900%, since February 2014 (when the size was 5 GiB). Similarly, according to etherscan.io, the Geth implementation (https://geth.ethereum.org/) of the Ethereum Blockchain has from doubled its size from 20 GiB to 44.14 GiB, between Sep 21 2017 - Jan 16 2018.

5) *Front-end problems*

Finally, we have the rest; anything that is not the Blockchain but interacts with it. We need to remember that Blockchain is just a backend technology, it is a distributed database, and in order to make it usable we need a front-end providing the user interface. As mentioned over and over again in the security space, an IT system is a chain; and this chain is only as secure as its weakest link. Therefore, if the front-end used to interact with the Blockchain is faulty and its design lacks basic security countermeasures, the whole system could be compromised. In fact, so far nearly all documented incidents and successful attacks on the Blockchain technology where related solely to the faulty implementations of the front-end solutions. Be it the infamous Parity multi-sig wallet breach [16] and freeze [17], ridiculous CoinDash Initial Coin Offering (ICO) $7 million hack [18], impactful and notable Tether token hack [19], or the recent hack of Korean exchange Coincheck [20], neither of this incidents were caused by the vulnerabilities present in the Blockchain technology.

B. *Blockchain-specific Threats*

1) *Double-spending attacks*

A huge chunk of threats in the Blockchain system can be grouped under the common name of double-spending attacks. Double-spending is literally a successful spending of some assets more than once. We can distinguish the following attack vectors:

Race attack, which is a basic attempt of an attacker issuing a transaction to the victim (e.g. a merchant), and simultaneously sending a conflicting transaction spending the same amount of the asset himself to the rest of the network. If none countermeasures are taken in the protocol, it is more than likely that the second conflicting transaction will be the one that has been mined into a block as first and generally accepted by network as the right one. There is also *Finney attack* variation of this scenario, where the attacker is additionally cooperating with the miners (which is also the only known successful scenario of double-spending in Bitcoin [21]).

Alternative history attacks, which are the attempts of an attacker to issue a seemingly valid transaction to the merchant, while simultaneously working on its own alternative version of Blockchain fork, where a fraudulent transaction (to himself) is included instead. Unlike the race attack, this attack aims to bypass the standard countermeasure of waiting for certain amount of confirmations (i.e. blocks on top of current block) to be issued in the Blockchain before accepting the transaction. In general, if at the time of transaction confirmation by the victim, the attacker was able to craft more blocks than was required to confirm the transaction he succeeds and loses no assets. Obviously, if he fails, he can still attempt to catch up with the network. Yet, he most likely fails as his overall probability to succeed is equivalent to his mining/minting power (depending on the consensus algorithm) relative to the whole network power and the number of confirmations the merchant waits for. A variation on this attack is called *Sybil attack*. In this scenario an attacker attempts to fill the network with his nodes in such a way, that his victim(s) are connected only to the nodes controlled by himself. That way they will only be broadcasted the fraudulent blocks from his rogue node-net. This could not only allow him to perform double-spending attacks but also e.g. block the victims from the rest of the network (please refer to Denial of Service attack below).

2) *Energy Consumption (in PoW systems)*

One of the most characteristic threats for the PoW Blockchain systems is related to their energy consumption. Extremely high volumes of raw electric power are required to maintain the largest PoW-based systems. This is mostly due to the mining process, that consumes a lot of CPU resources and hence energy. Since so far there are not many really efficient and green energy sources over the world, the problem could become much bigger in the nearest future considering the growth of overall Blockchain space. This could result in high fluctuations of electricity costs, as well as potential environmental impact on the whole planet.

3) *Illegal content storage*

Storing illegal content in the Blockchain can be a problem, as we live in highly regulated and governed world. A lot of data falls down into the scope of certain regulations that e.g. prohibit transferring certain data outside of specific jurisdictions. Some data is simply illegal to store, such as child pornography, pirate videos, or any other bootlegged digital content. By the definition, there's no technical limitation to what kind of data can be stored in the Blockchain; hence, it can be anything. Additionally the distributed nature of this technology makes the network usually geo-distributed across the globe. As a result, the Blockchain could be likely used as convenient tool for storing and processing illegal data. Furthermore, even for normally legitimate data, it is much easier to breach certain regulations and cross-jurisdictional data processing restrictions (extremely significant especially in the financial sector).

