2023 Fourth International Conference on Intelligent Data Science Technologies and Applications (IDSTA) | 979-8-3503-3925-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/IDSTA58916.2023.10317833

SECURING DOMAIN NAME SYSTEMS WITH BLOCKCHAIN

Joshua Theoder
College of Engineering, Al Ain University
Abu Dhabi, United Arab Emirates
joshuatheoder@outlook.com

Binusha Shabu Metharath
College of Engineering, Al Ain University
Abu Dhabi, United Arab Emirates
binushashabu@gmail.com

Sahel Alouneh
Al Ain University,
College of Engineering
Computer Engineering Department,
German Jordanian University
sahel.alouneh@aau.ac.ae
sahel.alouneh@gju.edu.jo

***Abstract*—This paper discusses the limitations of the hierarchical and centralized Domain Name System (*DNS*) and the challenges it poses to net-neutrality and privacy. Despite the use of digital signatures based on public key cryptography, *DNS* attacks continue to affect data integrity, confidentiality, and service availability. The paper argues for the exploration of emerging technologies, such as blockchain, to promote data encryption and authentication through a decentralized and non-hierarchical approach. The project aims to research and develop novel *DNS* name-resolution techniques using blockchain technology and to explore their potential benefits and limitations.**

Keywords—blockchain, DNS, smart contracts, integrity, security, privacy, decentralization

I. INTRODUCTION

The Domain Name Service (*DNS*) plays a vital role in navigating the internet, allowing computers to locate resources through domain names. However, the traditional *DNS* system operates in a hierarchical and centralized manner, with a root level, registries, and registrars managed by the Internet Corporation for Assigned Names and Numbers (ICANN), which can result in single points offailure and raise concerns about net-neutrality, privacy, and security [7]. *DNS* attacks can compromise data integrity, confidentiality, and service availability, leading to reputationdamage and financial losses. To address these limitations, there is a need to explore emerging technologies that offer a decentralized and non-hierarchical approach to *DNS* [19], with advanced cryptographic functions for improved data integrity and trust in digital transactions and identities. This research project aims to investigate and develop novel *DNS* name-resolution techniques using Blockchain [10], a promising technology known for its distributed and secure nature [5]. The objective is to assess the practicality and performance of using blockchain for *DNS* in comparison to conventional *DNS* methods.

II. ADVANTAGES OF USING BLOCKCHAIN TECHNOLOGY

Integrating blockchain into *DNS* can offer numerous benefits. Firstly, blockchain's decentralized and distributed nature can enhance the security and reliability of *DNS* [4]. Traditional *DNS* systems are vulnerable to single points of failure, while blockchain-based *DNS* can mitigate this risk by eliminating the need for a central authority [8].

Additionally, the use of blockchain's public key cryptography can provide robust data encryption and authentication, ensuring the integrity and confidentiality of *DNS* data. Moreover, blockchain's transparency and auditability features can enhance trust and accountability in *DNS* operations, as all changes to the *DNS* records are recorded on the blockchain and can be verified by network participants [3]. Furthermore, the integration of blockchain can foster innovation by enabling the development of novel *DNS* name-resolution [9]techniques and facilitating interoperability with other blockchain-based applications. Overall, integrating blockchain into *DNS* can offer improved security, reliability, transparency, and innovation, [20] making it a compelling idea with significant potential for enhancing the *DNS* ecosystem.

III. RELATED WORK

Although there are a few existing solutions for linking a domain name to a blockchain address, they usually only support a single type of blockchain address or, at best, a limited number of pre-selected ones [13]. These blockchain resolution ecosystems also support redirecting traditional TLDs.

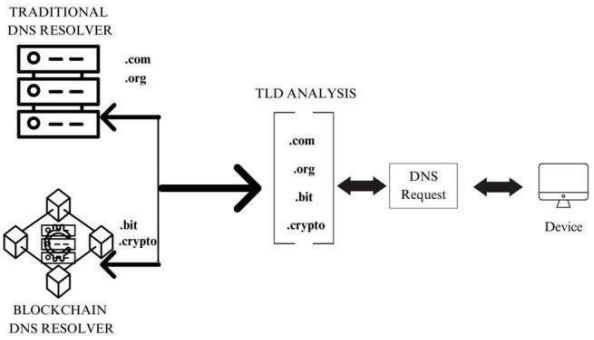

Fig. 1. Workflow of respective domain system resolvers to analyze and direct the query.

