Use Chains to Block DNS Attacks: A Trusty Blockchain-based Domain Name System

Wen-Bin Hsieh, Jenq-Shiou Leu *(Senior Member, IEEE)* and Jun-Ichi Takada *(Senior Member, IEEE)*

***Abstract*—The Internet has become one of the most important technologies in the world, and hackers use various methods to launch cyber attacks to profit from it. Phishing is one of famous social engineering attacks, it is often used to steal user data, including login credentials and credit card numbers. Although the Transport Layer Security certificate is used to verify the trust of websites, there are still a series of vulnerabilities. The demand for trusted IP addresses has led a lot of research, including IP whitelisting, DNS filtering and so on. However, these technologies still have many shortcomings. In view of this, we proposed a novel mechanism for verifying websites using blockchain technology. The URL and IP address of a permissioned website are recorded in blockchain through a specific smart contract. A DNS query is executed through a smart contract designed to avoid URL redirection attacks. With the help of immutable nature of blockchian, phishing websites can be detected. The mechanism will not add any load to users and provides tamper-proof functions based on the characteristics of blockchain. The comparison of related works shows that the proposed mechanism is more secure. We also provided a reference implementation of the proposed mechanism on Ethereum Quorum simulation platform, which proves the effectiveness and practicability of the mechanism.**

***Index Terms*—Blockchain, DNS security, Ethereum, smart contract.**

I. Introduction

INTERNET has been essential without doubt in the twenty-first century. Since the outbreak of the coronavirus (COVID-19) in spring 2020, 53 percent of Americans and 34 percent of Austrians stated they used the Internet more than usual. In a global survey of CIOs conducted in March 2021[28], 70 percent of respondents said that they currently work from home. Additionally, about 30 percent stated they are expecting to be working remotely permanently. Benefit from Internet, employees can work at home, students can learn remotely in the epidemic area. Conceivably, the importance of Internet has made it become the target of hacker and terrorist attacks. Cybercriminals use a variety of techniques to launch cyber attacks, including phishing, malware, denial of service and other techniques. The most common of these attacks is phishing, where criminals impersonate legitimate websites to trick users into obtaining sensitive information or data, such as usernames, passwords and credit card details. [1] reports a new phishing site launches every 20 seconds and [2] narrates 74% of phishing websites are served via HTTPs protocol. In [3], Microsoft warned users that VeriSign erroneously issued two digital certificates to someone masquerading as a Microsoft representative, giving the deceitful party the ways to trick users into a web site running malicious programs. [4] reported a new technique that visitors to a domain compromised by the attacker see a screen illustrated as Fig. 1. Victims are urged to install a "security certificate update" and two variants of malware are downloaded to steal victim's PC information.

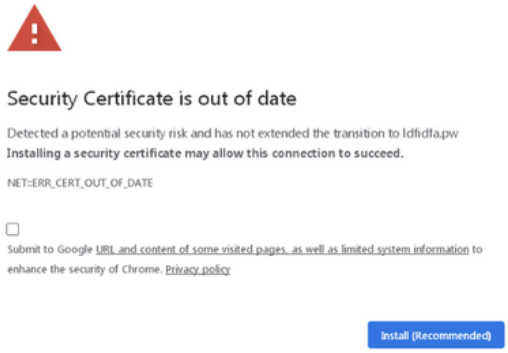

Fig. 1. Security certificate update.

In early 2020, the largest popular free certificate authority Let's encrypt claims to revoke 3 million TLS certificates because of a certificate authority authorization (CAA) bug [5]. In order to assure the trusty of IP address, lots of methods were proposed, including IP whitelisting, DNS filtering and so forth. IP whitelisting creates a list of trusted IP addresses which users can use to access these domains. Nevertheless, maintaining an up-to-date whitelist of IP addresses can be very labor intensive and unexpected IP address changes can cause downtime. In contrast, DNS filtering utilizes blacklists to block a particular website or IP address which is known to be malicious. However, it's easy to circumvent DNS filtering by means of using a different DNS or adding entries to the host file. With the vigorous development of the emerging technology of blockchain, many researches also tried to utilize its properties to provide a secure DNS mechanism. The first

Manuscript received July 15, 2021; revised February 7, 2022; approved for publication by Jeongyeup Paek, Division III Editor, February 24, 2022.

The authors gratefully acknowledge the support by the Taiwan Tech-Tokyo Tech Joint Research Program, under Grant TIT-NTUST-109-01.

W.-B. Hsieh is with the Department of Electronic and Computer Engineering, National Taiwan University of Science and Technology, Taipei City 106335, Taiwan (R.O.C.). He is now a researcher focusing machine learning, information security and cryptography. (e-mail: d9802106@mail.ntust.edu.tw).

J.-S. Leu is with the Department of Electronic and Computer Engineering, National Taiwan University of Science and Technology, Taipei City 106335, Taiwan (R.O.C.). (email: jsleu@mail.ntust.edu.tw).

J.-I. Takada is with the Department of Transdisciplinary Science and Engineering, School of Environment and Society, Tokyo Institute of Technology, Japan. (email: takada@tse.ens.titech.ac.jp)

Digital Object Identifier: 10.23919/JCN.2022.000009

work is proposed by Hari *et al.* [6] who developed a DNS infrastructure depend on PKI heavily. Benshoof *et al.* [7] proposed a system named distributed decentralized domain name service (D^3NS) which is based on a distributed hash table and utilizes a domain name ownership system based on the Bitcoin blockchain. In 2018, Liu *et al.* [8] also took advantage of decentralization of blockchain to present a DNS resolution method which mitigates single points of failure and domain name resolution data tampering. Yu *et al.* [9] further proposed a DNS Cache Resources Trusted Sharing Model abbreviated as DNSTSM. They claimed the model can improve the credibility of DNS resolution results. Meanwhile they presented a stochastic distributed decentralized storage mechanism to solve the problem of low efficiency in the consortium blockchain.