4) *Malicious or faulty Smart Contracts*

One of the main features of Blockchain protocols is the possibility to program them and customize the protocol behavior so it fits the purpose. Such programs, that run on the Blockchain network, are usually referred to as Smart Contracts (firstly proposed in [22]). Depending on the Blockchain implementation this functionality can be either very limited and simple (vide scripting in Bitcoin protocol) or very advanced and even Turing-complete (vide Smart Contracts in the Ethereum protocol). Smart contracts in open system can usually be created by anyone, and not necessarily require 4-eye checks or code reviews prior to releasing into the production environment. As a result, any malicious or faulty code can be introduced into the Blockchain environment. The more complex is the Smart Contact, the more it depends on the developer's skills and the more it is prone to human errors. Any bugs and vulnerabilities present in the Smart Contract may result in (a) unintended an faulty behavior of the contract or (b) potential exploitation of the contract by an attacker resulting in loss of data confidentiality, integrity, availability, or all of these combined.

5) *Majority attacks (51% attacks; >50% attacks)*

One of the key components of the Blockchain is its consensus mechanism that allows for establishing a network consensus through majority "voting". Therefore, taking control over the majority of the Blockchain network (in fact, more than 50% is enough) allows an attacker to take full control over the consensus mechanism and in turn the Ledger. This attack is known by the name of the Majority attack. The scenario can occur, when one entity controls or compromises the majority of the network "consensus" power (be it hashing power in the PoW systems, or stake in the PoS systems). A successful Majority attack allows an attacker to freely control the Blockchain, i.e. perform double-spending attacks or block other participants (denial of service).

6) *Privacy issues*

Blockchain became so popular with Bitcoin's cryptocurrency use case. One of the deciding factors for creating the cryptocurrencies was the presumed privacy that can be ensured for all participants in the Blockchain ecosystem. However, the reality verified this cases. Even though, theoretically a person is known to the network only by its public key/address, it is over when this data are correlated to this person's name. By then the privacy of this person's wallet is completely compromised. This is because, in the Blockchain system (especially a public one), anyone can freely download the full ledger and explore the entire history of transactions that happened in the system. In cryptocurrency system, this characteristics can be furthermore used to try and trace the coin's history and that way deduce the identity of the wallet's owner.

7) *Private Blockchains issues*

The main strength of the Blockchain is its decentralized, distributed, and open nature. All threats related to compromising the Consensus Mechanism of the Blockchain are much more straight-forward, much more severe, and much more likely in the private Blockchain case. Unlike in the standard case, where an attacker had to outbalance the whole network of independent nodes (and that way hijack the consensus), he now needs just to compromise and hijack a single entity (the one that owns and operates the Blockchain network). In generalized scenario, where private Blockchain is created by a conglomerate of entities, the attacker has to compromise the majority of entities, which is still much less than in the public network case. Additionally, such compromise must not (and usually will not) be related to any vulnerabilities in the Blockchain protocol itself. Instead, the attacker would gather the data on its victim and look for the weakest link in its security posture, that would in turn be used to infiltrate and take over the network.

III. Security Embeded in the Blockchain

A. *Decentralization is GOOD*

The main power and weakness of the Blockchain lays in its distributed and decentralized nature and its components. Even though network-control takeover and double-spending attacks are the most serious threats for the Blockchains and are only possible due to the distribution and decentralization, they are also the first things addressed in by the components of particular implementations being primarily mitigated with Consensus Mechanisms. In Proof-of-Work systems, the takeover and double-spending is made extremely costly from resource perspective - the bigger the Blockchain network is, the more computing resources you require to compromise the transaction flow (and this equals costs). That way the cost-benefit equitation does not sum up positively for the potential attackers. Similar approach is taken in Proof-of-Stake system, where the attacker would need to obtain the majority of tokens present in the network in order to have the majority of voting power. Additionally in PoS system, the entity that would be able to obtain 51% of the tokens would most likely have more interest to maintain the credibility of the network and in turn the value of the tokens, than in compromising the network.

