Attack Surface Analysis of Permissioned Blockchain Platforms for Smart Cities

Amanda Davenport
University of Minnesota
daven102@umn.edu

Sachin Shetty
Old Dominion University
sshetty@odu.edu

Xueping Liang
Old Dominion University
x1liang@odu.edu

***Abstract*—In this paper, we explore the attack surfaces in open source permissioned blockchain project Hyperledger Fabric that can be exploited and compromised through cryptographic tactics. Attacks such as insider threats, DNS attacks, private key attacks, and certificate authority (CA) attacks are proposed and discussed. Points in transaction flow where the proposed attacks are threats to the permissioned blockchain are specified and analyzed. Key management systems are discussed, and a deep analysis of Hierarchical Deterministic wallets is conducted. The Membership Service Provider (MSP) proves to be a centralizing aspect of an otherwise decentralized system and proves to be a weakness of the permissioned blockchain network.**

***Index Terms*—Blockchain, Permissioned blockchain, Cryptographic attack, Membership Service Provider**

I. Introduction

Blockchain, while most well known from the cryptocurrency Bitcoin, is quickly becoming a household term. Exploding in the finance sector, the advantages and payoffs of a secure, decentralized, trust-less, and immutable ledger have caught the interest of businesses and groups far beyond the realm of finance and into the realm of smart cities. Due to the sharing economy that accompanies urban developments, blockchain as a peer-to-peer network becomes a viable and natural solution to shared resources [1]. Beyond that, systems such as city wide sensor networks and the Internet of Things (IoT) can implement blockchains in areas such as supply chain or in detection of points of failure and fraud [2] [3]. With an increase in interest, the newest development emerging is that of a permissioned blockchain: a blockchain network with the ability to control administrative permissions such as read, write, and channel access.

With the permissions granted by some certificate authorities pre-established, the identity can be controlled and managed. Instead of the fully decentralized architecture in permissionless blockchains, permissioned blockchain minimizes the possibility of an adversarial pretending to behave normally with anonymous state. This provides the payoffs of a blockchain network with the luxury of exclusivity in peer-to-peer networking. The concept of a permissioned blockchain is also thought to mitigate the threat of many well known blockchain attacks. The 51% attack, sybil attack, and selfish mining are all perceived as lesser threats due to limited access to the network and trust in those granted access to the blockchain. Between the ability to control administrative duties and the perceived lesser threats of some of the most detrimental attacks to a blockchain, the concept of a permissioned blockchain has caught the attention and favor of many. For smart cities, the adoption of the permissioned blockchain helps with the data security and trust establishment. For one thing, the consensus scheme ensures the data integrity so that each peer in the network maintains a consistent view and thus the architecture achieves the data security by preserving the integrity of the system state. For the other, the trust among nodes can be established if there are several pre-established certificate authorities responsible for the identity management. In smart cities, there are still critical infrastructures that require pre-established central authorities to rely on such as the power grid or financial institutes. In such case, permissioned blockchains are needed to build a distributed network with inherent trust and the necessary resilience.

The rise of permissioned blockchain is accompanied by many unexplored security risks and concerns. One of the revolutionary aspects of public blockchain lies in the decentralization of a trust-less peer-to-peer network and the permissioned blockchain is found to infringe on just that. What a network gains in exclusivity it loses in decentralization. The centralizing aspect rests in the Membership Service Provider (MSP) and accompanying Certificate Authority (CA) system. In controlling and dolling out access to the network, the MSP becomes a point of centralization that every actor must go through to interact with the network, such as the application scenarios in healthcare domain [4] and IoT domain [5]. Every sensor in a city wide system is managed by the same administrative persons and while the network itself remains decentralized in a peer to peer fashion, there exists a funnel system in the way read, write, and channel permissions are distributed. Condensed, the security of the sensor network lies completely in the hands of administration through the MSP. As a point of centralization, the MSP becomes a target for attacks. There are also attack surfaces that can be easily targeted at, such as the interface between the sensors in the smart cities and the blockchain systems including the key management sector, the data storage sector and the communication sector. Key management security relies on the cryptographic materials generated for identity establishment and for confidential data handling. The data channel security relies on the proper transmitting and receiving of data objects by the distributed sensors.