One of the existing solutions being, *Unstoppable Domains*, an online-service that connects *Web2* to *Web3* through the use of crypto-TLDs i.e. blockchain domains. By utilizing an Unstoppable Domain, you can create a web address on the blockchain, akin to a URL, such as *Samplename.crypto* or *Samplename.zil*. This human-readable name serves as a simplified way to connect with decentralized apps (*dApps*) and exchanges, translating complex cryptocurrency wallet addresses into an easily memorable format. Unstoppable Domains provides a unique approach to web domain ownership, offering unparalleled control over the domains

you create [18]. Unlike traditional domain providers, when you purchase a crypto domain through Unstoppable Domains, you become the permanent owner instead of just renting it. You have the freedom to transfer, update, and link your domain to other services without relying on Unstoppable Domains for assistance. Additionally, Unstoppable Domains empowers you to create decentralized websites that can be uploaded to the InterPlanetary File System (*IPFS*), a distributed and peer-to-peer file storing andsharing system.

Another mainstream approach being, the *Ethereum Name Service (ENS*) [14]. A solution for the *Ethereum* blockchain that enables the creation and management of domain names. *ENS* consists of two main components: registrars and resolvers. Registrars are smart contracts that own top-level domains (*TLDs*) such as *'.eth'* or *'.test'* and maintain records of all domains and subdomains on the *Ethereum* blockchain. Resolvers, on the other hand, function similar to traditional *DNS* resolvers, providing domain name resolution services. Registrars specify the rules for allocating subdomains, allowing users to obtain ownership of a domain by adhering to these rules [6]. This enables domain owners to configure subdomains for themselves or others as needed. However, it's important to note that *Ethereum* smart contracts come with associated costs, so users are required to pay a fee in *ether* (*Ethereum's* cryptocurrency) when registering or resolving a name using *ENS*.

TABLE I. CHARACTERISTICS OF COMMON *DNS* SYSTEMS

SYSTEM	CATEGORY		
	PLATFORM	***REGISTRAR AND RESOLUTION***	***TLDs***
ICANN	Network of Servers and Resolvers	Centralised	.com, .org,
OpenNIC	Decentralised Servers	Hybrid	.bbs, .pirate
ENS	*Ethereum*	Decentralised	.eth, .onion
Unstoppable Domains	Blockchain Wallet	Decentralised	.crypto, .blo

IV. *DNS* ON BLOCKCHAIN (POC)

This section presents our solution.

The solution consists of three components: an *Ethereum* smart contract, a *DNS* server, and a simple command-line interface (CLI) for adding/changing *DNS* records. Additionally, the *DNS* server needs to connect to an *Ethereum* node. While *Ethereum* may not necessarily be the optimal choice, it was selected for its simplicity compared tobuilding a custom blockchain solution.

A. Ethereum NODE

In our system, both the CLI component and the *DNS* Server component need to connect to an *Ethereum* node. For testingpurposes, we deployed a local node on our test machine and used the Rinkeby test network's faucet to fund the local wallet, keeping the workload light as this is a proof-of-concept. This setup also allowed us to run the *Ethereum* node in light-mode. While a custom blockchain might be a more tailored option for a purpose-built environment, we opted for *Ethereum* in the interest of simplicity.

B. SMART CONTRACT

In the context of *Ethereum*, Solidity is utilized to define smart contracts. The specific smart contract we employed for this deployment is structured as follows:

```
Smart Contract Written in

Solidity pragma solidity

    0.6.3; contract

    Inet DNS Record {

mapping (string => mapping(uint16 => string))
        internal _DNSMapping;

function addRecord(string memory key, uint16
recType, string memory recValue)  public  {

        _DNSMapping[key][recType] = recValue;

                        }

function getRecord(string memory key, uint16
recType) public view returns (string memory){

          return _DNSMapping[key][recType];}}
```

The code snippet above defines a dictionary-like structure forstoring records and includes two methods for setting and getting records. Each time a record is set, a new transaction is created and the network status is updated. After a short delay, all nodes in the network receive the updated blockchainstatus, akin to "propagating" the record in *DNS* terms. It's worth noting that while the contract shown is simplistic, it is sufficient for supporting A, AAAA, and CNAME records, which were the only record types chosen to be supported in this project.

