A nameserver receives a request to access a webpage from a client device. The nameserver generates a blockchain query based on the request for querying a domain registrar blockchain storing domain name registration data in smart contracts. The nameserver sends the first blockchain query to the domain registrar blockchain. In response to receiving a nameserver identifier from the domain registrar blockchain, the nameserver retrieves a DNS record for the webpage using the received nameserver identifier. The nameserver then provides information from the retrieved DNS record to the client device to allow the client device to access the webpage.
(2.1) REQUEST FOR WEBPAGE

(2.2) TRANSLATE NAMESERVER LOOKUP REQUEST FOR WEBSITE TO A BLOCKCHAIN QUERY

(2.3) BLOCKCHAIN QUERY WITH NAMESERVER LOOKUP REQUEST

(2.4) ACCESS SMART CONTRACTS TO LOCATE NAMESERVER INFORMATION

(2.5) NAMESERVER LOOKUP RESPONSE

(2.6) REQUEST DNS RECORD FROM MIDDLEWARE FOR NAMESERVER

(2.7) GENERATE DNS RECORD LOOKUP REQUEST AS A BLOCKCHAIN QUERY

(2.8) RESOLVE DNS RECORD REQUEST

(2.9) ACCESS SMART CONTRACTS TO LOCATE DNS RECORD

(2.10) DNS RECORD

(2.11) REQUEST SENT TO WEB SERVER WITH IP ADDRESS FROM DNS RECORD

(2.12) RETRIEVED WEBPAGE TO USER DEVICE

FIG. 2
FIG. 4
FIG. 6

BLOCKCHAIN-BASED DOMAIN NAME REGISTRAR AND MANAGEMENT SYSTEM 100

(6.1) PERFORM REGISTRATION RENEWAL REQUIREMENTS

(6.2) IDENTIFY RENEWAL EVENT

(6.3) BLOCKCHAIN TRANSACTION TO BLOCKCHAIN

(6.4) ACCESS SMART CONTRACTS & MINT NEW DOMAIN TO USER'S DIGITAL WALLET

DOMAIN REGISTRAR BLOCKCHAIN 106

REGISTRAR MIDDLEWARE 104

CLIENT DEVICE 602

NAMESERVER SMART CONTRACTS 108
FIG. 9
RECEIVING, BY A NAMESERVER, A FIRST REQUEST FROM A CLIENT DEVICE, THE FIRST REQUEST TO ACCESS A WEBPAGE 1010

GENERATING A FIRST BLOCKCHAIN QUERY BASED ON THE FIRST REQUEST, THE FIRST BLOCKCHAIN QUERY FOR QUERYING A DOMAIN REGISTRAR BLOCKCHAIN 1015

SENDING THE FIRST BLOCKCHAIN QUERY TO THE DOMAIN REGISTRAR BLOCKCHAIN, THE DOMAIN REGISTRAR BLOCKCHAIN STORING DOMAIN NAME REGISTRATION DATA IN SMART CONTRACTS IN THE DOMAIN REGISTRAR BLOCKCHAIN 1020

RECEIVING A NAMESERVER IDENTIFIER FROM THE DOMAIN REGISTRAR BLOCKCHAIN 1025

RETRIEVING A DNS RECORD FOR THE WEBPAGE USING THE RECEIVED NAMESERVER IDENTIFIER 1030

PROVIDING INFORMATION FROM THE RETRIEVED DNS RECORD TO THE CLIENT DEVICE FOR ACCESSING THE WEBPAGE 1035

FIG. 10
BLOCKCHAIN-BASED DOMAIN NAME REGISTRAR AND MANAGEMENT SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application No. 63/408,826, filed Sep. 21, 2022, which is hereby incorporated by reference.

FIELD

[0002] Embodiments of the invention relate to the field of data storage using blockchains; and more specifically, to a blockchain-based domain name registrar.