After studying the existing advanced solutions, we found the main issue is the feasibility of these proposals. These studies need to greatly change the framework of the present DNS system or add its loads. Thus, we provide a novel mechanism to validate websites based on the advantages of blockchain. This mechanism is based on the existing DNS architecture, not only does not increase the burden on users, but also easy to implement. The major contribution of this paper is as follows:

1) Using Quorum blockchain to build a novel Domain Name system, only a permissioned website's URL and IP address can be recorded in blockchain through a specific smart contract.
2) Design smart contracts to insert or query a domain name. With the advantages of consortium blockchains and smart contracts, the proposed model combined with designed smart contracts can resist current cyber attacks such as phishing, URL redirection, and DNS spoofing.
3) Reduce power consumption problems of existing works. Compared with related works, the proposed mechanism using Quorum blockchain and Raft consensus algorithm can provide better security, fast verification and low power consumption.

The structure of this paper is as follows: The second section presents a brief review of the related research of the current blockchain-based DNS and explained the problems existing in current DNS services. We also summarize our contributions in this part. In the third part, we introduce the consortium blockchain technology and the concept of smart contract. Our DNS resolution mechanism is proposed in fourth part. The fifth section analyzes security and performance. We give the conclusion in the sixth part.

II. Related Work

In [10]–[14], decentralized systems are proposed to improve the robustness and availability of domain name resolution, and to bypass the censorship mechanism and tampering. Authors in [10] propose a solution to distribute signature keys through threshold cryptography. Therefore, the signature key can remain online, and "online signature" becomes possible. CoDoNS [11] restricts its proposal to fast lookup and resilience to attacks through proactive caching, but does not consider updates. Privacy issues are also not taken into account by them. The common drawback of these methods is poor performance. These approaches are computationally too expensive when the signer needs to provide services to a large number of users who generate requests constantly. Until recently, a distributed DNS solution [15] was designed based on the credibility, verifiability, and immutability of blockchain technology. However, only a brief overview of blockchain-based DNS and some skepticism was initially provided in this work. After that, a large number of blockchain related research has been proposed. In 2017, Hari *et al.* [6] utilized blockchain to propose a mechanism to secure the DNS infrastructure. The authors claim that the mechanism provides a scalable, distributed, and temper-resistant mechanism for managing Internet resources without using a public key infrastructure (PKI) system. Nevertheless, the proposal restricts itself to the BGP advertisements mainly. Benshoof *et al.* [7] presented a system called distributed decentralized domain name service (D^3NS) to replace the current top-level DNS system and certificate authorities. D^3NS is based on a distributed table and uses Bitcoin blockchain to implement a domain name system. The authors claim D^3NS provides solutions for current DNS vulnerabilities such as DDoS attacks, DNS spoofing and censorship of local governments. The significant drawback is that Bitcoin is extremely slow and power consumption is high. In [16], Gourley and Tewari utilize blockchain to enhance the certificate validation procedure to improve DNS security extensions, providing the same security advantages as DNSSEC while addressing its main drawbacks. In order to weaken the level of trust to the CAs over certificates, Guan *et al.* [17] presented a domain authentication scheme based on blockchain technology, called AuthLedger. The authors use smart contracts in Ethereum to implement the system. Considering that the public blockchain consumes a lot of computing power, and a domain registration requires a permission, we use the consortium blockchain to propose a new model to solve these problems. In 2020, Yu *et al.* [9] proposed a novel DNS cache resources trusted sharing model (DNSTSM) to improve the credibility of DNS resolution results. In DNSTSM, the trust-based incentive mechanism aims to reduce the impact of free-riding behavior and on trusted performance of the system. The authors proposed a multi-DNS recursive servers lookup mechanism (MDRSLM) to return the appropriate IP for the user. The architecture is based on Hyperledger Fabric permissioned blockchain infrastructure which implements a (practical Byzantine fault tolerance (PBFT) consensus. However, this model is based on an older version of Hyperledger Fabric, which cannot create a complete privacy-preserving infrastructure and is cumbersome. In [29], Liu *et al.* proposed a blockchain-based decentralized DNS resolution method with distributed data storage to reduce single point of failure and domain name resolution tampering. The proposed mechanism is primarily focuses on the decentralization of the centralized authoritative domain servers. However, our proposed mechanism focuses more on sharing trusted DNS resources through a peer-to-peer network to improve the trustworthiness of existing

DNS infrastructure and resolution results.

In summary, the aforementioned works attempt to propose a robust DNS to resist various attacks. However, they require major changes to the existing DNS infrastructure and have some limitations. However, the current infrastructure has been adopted for a long time and is difficult to modify. In view of this, the proposed mechanism focuses on utilizing a consortium blockchain and smart contracts to improve the DNS validation procedure without altering the existing DNS infrastructure. The model can resist well-known cyber attacks such as phishing, domain hijacking, DNS spoofing and so on. Furthermore, the proposed model uses Raft consensus algorithm, which consumes much less electricity than the original Ethereum and Bitcoin. Therefore, the mechanism solves the deficiencies of the previous studies and provides more secure and efficient domain name services.

III. Basic Details of Consortium Blockchaind and Smart Contract

In this section, we first give a brief explanation of consortium blockchain technology, and then an overview of how smart contract works is portrayed.

A. Consortium Blockchain Technology

Consortium blockchain, also called federated blockchain, is a blockchain technology open to specific organizations or groups which require participants' registration [18]. Consortium blockchain is permission-controlled. This means that read and write rights are limited to the participating consortium members. The consortium blockchain consists of two nodes, one manages the communication with clients as well as other nodes and the other manages private transactions by performing cryptographic transactions. Since proof of work (PoW) is not required in a permissioned network, consortium blockchains support multiple consensus mechanisms. The public can conduct consultations and transactions, and the permission of the consortium is required to verify transactions or issue smart contracts. Therefore, unauthorized nodes or users cannot access any service in the consortium blockchain, and authorized nodes or permissioned users can participate in the execution of transactions and smart contracts. Basic network permission [19] is a function that controls which nodes can connect to a given node and also to which nodes a given node can dial out to. Ethereum Quorum [20], Hyperledger Fabric [21], and FISCO BCOS [22] are the most popular consortium blockchain platforms, which have attracted a large amount of investment worldwide and have been researched and applied in many fields.