C. Command-Line Interface (CLI) for the DNS Registrar

The third component of the solution is a user-friendly Command Line Interface (CLI) that allows for easy setting of *DNS* records. When records are inputted through the CLI, the Registrar invokes the addRecord method on the smart contract, which generates a new transaction and saves the newly added records into a file for future reference.

Running the *DNS* Registrar would present the following menu:

```
Pick an option:
1. Set record.
2. Show record.
3. Exit.
4. Reset and exit.
```

Fig. 2. *DNS* Register Main Menu

Setting a record by typing into the CLI:

```
Type in the record type
>A
Type in the record name
>test.home.
Type in the record value
>1.2.3.4
Record set! Please wait for tx
0xe6c9d58eddee24e82cabacf697e3fefa0b87735cc47c5ebea
ee19361c11a8269 to be confirmed.
```

Fig. 3. *DNS* Register Input

The *DNS* Registrar component captures the details of the record type, record name, and its corresponding value. These details are then sent to the *DNS* server, which is based on a smart contract, resulting in a transaction hash that represents the transaction on the blockchain.

After a brief moment, verifying should just be a matter of querying the records by selecting the *'2. Show Records'* option and it should display the following information:

```
Type in the record type
          A
Type in the record name
      test.home.
                  Value: 1.2.3.4
```

Fig. 4. *DNS* Registar Show Records

D. DNS SERVER

The *DNS* server is configured to listen for incoming UDP connections on port 53, and upon receiving a message, it parses the message, queries the smart contract for the requested value, and sends a response back to the caller. To ensure consistency and reduce overhead, we opted to containerize the *DNS* server using a Docker container. This allows for easier deployment and management of the *DNS* server in a controlled environment.

Once the Docker container is launched and operational, testing our *Blockchain DNS Server* simply involved using the *'dig'* command on a stored record. This resulted in the following output:

```
;; Warning: query response not set

; <<>> DiG 9.11.3-1ubuntu1.11-Ubuntu <<>> test.home
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id:
8290
;; flags: rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 317ca66319783675 (echoed)
;; QUESTION SECTION:
;test.home.                      IN      A

;; ANSWER SECTION:
test.home.                  0       IN      A
1.2.3.4

;; Query time: 423 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Tue Apr 18 19:06:47 GST 2023
;; MSG SIZE  rcvd: 65
```

Fig. 5. Running dig on a Stored Record

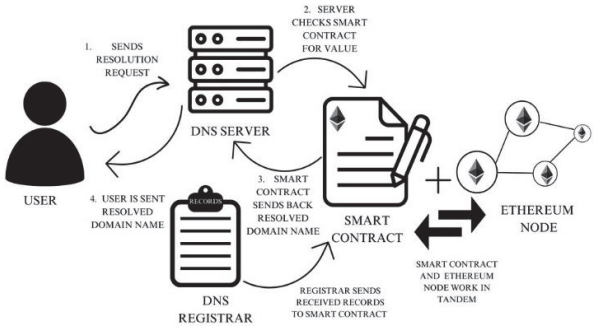

Fig. 6. Proposed Systems' Workflow Diagram

The output trail provides a clear depiction of the 'dig' command performing a comprehensive *DNS* lookup, showcasing response statistics, resolution of A-record queries, and relevant headers. This further substantiates the feasibility of such a system, albeit as a proof of concept, indicating its potential for practical implementation.

E. PERFORMANCE EVALUATION

Blockchain-based *DNS* systems are relatively new and innovative, and their performance, availability, and response times may depend on various factors, such as the specific implementation, consensus algorithm, and network size. Due to the distributed nature of blockchain networks and the use of consensus algorithms, which require agreement among multiple nodes, Blockchain-based *DNS* systems may introduce additional complexities that could potentially impact performance and response times [6]. For example, the processing overhead and latency associated with consensus algorithms, smart contracts, and distributed nodes may result in longer response times compared to traditional *DNS* systems. Additionally, the propagation of *DNS* updates across all nodes in a blockchain network may take time, leading to potential delays in *DNS* availability and response times. As our implementation was limited to a proof-of- concept, we relied on response times and availability asperformance metrics.