Decentralization means also that it is all in YOUR hands, when it comes to protecting your data/assets on the Blockchain. It is the end-user and only the end-user who (at least by design) knows and maintains his wallet private key. Unlike in the centralized system (such as classical banking), there is no entity that when compromised could put your valuable assets in danger. You are the only one ultimately responsible for your wallet's security and already there is a number of generally respected (and user-friendly) software/hardware solutions that can be used by the end-users to ensure the private key security, examples are: Trezor and Ledger hardware wallets, or Exodus software wallet.

B. *Cryptography is ROBUST*

The next big thing making the most of Blockchain implementations secure is its robust cryptography. As mentioned, the Blockchain highly relies on the combination of cryptographic primitives used in the system. Worth attention is that the vast majority of Blockchain implementations rely on the widely-used, widely-known, and generally considered as secured (and tested) crypto primitives, such as:

- SHA-256 (*Bitcoin*, *Neo*), Keccak/Sha-3 (*Ethereum*, *Monero*), RIPEMD160 (*Neo*) hashing algorithms.
- ECDSA (*Bitcoin*, *Cardano*, *Ethereum*) and DSA (*Cardano*) asymmetric cryptography algorithms.
- AES (*Monero*) symmetric cryptography algorithm.
- scrypt (*Monero*) key derivation function.

As long as these standard crypto primitives used throughout the tech industry (not only in the Blockchain systems) can be considered secure, we can safely assume that the cryptography of the majority of Blockchain solutions (the ones with publicly revealed and standard crypto methods in use) is unbreakable. This state should safely persist (in most cases) until the quantum computing generally emerges.

C. Privacy can be ENSURED

It is true that there are attacks and possibilities to compromise the privacy of the end-users in the Blockchain systems. However, (1) that does not apply equally to all the implementations and (2) also highly depends on the carefulness of the end-user himself. Unless in standard centralized (e.g. banking) systems, the privacy of the end-user does not really rely on some single entity controlling the system he is using (unless we are considering the private Blockchain). Since the network is distributed and decentralized and it uses only the end-user's public key to interact with him, even the compromise of the whole network would not impact the user's privacy. In fact, everything that happens in the network is publicly accessible anyway - anyone can download the ledger and investigate it. This means, that as long as the end-user is careful enough, it is extremely hard (up to impossible) to correlate his public key with his true identity. This in fact highly relies on the particular implementation of the Blockchain and we can distinguish a subgroup of Privacy Blockchains, that aim to ensure full and unbreakable privacy in their ecosystems. The most notable example in this space is probably the Monero cryptocurrency, where a novel and innovative Ring Signature/Ring Confidential Transactions [23] has been introduced to assure full anonymity in its network.

D. Integrity CANNOT be BROKEN

Consensus Mechanism and protocol rules in Blockchain are extremely powerful. As long as they are properly designed and restrict the malicious behavior, it is literally impossible to compromise the integrity of the generally approved (i.e. accepted by the majority of the network) version of the Ledger. Once the block has been added to the Ledger, it cannot be removed or even shifted in place from where it is. "What happened, happened", that's a golden rule of every Blockchain implementation. The only way, to change the Blockchain content (as described in e.g. double spending attacks) is to create an alternative chain that will for some reason become accepted by the whole network, in place of the old, valid one. Yet, this is effectively dealt with by the implementation of robust Proof-of mechanisms (be it Proof-of-Work, Proof-of-Stake, Proof-of-Space, or any other mechanism). In essence, these mechanisms introduce an artificial processing cost into the network. As a result, any network takeover attempts are simply costly for the attackers and the cost-benefit equation for the attack becomes negative.

IV. Real Life Use-Cases

For closure, let's bring up some real life use cases of Blockchain technology to showcase how its applications can vary.

A. Social Media

One of the most widely used applications nowadays comes from Steemit, a social media/blogging that introduces Smart Media Tokens used to reward publishers on their social media platform: https://steemit.com/. As of Feb 3, the approx. amount of rewards paid in USD on the platform was: $22,728,968.

B. Games and Entertainment

In the end of 2017, a CryptoKitties (www.cryptokitties.co) project based on the Ethereum platform smart contracts made a lot of noise in the Internet. A relatively simple game, that allowed people around the world to collect and breed digital cats, disrupted and literally overloaded the whole Ethereum network, creating network congestion and huge delays in the transactions [24]. Similarly, year 2018 started with CryptoCelebrities (www.cryptocelebrities.co/) boom, an analogous game where the players can collect the "One-of-a-kind Celebrity [Ethereum] Smart Contracts" and consequently trade them on the set up marketplace.