B. AN OVERVIEW OF SMART CONTRACT

A smart contract is simply a computer program stored on a blockchain and is a self-executing contract containing the terms and conditions of an agreement among peers. Smart contracts are not controlled by any user, but are deployed in the blockchain network, with self-verification and tamper-resistant properties [23]. Smart contracts are triggered by transactions submitted by user accounts which execute functions defined on smart contracts. No third party, such as a broker or an authority, is required to participate in the execution. All transaction information is presented in the smart contract and is traceable and irreversible. Smart contracts are distributed and ensure high availability by eliminating a single point of failure. The importance of smart contracts integrated with blockchain technology has become the focus of development because the transactions and databases can be maintained publicly in a secure and trusted environment. Nowadays solidity is the most popular programming language used to implement smart contracts in a variety of blockchain platforms. The language will be compiled into EVM executable bytecode and users can interact with it through application binary interface (abi). Blockchain-based smart contracts can provide many advantages such as speed, accuracy, lower execution risk and cost.

C. Consensus Algorithm

A consensus algorithm is a fault-tolerant mechanism that is used in blcokchain systems to reach an agreement to perform secure updates. A basic blockchain technology is state machine replication. Since the state is shared among several replicas within the network, the execution of the state will eventually result in the same output. Consensus helps replicas determine the finality of each state. However, the implementation of consensus in blockchain systems is complicated as it requires a consensus algorithm to maintain adversity resilience, failure tolerance and other important properties. The PoW is a common consensus algorithm used by the most popular cryptocurrency networks such as bitcoin and litecoin. In order to discover the target nonce in PoW, it is inevitable to waste computational power as a consequence for creating a new block. For the purpose of energy-saving, many consensus algorithms have been proposed. In 2017, Istanbul Byzantine fault tolerance (IBFT) was first introduced into Quorum. IBFT is an implementation of the practical Byzantine fault tolerance algorithm with modifications that can achieve distributed consensus without carrying out complex mathematical computations. IBFT is very energy-efficient and gives more room for more transactions. Raft is an underlying consensus algorithm that was initially adapted by Quorum to provide a crash fault tolerance (CFT). Raft is a simplified extension for the Paxos algorithm. It allows a leader node to generate the next block and eliminates the generation of unnecessary vacant block. A node in a system can only be in one of the three states at any point in time: Leader, follower, and candidate. All the followers replicate the entries proposed by the leader with no doubt. Unless there are pending transactions, Raft consensus will not generate blocks. This can significantly save storage space, especially in the case of low transaction load, because empty blocks containing zero transactions will not be minted. The comparison of throughput and power consumption of above algorithms is summarized in Table 1.

TABLE I
THE COMPARISON OF THE THREE CONSENSUS ALGORITHMS: POW, IBFT, RAFT

Algorithm	PoW	IBFT	Raft
Power consumption	High	Low	Low
Throught	Low	Moderate	High

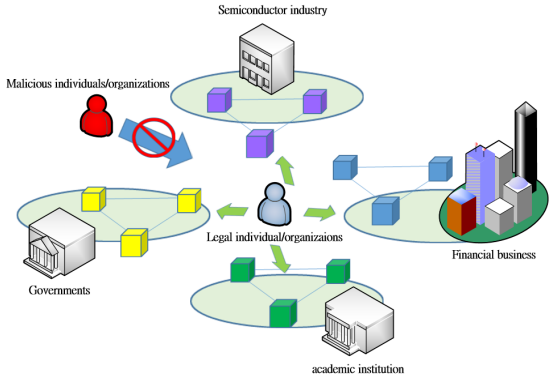

Fig. 2. Architecture overview - business alliances construct consortium blockchains.

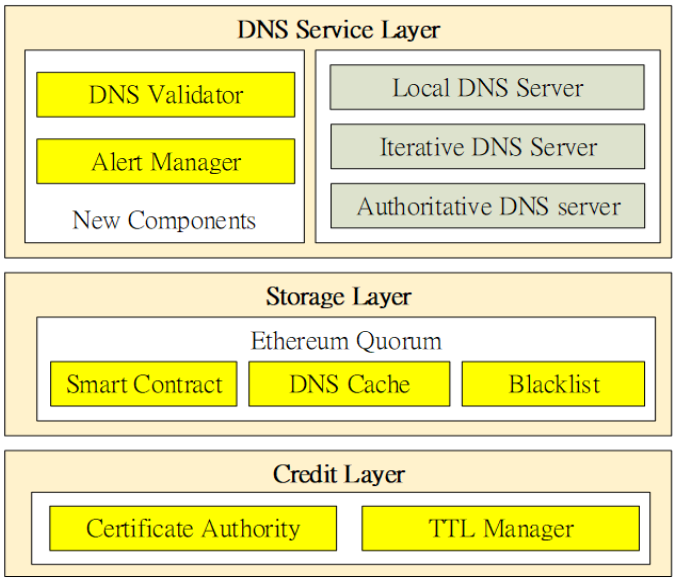

Fig. 3. The overall function module of the blockchain-based DNS system.

IV. PROPOSAL OF TRUST BLOCKCHAIN-BASED DNS SYSTEM

A. An Overview of System Architecture and Domain Attack Related Work

The conceptual system architecture of the solution is shown in Fig. 1. Every business field and industry gathers to construct their own consortium blockchain. They uploaded their uniform resource locator (URL) and Internet protocol (IP) to blocks in chain and nobody can modify theirs wantonly. Before joining the consortium blockchain, a new individual or organization will be investigated and reviewed by the consortium first. If their personal/organization data is valid, the content of the website and URL is correct and cogent, they are allowed to join the consortium and trigger a smart contract to make a transaction that uploads their URL and IP to a block. A malicious attacker cannot pass the authentication because of lacking concrete official business data. The threat will be isolated from the trusted network to ensure the credibility of the user's domain name resolution results.