Based on our localized implementation, the Blockchain *DNS* Server demonstrated comparable response times to a traditional *DNS* setup. However, occasional slight delayswere observed, which could be attributed to the additional steps involved, such as communication with the blockchain node via a smart contract [11]. This is an apparent disadvantage compared to a traditional *DNS* system. Nevertheless, considering the rapid advancements in the field, we anticipate that this bottleneck can be mitigated overtime.

Traditional *DNS* systems have been optimized for high performance, with low response times, and are widely deployed across the internet. Traditional *DNS* systems typically rely on hierarchical and distributed architecture, with caching mechanisms in place to improve response timesand reduce the load on *DNS* servers. They also benefit from alarge number of *DNS* servers worldwide, which ensures high availability and redundancy. Additionally, traditional *DNS* systems have been thoroughly tested and optimized over the years, resulting in robust and efficient performance.

Recently, [7] Recursive *DNS* resolvers have emerged as an alternative *DNS* methodology that has gained traction in large-scale implementations. A comparison between traditional DNS Response times with and without recursive resolvers is shown in Figure 7 based on the work done in [21].

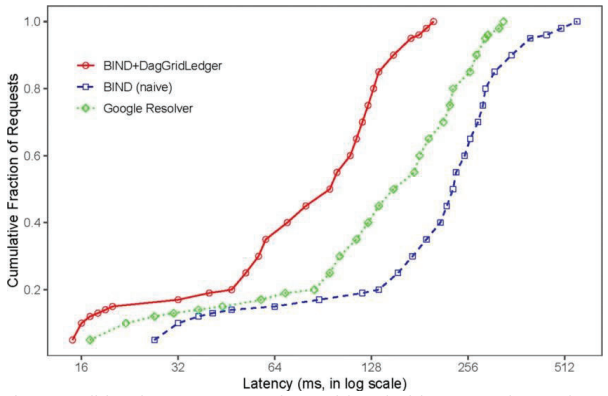

Fig. 7. Traditional DNS Response times with and without recursive resolvers [21].

These resolvers typically have established networks, infrastructure, and caching mechanisms in place to efficiently handle *DNS* queries, resulting in low response times and high availability. They are often optimized for high performance and reliability, with redundant servers, distributed architecture, and sophisticated load balancing techniques to handle a large number of queries from various sources.However, it's important to note that blockchain *DNS*is still a nascent technology, and ongoing research and development efforts may address these limitations in the future. With advancements in blockchain protocols, consensus mechanisms, and optimization techniques, it's possible that blockchain *DNS* systems could achieve comparable or even superior performance, availability, and response times as compared to traditional recursive *DNS*resolvers. Additionally, the inherent security features of blockchain, such as immutability and distributed consensus, could provide enhanced security and resilience to *DNS* systems, [11] which may offset some of the performance trade-offs. Further studies and real-world implementations are needed to evaluate the long-term performance and viability of blockchain *DNS* in comparison to traditionalrecursive *DNS* resolvers.

V. POTENTIAL OPERATIONAL CHALLENGES OF IMPLEMENTING BLOCKCHAIN FOR *DNS*

Although it has been demonstrated that blockchain technology holds promise for offering various benefits to domain name systems (*DNS*), it is also accompanied by operational challenges that could need addressing.

A. SCALABILITY

Blockchain networks, such as *Ethereum*, which are commonly used for blockchain-based *DNS* solutions, may face limitations in terms of transaction processing speed and scalability, potentially affecting the [13] performance of the *DNS* system during periods of high demand.

B. GOVERNANCE AND REGULATORY CHALLENGES

Blockchain-based *DNS* systems operate on a decentralized and distributed network, where decisions are made through consensus among participants. This distributed nature may make it challenging to establish clear governancemechanisms, decision-making processes, and dispute resolution mechanisms. Determining who has the authority to make decisions, how decisions are made, [16] and how disputes are resolved can be complex in a decentralizedenvironment.