Contributions. The contributions of this paper seek to

expand the understanding of the security risks that accompany permissioned blockchains at the cryptographic level. The first contribution offers an attack surface for the MSP and the ways in which access to the blockchain can be compromised are explored. The second contribution proposed is an analysis of where the MSP attack surface effects transaction flow and specific points of weakness. Lastly, we offer an analysis of key storage devices in reference to permissioned blockchains, as private keys are seen to be a defining aspect and potentially detrimental weakness to permissioned blockchain networks. Hierarchical Deterministic wallets are discussed in terms of usability pertaining to permissioned blockchains.

Organization. In section II the MSP construct is reviewed, including an overview of identities, how they work, and how they are implemented in permissioned blockchain as well as an explanation of the CA systems available. These materials are necessary to understand the attack surface of the MSP and how the attacks are possible and applicable. In section III we delve into the attack surface and each specific attack, outlining the threat it raises and how they are possible. In section IV we outline the transaction flow of permissioned blockchain and specify where in the process the attacks labeled in III present weaknesses in transaction flow construct. In section V we present an analysis on key management systems, specifically Hierarchical Deterministic (HD) wallets. Finally, in section VI, we conclude the paper and provide grounds for future works.

The leading permissioned blockchain at the moment is Hyperledger Fabric, an open source project hosted by the Linux Foundation. For the sake of this report and research study, Hyperledger Fabric will be used to model crypt attack strategies and analysis on permissioned blockchains. Note that, though this paper is not meant to overlook or overshadow the many attack surfaces not pertaining to the MSP or CA, we focus attention and scope of this paper on attacks implemented cryptographically and thus pertaining specifically to the centralized aspects of an otherwise decentralized system.

II. The Membership Service Provider

The Membership Service Provider is one of the defining aspects of Hyperledger Fabric and the role it plays is the defining aspect of permissioned blockchains. While there are many different implementations of the MSP, it is recommended as best practice by the Linux Foundation that organizations and MSPs run on a one-to-one basis and each organization utilizes one MSP for their blockchain network [6]. Each MSP is contained in a folder with various subfolders containing the administrator certificate(s), root CA certificates, the node's private key, the node's X.509 certificate, and other optional inclusions. The Linux Foundation doesn't limit or recommend an ideal number of administrators for an organizational wide MSP, however MSPs for all local channels, peers, and orderers are limited to one administrator. Administrative duties encompass providing access and permissions for the entire blockchain network and is thus a single point of centralization [6].

The MSP is enabled by the use of identities wrapped up in X.509 certificates and digital signatures enabled by public key cryptography. Each participant on the network is assigned a digital certificate that assures they are who they say they are and defines the levels of access and permissions they have. These permissions are set by the afore mentioned administrator. Along with a digital certificate, each participate is assigned what Fabric labels a digital signature, or the private key half of a public/private key pair. This is used to sign off on transactions and endorsements to ensure and retain the integrity of the blockchain.

There is no one required CA system for Hyperledger Fabric and organizations are left to decide how they want to set up and run a CA system. Fabric does provide Fabric CA, a CA system by the Linux Foundation and Hyperledger Fabric, however it is completely optional and organizations can instead choose to use any CA of their liking [7]. Another recommendation by the Linux Foundation is to use the Cryptogen tool. The only limitations in options for CAs is whatever implementation is chosen must utilize ECDSA cryptography as RSA cryptography is not currently supported by Hyperledger Fabric. More information on the Fabric CA and the Cryptogen tool can be found from [8] [9]. Hyperledger Fabric implements root and intermediate CAs. In order to mitigate the usage of root CAs as to not be backed up in creation of identities, intermediate CAs can be "validated" by a root CA as a trustworthy and usable CA. The certificate for the intermediate CA is signed by the root CA, and the intermediate CA is then allowed to sign off in the creation of identities. for all intermediate CAs, there exists a trail back to it's root CA.