In Fig. 3, we present the overall function module of the proposed Blockchain-based DNS System (BDS). The system is divided into three layers, i.e. DNS service layer, storage layer and credit layer. The DNS layer consists of conventional DNS servers and innovative services such as DNS validators and alert managers. The DNS validator is used to check if the returned response from the local recursive server is consistent with the cached domain name information. If the cached information is inconsistent, the alert manager is triggered to rank it as high-risk resolution information. In the storage layer, instead of SQL databases, we use Ethereum Quorum blockchain to store the uploaded information related to domain name. Quorum is a distributed decentralized storage that improves existing blockchain solutions and provides better performance, proper peer and network management and voting-based consensus mechanisms. The credit layer is designed to provide a credible authority which is responsible for issuing and revoking the certificate of each node. The Time-To-Live manager is added to maintain the life cycle of cached information. We summarize the most common attacks in the threat model in Table II which include access control and DNS security.

TABLE II
THREAT MODEL - THE MOST COMMON ATTACKS

Access control	DNS security
Identity forgery	Domain hijacking
	Cache poisoning
Illegal access	Typosquatting
	DNS hijack

Domain hijacking is performed by exploiting vulnerabilities in the domain name registrar's system, and utilized by attackers to set up a fake website identical to the original. The fake website will record critical personal information such as social security number, email address and so on. Cache poisoning, also known as DNS spoofing, is an attack which an attacker attempts to inject malicious links into your DNS resolver cache to redirect victims to a remote malicious server. Attackers register a domain name that is confusingly similar to an existing famous name is called typosquatting. DNS hijack, often confused with DNS spoofing, refers to injecting malware on the local computer to change the TCP/IP configurations, thereby redirecting traffic to a phishing website. Our solution is effective under these attacks.

B. Role enactment in DNS blockchains

The proposed BDS adopts a leader-follower model where the leader plays the role of a minter that is responsible for bundling transactions into a block and minting new blocks. The leader is elected after a period of voting and during that period all DNS servers are candidates. Once a leader is elected by a majority, the elected DNS server will play the role of the leader, and all other DNS servers will play the role of follower.

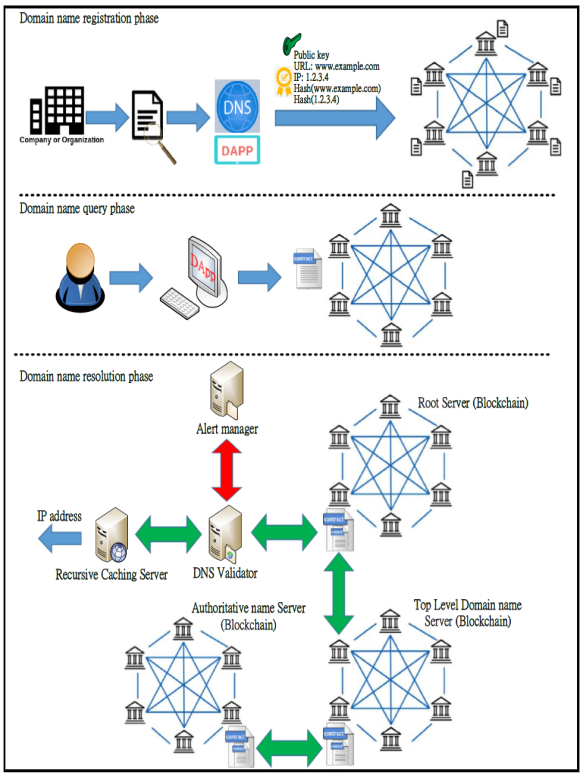

Fig. 4. The whole process of blockchain-based Domain Name Service – 1) Registration phase: Permissioned Companies uploaded their public keys, URL and IPaddress to blockchain. 2) Users send DNS queries from Distributed applictions (Dapp). 3) DNS resolution phase: Replace databases in the existing DNS architecture. The IP, URL and hash values of them are stored in blocks that are chained together. The blockchain is immutable and distributed across a peer-to-peer (P2P) network. The alert manager is used to calculate the risk rating. The DNS validator is used to check the returned response from the local recursive server.

Each DNS server is a node which consists of two sub-modules:

- Transaction manager

TX manager is responsible for storing and allowing access to encrypted transaction data. TX managers exchange encrypted payloads with each other but do not have access to any sensitive private keys. TX manager does not possess any cryptographic module but utilizes the crypto wrapper for cryptographic functionality.

- Crypto wrapper

Crypto wrapper coexists alongside the TX manager and is responsible for generating a symmetric key, encrypting and decrypting transaction payloads. Crypto wrapper plays as a virtual HSM.

C. *The Whole Process of the Blockchain-based DNS*

There are three phases in the proposed DNS mechanism. The first phase is domain name registration, the second phase is domain name query, and the domain name resolution is the third phase. Each phase is described in detail in the following paragraphs.

(A) *Domain name registration phase*

In this phase, individuals or companies who want to register a domain name should submit application documents and business-related information to prove their identities. The consortium will review and evaluate the applicant's documents before the applicant's domain name is added to the blockchain. We must know that the blockchain can guarantee the integrity of the data on the block, but it cannot prove the authenticity of the user who uploaded the data. Therefore, before creating a virtual identity, we need to perform strict authentication in the real world. After the authentication is passed, the applicant will generate a public and private key pair. The public key used to verify the applicant will be sent to the consortium's domain name servers. Then the uniform resource locator (abbreviated as URL) and IP address are hashed and are encrypted by the applicant's private key with padding. Next, the applicant's public key, the original URL and IP address with its hash value are uploaded to a block on the blockchain by a self-executed smart contract initiated by the decentralized application. The pseudocode of the smart contract is presented as **Algorithm 1**. The contract will use the public key to decrypt the signature which is the encrypted hashed value. Furthermore, it hashes the original URL and IP address to get a new hash value. Finally, compare the new hash value with the decrypted hash value. If they are equal, the uploaded data will be written into a block, otherwise, it will be rejected.

(B) *Domain name query phase*

In order to improve the query performance, all consortiums will construct the top-level domain (TLD) blockchain. A user types a URL in a DApp which interacts with a smart contract deployed on the block. After that, the smart contract will send a DNS query to the TLD blockchain. The user's behavior has not changed much.