Additionally, existing regulatory frameworks for *DNS* may not adequately address the unique characteristics of blockchain-based *DNS* systems. These systems may raise legal and regulatory concerns related to issues such as data privacy, data ownership, intellectual property rights, and liability. For example, the immutability of blockchain may raise challenges related to data deletion and the "right to be forgotten" under data privacy regulations. Moreover, blockchain-based *DNS* systems may need to comply with new regulations that are specifically designed for blockchain technology. [19] Regulatory bodies may imposerequirements on blockchain-based *DNS* systems to ensure compliance with existing laws, regulations, and policies, which can add complexities to the operation of these systems.

Another aspect of governance and regulatory challenges is the potential lack of accountability and transparency in blockchain-based *DNS* systems. Due to the pseudonymous nature of blockchain transactions, it may be challenging to identify and hold accountable the parties responsible for malicious activities, such as domain name squatting, phishing, or other forms of abuse. This may require theestablishment of new governance mechanisms, such as reputation-based systems or identity verification protocols, to enhance accountability and transparency in blockchain- based *DNS*. [20] Additionally, regulatory frameworks and compliance requirements may need to be developed or adapted to address the unique characteristics and challenges of blockchain-based *DNS* systems, ensuring that theycomply with relevant laws and regulations related to data privacy, cybersecurity, intellectual property, and consumer protection.

C. INTEROPERABILITY AND STANDARDIZATION

In the context of blockchain-based *DNS*, interoperability challenges can arise due to the lack of standardized protocols, formats, and conventions across different blockchain networks. [11] Each blockchain platform may have its own unique technical specifications, consensus algorithms, and smart contract languages, which can hinder the exchange of data and transactions between different blockchain-based *DNS* systems.

Furthermore, achieving interoperability between blockchain-based *DNS* systems and traditional *DNS* systems, which are currently in widespread use, can also be challenging. Traditional *DNS* systems follow the hierarchical structure and use the domain name hierarchy, such as the root zone, top-level domains (TLDs), and second-level domains (SLDs), which may not align with the structure and conventions of blockchain-based *DNS* systems. Bridging the gap between these two different systems, which may have different technical architectures, governance models, and operational procedures, can pose significant challenges.

Interoperability challenges in blockchain-based *DNS* systems may also [14] impact the scalability and usability of these systems. If different blockchain-based *DNS* systems cannot seamlessly communicate and exchange data with each other or with traditional *DNS* systems, it may limit the reach and effectiveness of these systems in the broader internet ecosystem.

D. COST AND RESOURCE CONCERNS

Implementing and operating a blockchain-based *DNS* system will involve various costs and resource requirements. Some of the key considerations include infrastructure costs, transaction fees, development costs, operational costs,training and expertise, and scalability challenges. Infrastructure costs are associated with setting up and maintaining the necessary blockchain infrastructure, such as nodes, wallets, and network connectivity. Robust hardware and software resources may be required to ensure reliable performance and security of the blockchain network.

Transaction fees are often required in many blockchain networks to incentivize network validators and miners. These fees can vary in amount and may impact the overall cost of using the blockchain-based *DNS* system, especially in high-traffic scenarios. Also, development costs, which arerelated to the creation and maintenance of the software and smart contracts required for the blockchain-based *DNS* system. Specialized technical expertise and ongoing development efforts may be needed, which can contribute tothe overall costs.

Operational costs will also involve ongoing expenses for monitoring, maintenance, and security measures to ensure the smooth functioning and protection against potential security threats of the blockchain-based *DNS* system. And asany other industry, training and expertise will be required for personnel involved in operating the blockchain-based *DNS* system. Acquiring and maintaining the necessary skillsand knowledge may involve additional costs and resource considerations.