III. The Attack Surface of the MSP

A. Insider Threat

As recorded in section II, the current design of the organizational wide MSPs has no documented limitations in administrator number and the all local MSPS allow allows for a single administrative certificate, which means that MSP is controlled by a single administrator [6]. In the case of an Insider Threat, the holder of the Administrator Certificate(s) is not to be trusted, and has free reign over the blockchain. Administrative controls such as adding or revoking access, adding identities to the CRL (essentially blacklisting identities), deciding which CA's are accepted by the MSP, and manipulating the amount and type of access a given identity has to the blockchain network are all managed solely by the administrator. In an IoT network of sensors, a sensor itself could be an insider threat. If the sensor is not behaving or acting how it's supposed to false information could be spread to the network. While not a malicious attack, this is still a weakness of the system. Malicious insider attacks could allow for further attacks such as the 51 percent attack or a sybil attack as the administrator could give themselves as many participants and nodes on the blockchain as desired. This is just one example of possible subsequent attacks stemming from an Insider Threat. In smart cities application scenarios, there could be physical limitations that leads to vulnerabilities. In some cases, attackers can easily bypass the crypto mechanisms easily due to weak security configuration and situations where there are not enough resources

for sophisticated security techniques. Insider threats caused by third party supply chains are also raising as a major threat to information infrastructures [22] that function together as different system components. Insider threats related to privacy issues cannot be neglected but is still challenging to detect. These insiders use legitimate rights and privileges to access some sensitive personal information, but for unauthorized purposes, making it difficult to prevent, detect and mitigate [23]. These rogue insiders could exist anywhere in different levels of the architecture, such as network layer, protocol layer or even application layer. Previous work [24] utilizes blockchain to maintain tamper-resistant data provenance for internal auditing but still the real-time goal is challenging.

B. Private Key Leakage

As discussed in section II, certificates and identities are validated and protected in Hyperledger Fabric by asymmetric cryptography. How each participant chooses to store and protect their private key is up to them. There are a wide range of wallets and management methods available as there is no cohesive management scheme required by Hyperledger Fabric. An outside attacker obtaining private key(s) could lead to any number of attacks. By obtaining the administrator's private key, secondary attacks possible include those previously outlined in subsection A, Insider Threat, in which the attacker acts as the administrator with free reign over the blockchain. By instead obtaining multiple private keys for non-administrative participants, secondary attacks such as a sybil or 51 percent attack become viable. As private key leakage attacks provide potential unlimited access to the blockchain and open the possibility for any number of secondary attacks, they are seen as one of the greatest threats to the MSP. Furthermore in the context of smart cities and sensor networks key management schemes are incredibly expensive and unrealistic to implement at this point in time [10]. This then makes permissioned blockchains an incredibly risky protocol as the security regarding the greatest threat to the network is admittedly lacking. The leakage of private keys could further lead to more serious attacks, such as man-in-the-middle attacks, replay attacks, message tampering attacks, and identity leakage attacks [25], making data and privacy at high risks. Once there are system leakages, the adversarial will obtain higher privileges to conduct further intrusions [27]. Man-in-the-Cloud attack [26] is also an outside attack which is caused by the leakage of personal credentials or the manipulation of credential uses, which could affect cloud storage applications.

C. DNS Attack

When a new participant's identity is being created and added to the MSP, there are any number of instances during which a DNS attack could take place [12]. The process of certificate creation to blockchain member has many places attacks such as man-in-the-middle, cache poisoning, DDOS, or many more under the DNS umbrella could be an appropriate and effective form of attack. For example, DNS spoofing attack, also known as DNS cache poisoning attack, is an attack where adversarials corrupt the DNS resolver's cache and force the server to return a false value, making certain network location unavailable [28]. Moreover, the DNS amplification attack is a reflection-based distributed denial of service (DDos) attack. The attacker spoofs look-up requests to domain name system (DNS) servers to hide the source of the exploit and direct the response to the target network. The adversarial could transform the simple DNS query to a larger payload to cause the DDoS attack [29]. Similar to a CA attack, this attack results in certificate tampering and/or stealing such as the permissions and access a certain blockchain member will have. Sensor networks are especially prone to DDOS attacks [11]. Smart cities face the challenge of not only implementing sensor networks with a large weakness for DDOS attacks but an accompanying blockchain system with a weakness there as well.

D. CA Attack

Digital certificates and identities are crucial to the operation of the MSP. As described in section II, Hyperledger Fabric allows the user to choose how to run a certificate authority and produce cryptographic materials. Options include the Fabric-CA, a process built by the contributors to Hyperledger Fabric, Cryptogen, and your own/a third party CA. Implementations of these CA's themselves have their own flaws. Cryptogen produces all private keys in one centralized location and it is up then to the user to adequately and safely copy them to appropriate hosts and containers [13]. This lends itself towards subsection B, private key leakage attack by providing all private keys in one place.