(C) *Domain name resolution phase*

The overall process of this phase is presented in Fig. 3. The root blockchain is built by 12 institutions, including Internet corporation for assigned names and numbers (ICANN). A DNS query initiated by a smart contract of a user's DApp will be sent to the root blockchain first. The root blockchain returns the address of the smart contract in the target TLD blockchain. Then the target smart contract in the TLD blockchain is automatically triggered to search for the location of the authoritative blockchain. In the same way, the address of smart contract in the target authoritative blockchain will be returned. Next, the returned smart contract will be launched in the current contract. Finally, the IP address of the URL is passed to the user through the smart contract on the query path; the pseudocode of the query is presented in **Algorithm 2**. There is no way to falsify the data in the process because of the features of a smart contract which are automatic execution and security. Based on the returned IP address, the DApp will connect to a trusted website.

In the query process, if the domain name returned by the authoritative name server is resolved for the first time, the DNS validator simply writes it into the recursive caching server. If the response from the authoritative name server is inconsistent with the cached information in the recursive caching server, the alert manager is activated to rank it as high-risk domain

Algorithm 1 *Smart Contract* – Insert (Pseudocode)

```
1. procedure Insert(STRING URL, STRING IPaddr,
   STRING reserveYears, BYTES Sig, BYTES PKey,
   STRING OwnerInfo)
2.    bool created = false
3.    if URL exists || IPaddr exists:
4.       return created
5.    else:
6.       BYTES hashurl = hash256(URL)
7.       BYTES haship = hash256(IPaddr)
8.       STRING msg = encode.Packed(hashurl, haship)
9.       if Eth.Decrypt(PKey, Sig) <> msg:
10.         return created
11.      else:
12.         uint ttl = getTTLValue(reserveYears)
            created = createNewDNSEntry(URL, IPaddr, ttl,
                      Ownerinfo)
13.         Return created
```

Algorithm 2 *Smart Contract* – Query (Pseudocode)

```
1. procedure Query(STRING URL)
2.    STRING result = []
3.    if URL not exists:
4.       result.push("404 Not Found")
5.    else:
6.       result.push(IPaddr[URL])
7.    return result
```

name information. The IP address will be blocked before the consortium validates the correctness of the domain name and the registered IP address. In the case of no data found, the smart contract will return a prompt message as shown in **Algorithm 2**.

V. Performance Evaluation and Security Analysis

In our experiments, we implemented a smart contract that allows permissioned members in the consortium blockchain to register IP addresses and URLs. In addition, we also implemented a smart contract that performs DNS query resolution. The benchmark shows the concept of DNS smart contracts is feasible. The simulation result is presented and explained in (A). Moreover, we reference [24], [25] to describe the performance of Quorum in different consensuses. Finally, we make an in-depth analysis to explain the security features of the proposed mechanism.

(A) *Simulation of DNS smart contract*

The simulation environment is relatively simple. All nodes run on a 2.5GHz CPU virtual machine, using Ubuntu 20.04 LTS system, with 12G memory. In Fig. 6, we illustrate the deployment of our smart contract which is written in solidity 0.8.7. The contract address is used to interact with decentralized applications (Dapp). We use Ganache-CLI v6.12.2

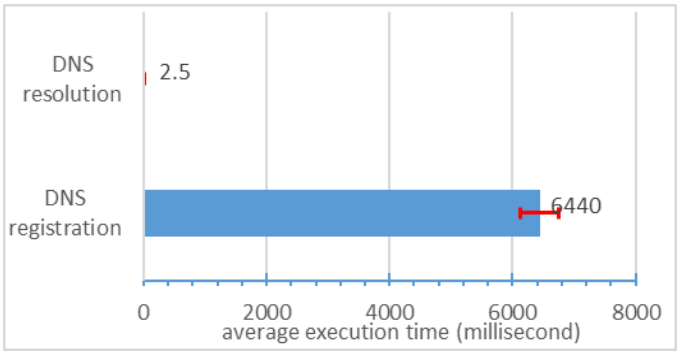

Fig. 5. The simulation results of the smart contracts.

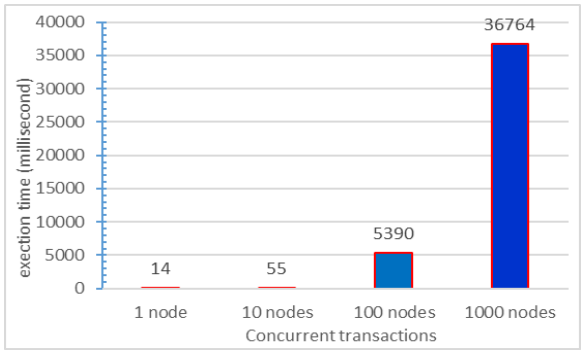

Fig. 6. Execution time of concurrent transactions.

as a blockchain emulator and Truffle v5.3.2 to compile and deploy the smart contracts. The simulation result illustrated in Fig. 5 shows that the average time to write the registration information into the block is 6.44 s. This process takes a while. However, the registration of an IP address and a URL is only a one-time job. DNS query resolution using smart contracts takes 2.5 ms on average. The standard deviations are smaller than 5%.

We must use abi and the hash address of the smart contract to perform contract functions. Each transaction will generate a block on Ethereum test-net. Fig. 5 shows the execution time of concurrent transactions. As the number of the nodes increases, the execution time will become longer. However, every member of the consortium will contribute one contract address to reduce query load. Fig. 7 shows the deployment result of the smart contract. Figs. 8 and 9 present how to use the designed smart contract to register a domain name with an IP address and the result of sending a transaction to create a block. We trim the input to make the illustration of the output clear and set the limit to avoid running out of gas. In reality, the consumption of gas is not the point in the mechanism. The result of finding the IP address of the domain name with the designed smart contract is shown in Fig. 10. A query of the blockchain state need not to send a transaction.