VI. POTENTIAL SECURITY CHALLENGES OF IMPLEMENTINGBLOCKCHAIN FOR *DNS*

As mentioned earlier, while Blockchain-based *DNS* presentsdistinct characteristics that set it apart from traditional *DNS*, indicating that traditional *DNS* may be outdated in comparison to the innovative Blockchain *DNS* [15], it's important to consider potential threats and attack vectors associated with Blockchain-based *DNS* [16]. One significant potential advantage of Blockchain-based *DNS*s is their applicability to malware [2]. These technologies allow for the registration of a large number of domains with low entropy, which may not be available in the regular market [17]. Currently, malware authors use algorithmically generated domains (AGDs) known as Domain Generation Algorithms (DGAs) [11] to generate domain names. However, due to the unavailability of short and meaningful domain names, they resort to using long and random-looking domain names. As a result, a compromised host that uses a DGA to resolve the C2 (Command and Control) server issues numerous Non-Existent Domain (NXDomain) [12] requests, which can be analyzed to attribute them to the proper DGA efficiently. Moreover, the use of Blockchain-based *DNS*s [1] presents further challenges for malware analysts. During static analysis of the malware and its reverse-engineered code, the analyst and the tools used must be aware of the new domains introduced by Blockchain-based *DNS*s. Another potential risk is the 51% attack, which involves a group gaining control of more than 50% of the hashingpower, the computing power used to solve the cryptographic puzzle, of a blockchain network. With majority control, this group can introduce a modified version of the blockchain at aspecific point, which could be accepted by the network due to their dominant influence.

However, altering historical blocks, which contain transactions that were confirmed before the attack, would be extremely challenging, especially the further back in the blockchain's history these transactionsare. Transactions made before a checkpoint, where transactions become permanent in Bitcoin's blockchain, would be impossible to change. While a well-funded attackercould potentially execute a 51% attack on the Bitcoin blockchain, the high cost of acquiring sufficient hashing power generally serves as a deterrent against such attacks. It's important to note that 51% attacks are rare in practice.

In addition to the aforementioned risks, implementing blockchain into *DNS* also introduces new security risks. For instance, the compromise of private keys used for managing *DNS* records could result in unauthorized changes. Smart contracts, commonly used in blockchain-based *DNS*, may also be vulnerable to exploits, leading to potential security breaches. It is imperative to thoroughly evaluate and mitigate these security risks when implementing blockchain-based*DNS* systems, and to ensure proper measures are in place to safeguard the integrity and confidentiality of *DNS* data and operations. This highlights the need for robust security measures and best practices to be implemented in the design, development, and operation of blockchain-based *DNS* systems to ensure their secure and reliable functioning.

CONCLUSION

This research paper has introduced an innovative and decentralized domain name resolution system that utilizes smart contracts on the *Ethereum* blockchain. Through the use of a light *Ethereum* node, a smart contract, and a containerized docker application, a minimally functional Blockchain *DNS* server was proposed, which offers a trusted *DNS* registration process. The findings of this study suggest that the proposed *DNS* mechanism has the potential to be scaled up and effectively deployed in a live environment, demonstrating its viability.

Furthermore, this research opens up opportunities for future exploration and advancements in the field of domain name systems. The proposed mechanism holds promise for building domain name systems not only for wide area networks, but also for local area networks or intranets, utilizing the decentralized nature of blockchain-based *DNS* solutions. This could lead to the construction of the next-generation *DNS* infrastructure, which has the potential to enhance security, transparency, and reliability in domain name resolution processes.

This research contributes to the body of knowledge on blockchain-based *DNS* solutions and presents a viable and scalable approach for domain name resolution. As the field of blockchain technology continues to evolve, further research and development in this area could potentially revolutionize the way domain name systems are managed and operated. The decentralized nature of blockchain-based *DNS* solutions offers new possibilities for addressing the limitations of traditional *DNS* systems, such as vulnerability to cyber attacks and lack of transparency. By leveraging the power of blockchain technology, domain name systems could become more secure, transparent, and decentralized inthe future. In conclusion, this research paper has proposed a novel approach for domain name resolution using blockchain technology, demonstrating its potential for scalability and effectiveness. The findings of this study provide a foundation

for further research and development in the field of blockchain-based *DNS* solutions, with the goal of transforming the way domain name systems are managed and operated, leading to a more secure, transparent, and decentralized *DNS* infrastructure in the future.

REFERENCES

[1] A. Klein, H. Shulman, and M. Waidner, "Internet-Wide Study of *DNS* Cache Injections," in IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, 2017, pp. 1–9

[2] S. Gao, Z. Peng, B. Xiao, A. Hu, Y. Song, and K. Ren, "Detection and Mitigation of DoS Attacks in Software Defined Networks," IEEE/ACM Transactions on Networking, vol. 28, no. 3, pp. 1419–1433, 2020.