Outside of implementation weaknesses, the entirety of the MSP and therefore membership to the blockchain runs on CAs and the ability to trust that certificates are valid, and owners of certificates are who they say they are. Attacks on well known third party CA's have been known to occur as well [14] [15] [16], and if executed successfully, could be detrimental to the security of the MSP resulting in falsified identities.

A further weakness of CAs in Hyperledger Fabric is in the way they're implemented in the MSP. The MSP requires at least one root CA and can support as root and intermediate CA's as desired. If the root CA certificate or implementation were to be attacked, all certificates leading back to the root certificate are compromised.

IV. Transaction Flow of Hyperledger Fabric

Transaction flow presents many points of attack in a permissioned blockchain. From sensor activation and information gathering to being part of the blockchain, Figure 1 presents a breakdown of how transaction flow executes in Hyperledger Fabric, as documented by [17]. Each step is explained as follows.

- Box 1: The requester sends a proposal to themselves and the other transacting party to be signed and a request to participating peers to be endorsed.
- Box 2: Endorsing peers verify that the signatures from the requester and transactor are valid and the requester is authorized to preform the proposed operation.

- Box 3: The proposal is executed, and a set of read and write values is produced along with a response value and is then sent to the SDK.
- Box 4: The SDK verifies endorsing peer signatures and determines if proposal responses match.
- Box 5: The SDK submits validated responses to the ordering service.
- Box 6: The ordering service orders transactions chronologically and sends a block of ordered transactions to each peer.
- Box 7: Blocks are validated by peers to ensure no changes to the ledger have been made since the read/write sets and proposal response were produced and tagged as valid if requirements are met or invalid if not. Validation does not involve validating signatures at this time.
- Box 8: If tagged valid, all peers append the block to the chain.
- Box 9: An event is emitted notifying the client that their transaction has been appended to the blockchain.

Transaction flow as documented in [17] has been condensed to focus on what is important for the purpose of this project. For a more complete and in-depth understanding see [17]. Blocks highlighted in red indicate vulnerabilities due to attacks outlined in section III and are as follows:

- Box 1: Vulnerable to insider threat or private key leakage. The act of submitting a new transaction that didn't happen or shouldn't be submitted can be carried out by compromised or untrustworthy actors on the network due to insider threats and/or private key leakage.
- Box 2: Vulnerable to CA attacks. The signatures used in Hyperledger Fabric are unique and depend on the identities of those signing, which in turn depend on a valid digital certificate. A compromised identity due to any type of certificate authority manipulation (a CA attack) can result in falsified signatures and a compromised blockchain.
- Box 4: Vulnerable to CA attacks. Validating endorsing peer signatures and vulnerable to a CA attack due to same reasons as Box 2. See bullet above for analysis.
- Box 7: Vulnerable to insider threat and private key leakage attacks. Untrustworthy actors on the network due to afore mentioned threats can mark blocks as valid or invalid incorrectly to suit their purposes.
- Box 8: Vulnerable to insider threat and private key leakage attacks. Untrustworthy actors on the network due to afore mentioned threats can fail to append blocks on to the blockchain.

For many of the threats outlined above a single point of failure can be recovered by the blockchain being a peer-to-peer network by nature. A single sensor sending false information won't bring down the entire network. However, as the number of compromised actors goes up the success rate of outlined attacks rises. It's important to note here that the DNS attack does not present itself in transaction flow. The DNS attack presents a security threat in a specific point in identity creation that takes place outside of transaction flow. While it is seemingly low risk due to the specific nature of the attack, that is not to mitigate the threat it presents and should not be overlooked.