(B) *Performance of the quorum blockchain platform*

The proposed framework is based on Quorum which is a permissioned consortium blockchain protocol. In Quorum, Raft and IBFT (Istanbul Byzantine fault tolerance) consensus algorithms are supported. The performance measurement

TABLE III
EFFECT OF RAFT BLOCK TIME ON THROUGHPUT AND LATENCY

	Block time (ms)			
	100	250	500	1000
Throughput(tx/sec)	752	750	747	748
Latency(seconds)	0.414	0.533	0.589	1.006

refers to [24] that set up the blockchain network with three peers using the Raft consensus algorithm and with four peers in IBFT consensus algorithm. Each peer runs Ubuntu 14.04 LTS operating system on a hardware machine which has a CPU with 4 cores at 3.6 GHz and 16 GB RAM. The Caliper benchmarking tool [27] is enabled by the Quorum plugin to send controlled transactions to the blockchain network to record its throughput and latency. When the Quorum peer issues a block event demonstrating the inclusion of the transaction in the block, the transaction is confirmed. The effect of block time in throughput and latency is shown in Table 3 [24]. The block time of Quorum with Raft consensus is set to 100, 250, 500 and 1000 ms individually. The results show that the throughput is almost the same, which means the throughput is not affected by the block time. Nevertheless, the latency of the transaction is increased distinctly when the block time is raised obviously.

Quorum platform uses Raft as the default consensus algorithm and can be changed to adopt IBFT as needed. The comparison of performance between Raft and IBFT is under the setting of block time of 1 s. Fig. 11 [24] points out that IBFT provides marginally higher throughput than Raft when the transaction load is under 1650 tx/sec. Beyond 1650 tx/sec, the Raft algorithm obtains a slightly better performance than that of IBFT.

In [24], the experimental results show that when the proposed framework adopts the Raft algorithm, DNS registration transactions can be committed at an acceptable rate. With Raft algorithm, an average of 750 transactions can be completed per second. In the domain name resolution phase, the performance can also meet the requirement of DNS queries that do not involve writing. Users interact with a decentralized application (Dapp) that combines a frontend user interface and a dedicated smart contract. There is no significant change in the client side. The client side only needs to install a decentralized application plug-in on the browser to interact with the smart contract. In Fig. 12 [24], the transaction latency for IBFT consensus is significantly higher in compared with that of Raft. Most observation points in IBFT are almost double or more than Raft.

(C) *Security analysis*

The proposed mechanism combing the participant authentication, key management and consensus algorithm of the consortium blockchain to provide a more secure, trusted and reliable DNS resolution service. In this Section, we analyze the security of the proposed program.

We compare our scheme with the previous well-known designs and summarize the comparison result of major features in table 4. The feature of stratification imitates the framework of the modern global domain name system that recursive DNS query is hierarchically sent to different blockchain-based domains.

This feature avoids sending bulk queries to recursive servers which increase the load of network resources and form a service bottleneck. In addition to the feature of the stratification, the comparison result shows that the proposed mechanism has better security than others. The major advantages are as follows:

1) Reliable domain name service

The blockchain can guarantee the integrity and the reliability of data on the chain. However, if the data is uploaded by a malicious applicant, then the fake domain name gets the same protection. In our mechanism, the applicant is audited in accordance with legal documents of application before the domain name is registered. The on-chain data is managed by the consensus of the consortium. The block structure contains a timestamp and the hash of previous block (prehash). Changes to any block require the modification of all on-chain data which is infeasible. If the applicant registers a new domain name with the same IP address, the new block will have a fresh timestamp that can be checked easily. Information on the chain is shared clearly and protected based on the features of blockchain, the proposed DNS can offer reliable service consequently.

2) Trust key-based authentication

Based on a permissioned blockchain, only authenticated nodes can participate in message exchange. The key-based authentication mechanism uses a recoverable ECDSA signature to authenticate sender nodes. A pair of asymmetric keys generated on the elliptical curve secp256k1 is assigned to each user. Each sender signs exchanged messages using recoverable ECDSA and allows the receiver to extract the public key from the message signature. The receiver compares the extracted public key to the list of other permissioned node's public key. The sender node is authenticated by the receiver only if one of the entries matches. Otherwise, the receiver rejects the connection. Quorum needs a set number of authenticators and operations in Quorum require a sufficient number of nodes to enter their credentials that assures no single node can make a critical change.

3) Avoid a single point of failure

Blockchain is a specific type of distributed system in which a data object has copies on each peer. The proposed mechanism adopts Quorum with a default Raft consensus algorithm. Raft uses a stronger form of leadership to simplify the management of replicated logs, it defines the following rules to make the distributed system more reliable.

- Log replication: The leader accepts log entries from clients and replicates them on other servers. The leader is also responsible for telling servers when to apply log entries to their state machines. This feature makes data more consistent.
- Leader election: A leader can fail or disconnect from other servers, in which case a new leader needs to be elected. A randomized election timeout is utilized to make

```
=====================

   Deploying 'DNSmart'
   -------------------
   > transaction hash:    0x51f953b331217b0f26230d8afe720a5cea690e96fff5c4b626b907617eccdef9
   > Blocks: 0            Seconds: 0
   > contract address:    0xC41F39d994B9fc0Dc5336D9A52A955d670AeD407
   > block number:        3
   > block timestamp:     1622557929
   > account:             0xf54a07178AeF8F984806bC36fdF6421C53b61744
   > balance:             99.9818569
   > gas used:            672874 (0xa446a)
   > gas price:           20 gwei
   > value sent:          0 ETH
   > total cost:          0.01345748 ETH


   > Saving migration to chain.
   > Saving artifacts
   -------------------------------------
   > Total cost:          0.01345748 ETH


Summary
=======
> Total deployments:   1
> Final cost:          0.01345748 ETH
```