C. Data Reconciliation

In late 2017, the Swiss banking giant UBS announced their Massive Autonomous Distributed Reconciliation (MadRec) platform project based on the Ethereum protocol [25]. Together with other banking entities (such as Barclays, Credit Suisse, or Thomson Reuters), UBS wants to achieve easier and anonymous (that way regulation compliant) reconciliation of static data (legal entity identifiers) about their counterparties using the Blockchain technology. In brief summary, the project will put the anonymized and hashed reference data into the Ethereum Blockchain and reconcile it using the smart contracts. That way identifying any discrepancies and defects.

D. Gambling

Pretty much from the beginning of Bitcoin's existence, gambling sites that accept cryptocurrencies have been created. Yet, this sites not necessarily benefit or utilize the Blockchain technology, rather a cryptocurrency token is simply an acceptable method of payment in there. A notable project in the gambling space is CoinPoker, already live cryptocurrency based online poker lobby built on Ethereum Blockchain. The platform utilizes the Blockchain capabilities to make their solution fully transparent (e.g. via having Random Number Generator publicly available in the Blockchain), private (Blockchain assurance of the token holder anonymity), and accessible (Blockchain transactions are nearly immediate and technology itself is accessible for anyone around the world).

V. Conclusions

The Blockchain is already big, no-one can deny it, and it will likely grow much bigger in the nearest future. The technology provides us with amazing possibilities and enables the use-cases that were literally locked down up to now.

Without a doubt, there is a number of security threats applicable for the Blockchain. These threats must be noticed and remembered. Yet, most of them are already being countermeasured on the protocol design and software development side. For sure a standardized approach for the Blockchain security is needed and baseline framework and standards should be created by recognized organizations. The world is ready to embrace the Blockchain and we must ensure the Blockchain is ready to be securely deployed on the global scale. A caution approach to the technology development should be assumed with a goal to make the continuous improvement of Blockchain's security a standard. In future works it will be made some attempts to also use Blockchain technologies in connection with biologically inspired cryptographic solutions [26][27] as well as secure information management tasks.

ACKNOWLEDGMENT

This work has been supported by the AGH University of Science and Technology research Grant No 11.11.120.329.

REFERENCES

[1] Ethereum.org, “Etherum White Paper,” 2014. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper [Accessed: 02- Feb- 2018].

[2] Insurepal.io, “InsurePal White Paper,” 2017. [Online]. Available: https://insurepal.io/InsurePal_whitepaper.pdf [Accessed: 02- Feb- 2018].

[3] Robomed.io, “Robomed White Paper,” 2017. [Online]. Available: https://robomed.io/download/Robomed_whitepaper_eng_final.pdf [Accessed: 02- Feb- 2018].

[4] S. Haber and W. S. Stornetta, “How to time-stamp a digital document,” Journal of Cryptology, 3(2): 99-111, January 1991.

[5] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” October 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf [Accessed: 02- Feb- 2018].

[6] C. Dwork and M. Naor, “Pricing via Processing, Or, Combatting Junk Mail, Advances in Cryptology,” CRYPTO’92: Lecture Notes in Computer Science No. 740. Springer: 139–147, 1993.

[7] A. Juels and J. Brainard, “Client Puzzles: A Cryptographic Countermeasure Against Connection Depletion Attacks,” In Kent, S. Proceedings of NDSS '99 (Networks and Distributed Security Systems). pp. 151–165, 1999.

[8] S. King, “PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake,” August 2012. [Online]. Avaiable:

[9] R. C. Merkle, “A Digital Signature Based on a Conventional Encryption Function,” Advances in Cryptology — CRYPTO '87. Lecture Notes in Computer Science. 293. p. 369. doi:10.1007/3-540-48184-2_32. ISBN 978-3-540-18796-7, 1988.

[10] R.C. Merkle, “Method of providing digital signatures,” US patent 4309569 assigned to The Board Of Trustees Of The Leland Stanford Junior University, January 1982.