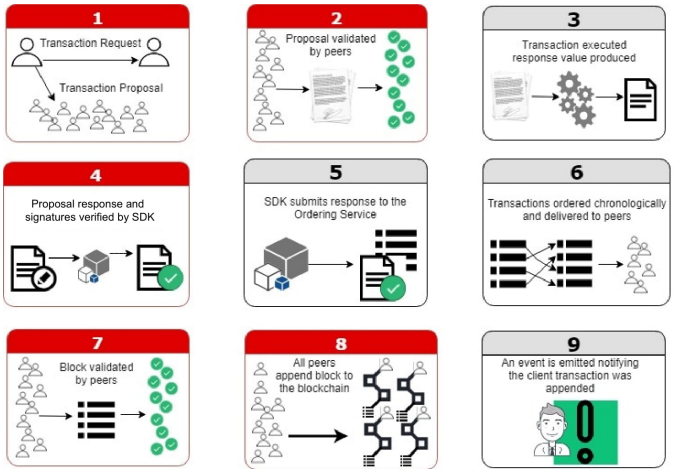

Fig. 1. Transaction Flow of Hyperledger Fabric

V. Key Management and HD Wallet Analysis

The use of asymmetric cryptography in permissioned blockchain is what allows for the permissioned aspect of the network. Storing private keys confidently and securely becomes vital, and finding a key management system that fits the needs and nuances of permissioned blockchains is crucial to the security and integrity of the blockchain network. Due to time and resource constraints, for the sake of this research paper we condense our time and focus on one management system, the Hierarchical Deterministic (HD) wallet. We preform a deep analysis in the topics of creation, issuing, storing, revocation, and re-revocation.

A. Creation

HD wallets have a unique creation process. Rooted at the Master Seed all private keys available for use, of which there are theoretically infinite, are predetermined. By the use of hash functions the Master Seed produces the Master Private Key (MPrK), which is then used to produce all child private keys. The process for how the Master Seed is created and how child keys are derived can be found at [18] [19].

B. Issuing

Issuance of child keys occurs in chronological order of creation. As each subsequent key depends on the hash value of the previous key, in order for the hierarchical component of the HD wallet to exist keys must be issued in the chronological order of creation.

C. Storing

The Master Seed's unique ability to recreate all subsequent children keys allows for easy transportation and storage. All that needs to be kept is the Master Seed, as securely storing

all child private keys individually isn't necessary for the wallet to function. Simply storing or remembering the mnemonic for the Master Seed and in turn the MPrK is enough to store and transport every single key created by the wallet.

D. Revocation and Re-revocation

While HD wallets have understandable allure, there are drawbacks and usability issues to consider. As it is an asymmetric system, for both the MPrK and any child keys there exist corresponding public keys. By obtaining both the public half of the MPrK, the Master Public Key (MPuK), and any child private key, full regeneration of the HD wallet back to the Master Seed is possible and the entire wallet is compromised [20]. Secondly, the mnemonics generated are not user friendly or easy to remember. Made up of arbitrary and random words, many often resort to writing down or storing their mnemonic elsewhere on their computer which then acts as a key itself and defeats the purpose of an HD wallet [21].

In thinking about the usability of HD wallets in a permissioned blockchain setting many questions and concerns arise. If the Master Seed is kept by the administration and a child private key is dolled out to each new member, were the wallet system to be compromised the entire blockchain network would be unreliable. Issuing each member their own HD wallet is unnecessary as having more than one private key is not an integral part of the network. In the case of sensor networks, providing each sensor with an entire HD wallet would actually mitigate the purpose of the wallet in the first place as individual sensors have no capacity to "remember" a mnemonic. The Master Seed generating phrase would have to be stored on the sensor in which case this becomes no different than storing a private key, which was what HD wallets try to circumvent.

VI. Conclusion and Future Work

It's important to note that permissioned blockchain systems and Hyperledger Fabric specifically are still in stages of development. That being said, Fabric as of recent did release a 1.0 version of their source code, and as such the security and integrity of the Membership Service Provider as the centralized aspect of the network needs to be put to the test. As stated previously, there are advantages and disadvantages to such a permissioned blockchain network. This paper seeks to highlight the disadvantages in hopes to shed light on the real risks that accompany permissioned blockchain networks and in this case specifically Hyperledger Fabric.

A recent study adopts Intel SGX technology [30] to secure the Membership Service of Hyperledger Fabric [31]. With SGX remote attestation and isolated execution features, each distributed node can be enrolled as a trusted entity. Security properties for membership service in distributed ledger and illustration of how SGX capabilities help to achieve these properties in each phase of membership service are presented, including member registration, enrollment, transaction signing and verifying and transacting auditing. The SGX enabled membership service could enhance the support of privacy preservation, and defense capabilities against adversarial attacks. The accountability could also be achieved by adopting this hardware assisted scheme [32]. However, considering the hardware cost involved, this solution is not a general proposal for all membership service scenarios.