Fig. 7. Demo of deploying a smart contract — DNSmart.

```
truffle(development)> instance.registerDNS(accounts[1], 50, "ntust.block.tw", "192.168.110.42","test transaction")
{ tx:
   '0x65452b804ef54978e62837f3d02c0db42c619acaea379f86a6933128f08b15a7',
  receipt:
   { transactionHash:
      '0x65452b804ef54978e62837f3d02c0db42c619acaea379f86a6933128f08b15a7',
     transactionIndex: 0,
     blockHash:
      '0x505ec30b6f5f2db1acaca00d2c13ec212d23c819825901118f0f67c567f5e686',
     blockNumber: 5,
     from: '0xf54a07178aef8f984806bc36fdf6421c53b61744',
     to: '0xc41f39d994b9fc0dc5336d9a52a955d670aed407',
     gasUsed: 150299,
     cumulativeGasUsed: 150299,
     contractAddress: null,
     logs: [],
     status: true,
     logsBloom:
      '0x00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000',
     rawLogs: [] },
  logs: [] }
```

Fig. 8. Register a domain name with an IP address by sending a transaction.

```
eth_sendTransaction

  Transaction: 0x65452b804ef54978e62837f3d02c0db42c619acaea379f86a6933128f08b15a7
  Gas usage: 150299
  Block Number: 5
  Block Time: Tue Jun 01 2021 22:40:35 GMT+0800 (台北標準時間)

eth_getTransactionReceipt
```

Fig. 9. Verify the transaction to create a block.

```
> instance.getIPaddr(accounts[1],{from:accounts[0]})
'192.168.110.42'
```

Fig. 10. The result of querying IP address.

sure that vote splitting problems are resolved and the service can be available quickly.

4) Avoid DNS Distributed Denial of Service (DDoS)

In Ethereum, you must pay a gas fee in order to make a transaction or execute a smart contract. If the attacker constantly executes the smart contract to send DNS queries, the contract execution will stop immediately because of a gas exhaustion exception. Moreover, the setting of gas limit is the threshold of gas that can be spent on the contract. The gas limit also restricts the computational workload for miners to process a single block on the network. In order to defend against DDoS attacks, the gas limit of a single transaction in the smart contract is set at initialization. It not only authorized users but also restricted their use no more than the gas limit.

TABLE IV
FEATURE COMPARISON OF THE PROPOSED MECHANISM WITH RELATED WORKS – THE PROPOSED MECHANISM PROVIDES BETTER SECURITY IN ACCESS CONTROL AND IDENTITY MANAGEMENT. INFORMATION DISCLOSE, CONSPIRACY AND CACHE POISONING ATTACK CAN BE AVOIDED. STRATIFICATION IMPROVES QUERY PERFORMANCE.

	Information disclose	Access control	Identity management	Conspiracy[1]	Cache Poisoning attack	Stratification[2]
HARD-DNS	No	No	No	Yes	Excellent	No
NConfiDNS	Yes	No	No	Yes	Good	No
CoDNS	Yes	No	No	Yes	Good	No
DNSTMS	No	No	Yes	No	Excellent	No
The proposed	**No**	**Yes**	**Yes**	**No**	**Excellent**	**Yes**

[1] Conspiracy: A network of rogue DNS servers to allow people to register and use domain names
[2] Stratification: A hierarchical architecture. From top to bottom, DNS servers divided into root servers, top level domain servers and authoritative servers.

Fig. 11. Throughput of Raft and IBFT consensuses.

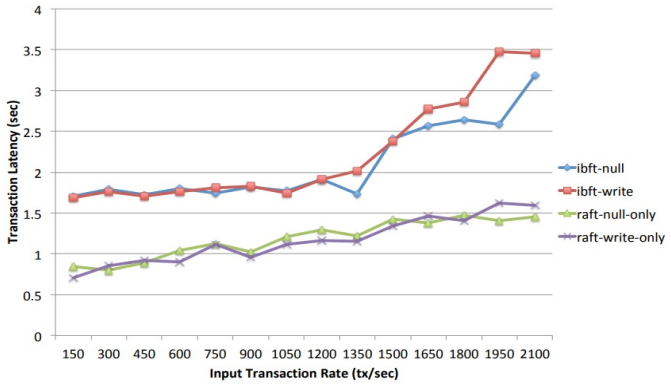

Fig. 12. Latency of Raft and IBFT consensuses.

VI. CONCLUSION

In this paper, after referring to the related works of blockchain-based DNS solutions, we introduce an innovative and secure domain name service based on consortium blockchain. With smart contracts, a trusted DNS registration and resolution process is proposed. The presented mechanism utilizes Quorum with Raft consensus algorithm to obtain a good performance under the premise of ensuring the security of domain name resolution. The comparison of the performance results between Raft and IBFT is presented and an in-depth analysis is given. We believe that the proposed DNS mechanism can effectively resist a variety of attacks.

In the future, we will try to integrate the proposed mechanism with the existing DNS with minimum cost. Moreover, we will design a certificate authority that improves the transaction authentication and remote user authentication in Quorum to provide a more efficient and secure system.