[3] Blockchain-*DNS*, "Blockchain-*DNS*," 2019. [Online]. Available: https://blockchain-*DNS*.info

[4] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. IEEE Transactions on Networking, 11, Feb. 2003.

[5] H. Shulman and M. Waidner, "One Key to Sign Them All Considered Vulnerable: Evaluation of *DNS*SEC in the Internet," in 14th USENIX Symposium on Networked Systems Design and Implementation. USENIX Association, 2017, pp. 131–144.

[6] H. A. Kalodner, M. Carlsten, P. Ellenbogen, J. Bonneau, and A. Narayanan, "An Empirical Study of Namecoin and Lessons for Decentralized Namespace Design," in WEIS. Citeseer, 2015, pp. 1–21.

[7] R. Perdisci, M. Antonakakis, X. Luo, and W. Lee, "WSEC *DNS*: Protecting Recursive *DNS* Resolvers from Poisoning Attacks," in 2009 IEEE/IFIP International Conference on Dependable Systems & Networks. IEEE, 2009, pp. 3–12.

[8] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," Tech. Rep., 2008

[9] Jung, J., Berger, A. W., & Balakrishnan, H. 2003. Modeling TTL-based Internet caches. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies(March, 2003). 417--426. DOI= http://doi.acm.org/10.1109/INFCOM.2003.1208693.

[10] X. Wang, K. Li, H. Li, Y. Li, and Z. Liang. Consortium*DNS*: A distributed domain name service based on consortium chain. In Proceedings – 2017 IEEE 19th Intl Conference on High Performance Computing and Communications, HPCC 2017, 2017 IEEE 15th Intl Conference on Smart City, SmartCity 2017 and 2017 IEEE 3rd Intl Conference on Data Science and Systems, DSS 2017, volume 2018-January, pages 617–620, 2018.

[11] E. Politou, F. Casino, E. Alepis, and C. Patsakis. Blockchain mutability: Challenges and proposed solutions. IEEE Transactions on Emerging Topicsin Computing, pages 1–1, 2019.

[12] S. Pletinckx, C. Trap, and C. Doerr. Malware coordination using the blockchain: An analysis of the cerber ransomware. In 2018 IEEE Conference on Communications and Network Security, CNS 2018, pages 1–9, 2018.

[13] J. Liu et al. A data storage method based on blockchain for decentralization *DNS*. In Proceedings - 2018 IEEE 3rd International Conference on Data Science in Cyberspace, DSC 2018, pages 189–196, 2018.

[14] Hui Li et al. Systems and methods for managing top-level domain names using consortium blockchain, 2019. US Patent App. 10/178,069.

[15] A. Hari and T.V. Lakshman. The internet blockchain: A distributed, tamper-resistant transaction framework for the internet. In HotNets 2016 - Proceedings of the 15th ACM Workshop on Hot Topics in Networks, pages 204–210, 2016.

[16] Z. Guan, A. Garba, A. Li, Z. Chen, and N. Kaaniche. Authledger: A novel blockchain-based domain name authentication scheme. In ICISSP 2019 - Proceedings of the 5th International Conference on Information Systems Security and Privacy, pages 345–352, 2019

[17] Anwaar Ali, Siddique Latif, Junaid Qadir, Salil Kanhere, Jatinder Singh, Jon Crowcroft, et al. Blockchain and the future of the internet: A comprehensive review. arXiv preprint arXiv:1904.00733, 2019.

[18]. Muma, S., Kappos, D., & Sumroy, R. (2012). The Right to Be Forgotten Meets the Immutable - A Practical Guide to GDPR-Compliant Blockchain

[19]. Distributed, C., & Union, E. (n.d.). Distributed Ledger Technologies and Data Protection in the European Union. 1–57.

[20] Compert, C. M. L. (@MauLui) B. P. (@lebertrand). (2018). Blockchain and GDPR. 1(1), 8–23.

[21] Chen, Wenyu & Yang, Xue & Zhang, Haikuo & Xu, Yanzhi & Pang, Zihan. (2021). Big Data Architecture for Scalable and Trustful DNS based on Sharded DAG Blockchain. Journal of Signal Processing Systems. 93. 10.1007/s11265-021-01645-3.n, H. Shulman, and M. Waidner, "Internet-Wide Study of DNS Cache Injections," in IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, 2017, pp. 1–9.