Many revolutionary aspects of the blockchain concept are lost in permissioned blockchain. White still an immutable ledger, the network is no longer trust-less as absolute trust must be put into the MSP administrator. While the network itself is peer-to-peer and decentralized, the permissioned blockchain as a whole can not be categorized as decentralized as the MSP in itself is a centralizing unit. Ultimately, permissioned blockchains provide exclusivity at the expense of centralization in key components. Furthermore, the MSP has vulnerabilities that can be found in any centralized database or network. The problems that currently accompany centralized databases are not solved with permissioned blockchains.

The idea of a permissioned blockchain presents many hopeful solutions to development and integration of smart cities. In practice however many more efforts need to be conducted to security of permissioned blockchains before they can be a realistic implementation. In the future we move on to a more in-depth approach to permissioned blockchain security, whether an analytical or experimental analysis. The attack surface of the MSP needs to be analyzed comparatively and rigorous proofs need to be built to numerically quantify the threats that accompany permissioned blockchain. Further work is encouraged in comparative analysis of key management systems to find a best fit system for permissioned blockchain.

VII. Acknowledgements

The research of Amanda Davenport is supported in part by NSF under grant CNS-1659795. This work is also supported by Air Force Material Command award FA8750-16-0301 and Office of the Assistant Secretary of Defense for Research and Engineering agreement FA8750-15-2-0120

References

[1] J. Sun, J. Yan, K. Zhang. "Blockchain-based sharing services: What blockhchain technology can contribute to smart cities" in *Financial Inovation*, Vol 2, pp 1-9, Dec. 2016.

[2] C. O'Connor. "What blockchain means for you, and the Internet of Things". *IBM Internet of Things BLog* (2017). http://www.pwc.com/us/en/technology-forecast/blockchain/digital-business.html

[3] "Continuous interconnected supply chain". *Deloitte*. (2017). (Online Documentation). https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/technology/lu-blockchain-internet-things-supply-chain-traceability.pdf

[4] Liang X, Zhao J, Shetty S, Liu J, Li D. Integrating blockchain for data sharing and collaboration in mobile healthcare applications. InPersonal, Indoor, and Mobile Radio Communications (PIMRC), 2017 IEEE 28th Annual International Symposium on 2017 Oct 8 (pp. 1-5). IEEE.

[5] Liang X, Zhao J, Shetty S, Li D. Towards data assurance and resilience in iot using blockchain. InMilitary Communications Conference (MILCOM), MILCOM 2017-2017 IEEE 2017 Oct 23 (pp. 261-266). IEEE.

[6] Membership Service Providers (MSP). *Hyperledger-Fabricdocs Master Documentation, The Linux Foundation*. (2017). (Online Documentation). https://hyperledger-fabric.readthedocs.io/en/release-1.2/msp.html#

[7] "Identity." *Hyperledger-Fabricdocs Master Documentation, The Linux Foundation*. (2017). (Online Documentation). https://hyperledger-fabric.readthedocs.io/en/release-1.2/identity/identity.html

[8] Fabric CA User's Guide. *Hyperledger-Fabric-Cadocs Master Documentation, The Linux Foundation*. (2017). (Online Documentation). https://hyperledger-fabric-ca.readthedocs.io/en/latest/users-guide.html.

[9] Cryptogen. *Hyperledger-Fabricdocs Master Documentation, The Linux Foundation*. (2017). (Online Documentation). https://hyperledger-fabric.readthedocs.io/en/master/commands/cryptogen.html.

[10] R. Roman, C. Alcaraz, J. Lopez, and N. Sklavos, "Key Management Systems for Sensor Networks in the Context of the Internet of Things". (2011). Computers & Electrical Engineering. 37. 147-159. 10.1016/j.compeleceng.2011.01.009.

[11] T. Kaur, K. K. Saluja and A. K. Sharma, "DDOS attack in WSN: A survey," 2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE), Jaipur, 2016, pp. 1-5. doi: 10.1109/ICRAIE.2016.7939566

[12] M. Saad, J. Spaulding, A. Mohaisen, C. Kamhoua, L. Nijila, and D. H. Nyang, Exploring the Attack Surface of Blockchain, rep.