REFERENCES

[1] wandera.com, *"Mobile Threat Landscape Report,"* 2020. [Online]. Available: http://go.wandera.com/rs/988-EGM-040/images/Mobile%20Threat%20Landscape%202020.pdf.
[2] apwg.org, *"Q4 2019 Phishing Activity Trends Report,"* 2021. [Online]. Available: https://docs.apwg.org/reports/apwg_trends_report_q1_2021.pdf.
[3] B. Fonseca, *"VeriSign issues false Microsoft digital certificates,"* ComputerWorld:Security, Mar. 23, 2001. [Online]. Available: https://www.computerworld.com/ article/2798454/verisign-issues-false-microsoft-digital-certificates.html.
[4] C. Osborne, *"Backdoor malware is being spread through fake security certificate alerts,"* ZDNet: Security, Mar. 5, 2020. [Online]. Available: https://www.zdnet.com/article/backdoor-malware-is-being-spread-through-fake-security-certificate-alerts/.
[5] C. Cimpanu, *"Let's Encrypt to revoke 3 million certificates on March 4 due to software bug,"* ZDNet: Security, Mar. 4, 2020. [Online] Available: https://www.zdnet.com/article/lets-encrypt-to-revoke-3-million-certificates-on-march-4-due-to-bug/.
[6] A. Har and T. V. Lakshman, "The Internet blockchain: A distributed, tamper-resistant transaction framework for the Internet," in *Proc. ACM HotNets*, 2016.
[7] B. Benshoof, A. Rosen, A. G. Bourgeois, and R. W. Harriso, "Distributed decentralized domain name service," in *Proc. IEEE IPDPSW*, May 2016.
[8] J. Liu, B. Li, L. Chen, M. Hou, F. Xiang, and P. Wang, "A data storage method based on blockchain for decentralization DNS," in *Proc. IEEE DSC*, Jun. 2018.
[9] Z. Yu, D. Xue, J. Fan and C. Guo, "DNSTSM: DNS cache resources trusted sharing model based on consortium Blockchain," *IEEE Access*, vol. 8, pp. 13640-13650, 2020.
[10] C. Cachin and A. Samar, "Secure distributed DNS," in Proc. *IEEE/ISIP ICDSN*, 2004.
[11] V. Ramasubramanian and E. G. Sirer, "The design and implementation of a next generation name service for the Internet," in *Proc. ACM SIGCOMM*, 2004.
[12] Z. Qiang, Z. Zheng, and Y. Shu, "P2PDNS: A free domain name system based on P2P philosophy," in *Proc. IEEE CCECE*, 2006.
[13] Z. Liu, E. S.-J. Swildens, and R. D. Day, "Domain name resolution using a distributed DNS network," U.S. Patent 7725602 B2, May 25, 2010.
[14] M. Wachs, M. Schanzenbach, and C. Grothoff, "A censorship-resistant, privacy-enhancing and fully decentralized name system," in *Proc. CANS*, 2014.
[15] E. Karaarslan and E. Adiguzel, "Blockchain based DNS and PKI solutions," *IEEE Commun. Standards Mag.*, vol. 2, no. 3, pp. 52–57, Sep. 2018.
[16] S. Gourley and H. Tewari, "Blockchain backed DNSSEC," in *Proc. ICBISIT*, 2019.
[17] Guan, A.Garba, A.Li, Z.Chen, and N.Kaaniche, "AuthLedger: Anovel blockchain-based domain name authentication scheme," in *Proc. ICISSP*, 2019.
[18] Q. Feng, D. He, S. Zeadally, M. K. Khan, and N. Kumar, "A survey on privacy protection in blockchain system," *J. Netw. Comput. Appl.*, vol. 126, pp. 45-–58, Jan. 2019.

[19] GoQuorum, Accessed: Aug. 11, 2021. [Online]. Available: https://docs.goquorum.consensys.net/en/stable/.
[20] Quorum, Accessed: Aug. 19, 2021. [Online]. Available: https://consensys.net/quorum/.
[21] Hyperledger Fabric, Accessed: Sep. 8, 2021. [Online]. Available: https://www.hyperledger.org/.
[22] FISCO BCOS, Accessed: Sep. 24, 2021. [Online]. Available: http://www.fisco-bcos.org/.
[23] B. K. Mohanta, S. S. Panda, and D. Jena, "An Overview of smart contract and use cases in Blockchain technology," in *Proc. IEEE ICCCNT*, 2018.
[24] A, Baliga, I. Subhod, P. Kamat and S. Chatterjee, "Performance evaluation of the quorum Blockchain platform," *arXiv preprint arXiv:1809.03421*, 2018.
[25] D, Ongaro and J. Ousterhout, "In search of an understandable consensus algorithm," in *Proc. USENIX*, 2014.
[26] B. Lashkari and P. Musilek, *'A comprehensive review of Blockchain consensus mechanisms,'* in *IEEE Access*, vol. 9, pp. 43620–43652, 2021.
[27] Hyperldger Caliper Documentation, [online] Available: https://github.com/hyperledger/caliper.
[28] K. Mlitz, *'CIO COVID survey current and future trends in remote work worldwide from 2020 to 2021,'* statista, Oct. 14, 2021. Accessed on: Feb. 4, 2022. [Online]. Available: https://www.statista.com/statistics/1199110/remote-work-trends-covid-survey-september-december/.
[29] J. Liu, B. Li, L. Chen, M. Hou, F. Xiang, and P. Wang, "A data storage method based on blockchain for decentralization DNS," in *Proc. IEEE DSC*, Jun. 2018, pp. 189–196.

Wen-Bin Hsieh received his BS degree in Computer Science and Information Engineering from Tamkang University, Taipei, Taiwan in 2003, and his Ph.D. degree in Electronic Engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 2013. He worked in the Information Department of Landbank from 2006 to 2009, as a Software Engineer. Since 2010, he has worked in the information department of the government, focusing on information security. After work, he is a Standalone Researcher. His research interests include cryptography, communication protocol, network security and mobile communication.

Jenq-Shiou Leu (Senior Member, IEEE) received the BS degree in mathematics and the MS degree in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 1991 and 1993, respectively, and the PhD degree on a part-time basis in computer science from National Tsing Hua University, HsinChu, Taiwan, in 2006. He was with Rising Star Technology, Taiwan, as an R&D Engineer from 1995 to 1997, and worked in the telecommunication industry (Mobitai Communications and Taiwan Mobile) from 1997 to 2007 as an Assistant Manager. In February 2007, he joined the Department of Electronic and Computer Engineering at National Taiwan University of Science and Technology as an Assistant Professor. From February 2011 to January 2014, he was an Associate Professor. Since February 2014, he is a Professor. His research interests include mobile service and platform design and application development of computational intelligence He is a senior member of IEEE.

Jun-Ichi Takada (Senior Member, IEEE) received the Ph.D. degree in electrical and electronic engineering from the Tokyo Institute of Technology (Tokyo Tech), in 1992. After serving as a Research Associate with Chiba University, from 1992 to 1994, and as an Associate Professor with Tokyo Tech, from 1994 to 2006, he has been a Professor with Tokyo Tech, since 2006. From 2003 to 2007, he was a Researcher with the National Institute of Information and Communication Technology (NICT), Japan. His current research interests include radiowave propagation and channel modeling for mobile and short range wireless systems, applied measurement using radio waves, and ICT applications for international development. He is a fellow of the Institute of Electronics, Information and Communication Engineering (IEICE), Japan, and a member of the Japan Society for International Development.