[11] D. Bayer, S. Haber, and W. S. Stornetta, “Improving the Efficiency and Reliability of Digital Time-Stamping,” Sequences. 2: 329–334, March 1992.

[12] ENISA, “Distributed Ledger Technology & Cybersecurity - Improving information security in the financial sector,” January 2017. [Online]. Available: https://www.enisa.europa.eu/publications/blockchain-security/at_download/fullReport [Accessed: 08- Feb- 2018].

[13] N. Schneider, “Recovering Bitcoin private keys using weak signatures from the blockchain,” 2013. [Online]. Available: https://www.nilsschneider.net/2013/01/28/recovering-bitcoin-private-keys.html [Accessed: 02- Feb- 2018].

[14] F. Valsorda, “Exploiting ECDSA failures in the Bitcoin blockchain,” HITN2014KUL, 2014. [Online]. Available: https://conference.hitb.org/hitbsecconf2014kul/materials/D1T1%20-%20Filippo%20Valsorda%20-%20Exploiting%20ECDSA%20Failures%20in%20the%20Bitcoin%20Blockchain.pdf [Accessed: 02- Feb- 2018].

[15] Z. Kirsch and M. Chow, “Quantum Computing: The Risk to Existing Ecnryption Methods,” December 2015. [Online]. Available: http://www.cs.tufts.edu/comp/116/archive/fall2015/zkirsch.pdf [Accessed: 02- Feb- 2018].

[16] ParityTech, “Security Alert,” July 2017. [Online]. Available: https://paritytech.io/security-alert/ [Accessed: 02- Feb- 2018].

[17] ParityTech, “Security Alert,” November 2017. [Online]. Available: https://paritytech.io/security-alert-2/ [Accessed: 02- Feb- 2018].

[18] CoinDash, “CoinDash TGE Hack findings report 15.11.17,” November 2017. [Online]. Available: https://blog.coindash.io/coindash-tge-hack-findings-report-15-11-17-9657465192e1 [Accessed: 02- Feb- 2018].

[19] Tether, “Tether Critical Announcement,” 2017. [Online]. Available: https://tether.to/tether-critical-announcement/ [Accessed: 02- Feb- 2018].

[20] D. Shane, “$530 million cryptocurrency heist may be biggest ever,” January 2018. [Online]. Available: http://money.cnn.com/2018/01/29/technology/coincheck-cryptocurrency-exchange-hack-japan/index.html [Accessed: 02- Feb- 2018].

[21] Bitcointalk.org, “GHash.IO and double-spending against BetCoin Dice,” November 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=327767.0 [Accessed: 08- Feb- 2018]

[22] N. Szabo, “Smart Contracts: Building Blocks for Digital Markets,” 1996. [Online]. Available: "http://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html [Accessed: 02- Feb- 2018].

[23] S. Noether, A. Mackenzie, et al., “Ring Confidential Transactions,” February 2016. [Online]. Available: https://lab.getmonero.org/pubs/MRL-0005.pdf [Accessed: 02- Feb- 2018].

[24] Hacker Noon, “How Crypto-Kitties Disrupted the Ethereum Network,” December 2017. [Online]. Available: https://hackernoon.com/how-crypto-kitties-disrupted-the-ethereum-network-845c22aa1e6e [Accessed: 02- Feb- 2018].

[25] M. del Castillo, “UBS to Launch Live Ethereum Compliance Platform,” December 2017. [Online]. Available: https://www.coindesk.com/ubs-launch-live-ethereum-platform-barclays-credit-suisse/ [Accessed: 02- Feb- 2018].

[26] M. R. Ogiela, and L. Ogiela, “On using Cognitive Models in Cryptography,” IEEE AINA 2016 - The IEEE 30th International Conference on Advanced Information Networking and Applications, Crans-Montana, Switzerland, March 23-25, 2016, pp. 1055-1058, DOI 10.1109/AINA.2016.159.

[27] M. R. Ogiela, and L. Ogiela, “Cognitive Keys in Personalized Cryptography,” IEEE AINA 2017 - The 31st IEEE International Conference on Advanced Information Networking and Applications, Taipei, Taiwan, March 27-29, 2017, pp. 1050-1054, IEEE 2017, DOI 10.1109/AINA.2017.164.