[13] odu-vmasc. Odu-Vmasc/Blockchain. GitHub, github.com/odu-vmasc/Blockchain/blob/master/fabric/examples/fabric-samples/fabric-ca/README.md.

[14] Trustico Revokes 23,000 SSL Certificates Due to Compromise. Cyberscoop, 1 Mar. 2018, www.cyberscoop.com/trustico-digicert-ssl-certificates-revoked/.

[15] IEEE Xplore Full-Text PDF: Design and Implementation of Autonomous Vehicle Valet Parking System - IEEE Conference Publication, Wiley-IEEE Press, Dec. 2011, ieeexplore.ieee.org/stamp/stamp.jsp?tp=& arnumber=6096548.

[16] D. Fisher. (2012). "Final Report on DigiNotar Hack Shows Total Compromise of CA Servers." *Threatpost*. (Online article). https://threatpost.com/final-report-diginotar-hack-shows-total-compromise-ca-servers-103112/77170/

[17] "Transaction Flow" *Hyperledger-Fabricdocs Transaction Flow, The Linux Foundation*. (2017). (Online Documentation). http://hyperledger-fabric.readthedocs.io/en/release-1.0/txflow.html

[18] "The Master Seed" *The Master Seed - Ledger Documentation Hub 2 documentation*. (2016). (Online Documentation). http://ledger.readthedocs.io/en/latest/background/master_seed.html

[19] "HD Key Generation" *HD Key Generation - Ledger Documentation Hub 2 documentation*. (2016). (Online Documentation). http://ledger.readthedocs.io/en/latest/background/hd_keys.html

[20] P. Wuille "BIP32: Hierarchical Deterministic Wallets," 2012. https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki

[21] V. Buterin "Deterministic Wallets, Their Advantages and their Understated Flaws", *Bitcoin Magazine*, 2013. https://bitcoinmagazine.com/articles/deterministic-wallets-advantages-flaw-1385450276/

[22] P. Wang, A. Ali and W. Kelly, "Data security and threat modeling for smart city infrastructure," 2015 International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC), Shanghai, 2015, pp. 1-6.

[23] Cavoukian A, Polonetsky J, Wolf C. Smartprivacy for the smart grid: embedding privacy into the design of electricity conservation. Identity in the Information Society. 2010 Aug 1;3(2):275-94.

[24] Liang X, Shetty S, Tosh D, Kamhoua C, Kwiat K, Njilla L. Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. InProceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing 2017 May 14 (pp. 468-477). IEEE Press.

[25] Khan Z, Pervez Z, Ghafoor A. Towards cloud based smart cities data security and privacy management. InUtility and cloud computing (UCC), 2014 IEEE/ACM 7th international conference on 2014 Dec 8 (pp. 806-811). IEEE.

[26] Liang X, Shetty S, Zhang L, Kamhoua C, Kwiat K. Man in the cloud (mitc) defender: Sgx-based user credential protection for synchronization applications in cloud computing platform. InCloud Computing (CLOUD), 2017 IEEE 10th International Conference on 2017 Jun 25 (pp. 302-309). IEEE.

[27] Yan G, Wen D, Olariu S, Weigle MC. Security challenges in vehicular cloud computing. IEEE Transactions on Intelligent Transportation Systems. 2013 Mar 1;14(1):284-94.

[28] Khatoun R, Zeadally S. Cybersecurity and privacy solutions in smart cities. IEEE Communications Magazine. 2017 Mar;55(3):51-9.

[29] https://whatis.techtarget.com/definition/DNS-amplification-attack

[30] Costan V, Devadas S. Intel SGX Explained. IACR Cryptology ePrint Archive. 2016 Feb;2016(086):1-18.

[31] Liang X, Shetty S, Tosh D, Foytik P, Zhang L. Towards a Trusted and Privacy Preserving Membership Service in Distributed Ledger Using Intel Software Guard Extensions. InInternational Conference on Information and Communications Security 2017 Dec 6 (pp. 304-310). Springer, Cham.

[32] Liang X, Shetty S, Zhao J, Bowden D, Li D, Liu J. Towards Decentralized Accountability and Self-sovereignty in Healthcare Systems. InInternational Conference on Information and Communications Security 2017 Dec 6 (pp. 387-398). Springer, Cham.