BACKGROUND

[0003] The domain name system is a naming system for computing devices and resource (e.g., webpages) on the internet. The domain name system utilizes Internet Protocol (IP) addresses that are assigned to computing devices and resources that allow them to be identifiable and reachable for communications, exchanging data, etc. One important aspect of the domain name system is the storing of domain names and their corresponding IP addresses. This allows users to access a desired webpage using a web address (e.g., URL), rather than requiring users to memorize the specific numerical values of the IP address corresponding to the desired webpage. When a domain name is registered with a domain name registrar, the registrar can provide domain name resolution services for the domain name. Thus, when a user requests a webpage (e.g., by providing a URL in a browser application), a domain name resolution process is performed to identify the corresponding IP address and return the address to the browser application. The domain name resolution process can include multiple recursive steps that query nameserver databases to locate the DNS record that includes the IP address.

SUMMARY

[0004] Introduced here are techniques/technologies that allow a blockchain-based domain name registrar and management system to convert domain names and their corresponding DNS records into on-chain assets that are stored and managed directly on a blockchain. The blockchain-based registrar system uses smart contracts, on the Ethereum network, to support the features and functions required by the Internet Corporation for Assigned Names and Numbers (ICANN) to be an accredited domain registrar. More specifically, the blockchain-based registrar system supports full feature parity with traditional web2 registrars, allowing for new domain names to be registered and existing domain names to be transferred and renewed.

[0005] A nameserver of blockchain-based domain name registrar and management system receives a first request from a client device to access a webpage. The nameserver generates a blockchain query based on the request for querying a domain registrar blockchain. The domain registrar blockchain stores domain name registration data in smart contracts. The nameserver sends the blockchain query to the domain registrar blockchain. The domain registrar blockchain checks the smart contracts to determine whether the requested domain name is stored on the domain registrar blockchain and retrieves a nameserver identifier. The nameserver identifier is used to retrieve a DNS record for the webpage and information from the retrieved DNS record (e.g., an IP address) is provided to the client device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The invention may best be understood by referring to the following description and accompanying drawings that are used to illustrate embodiments of the invention. In the drawings:

[0007] FIG. 1 illustrates a blockchain-based domain name registrar and management system according to some embodiments described herein;

[0008] FIG. 2 illustrates an exemplary networked system for resolving a domain name using a blockchain-based domain registrar according to some embodiments described herein;

[0009] FIG. 3 illustrates an exemplary networked system for transmitting an email message using a blockchain-based domain registrar according to some embodiments described herein;

[0010] FIG. 4 illustrates an exemplary networked system for performing a registration process for a new domain name using a blockchain-based domain registrar according to some embodiments described herein;

[0011] FIG. 5 illustrates an exemplary networked system for performing an off-chain renewal process for a domain name using a blockchain-based domain registrar according to some embodiments described herein;

[0012] FIG. 6 illustrates an exemplary networked system for performing an on-chain renewal process for a domain name using a blockchain-based domain registrar according to some embodiments described herein;

[0013] FIG. 7 illustrates an exemplary networked system for performing a process of transferring a domain name from a blockchain-based domain registrar to a non-blockchain registrar according to some embodiments described herein;

[0014] FIG. 8 illustrates an exemplary networked system for performing a process of transferring a domain name from a non-blockchain registrar to a blockchain-based domain registrar according to some embodiments described herein;

[0015] FIG. 9 illustrates an exemplary networked system for setting a DNS record for a domain name with a blockchain-based domain registrar according to some embodiments described herein;

[0016] FIG. 10 is a flow diagram that illustrates exemplary operations for a blockchain-based domain name registrar according to an embodiment; and

[0017] FIG. 11 is a block diagram for an exemplary data processing system that can be used in some embodiments.

DESCRIPTION OF EMBODIMENTS

[0018] Embodiments of the present disclosure include a blockchain-based domain name registrar and management system. The blockchain serves as a key infrastructure layer for performing distributed computation across a trustless network. Using smart contracts, any set of parties can perform trustless interactions with one another off a predefined set of rules. Actions that occur off-chain are fully auditable, immutable, and cryptographically administered. Blockchains are decentralized systems that allow for heightened security and direct ownership/administration of assets. The blockchain-based domain name registrar and management system can perform operations and processes using
blockchain queries and transactions, including registering, renewing, and transferring domain names, and resolving DNS record requests.

[0019] Traditional domain name registrars manage the registration of domain names by maintaining domain name registration information in centralized databases. Typical disadvantages of using centralized databases include introducing a single point of failure. In addition, centralized databases typically have over privileged administrators, inability to prove and verify the integrity of data, and a lack of auditable trails.

[0020] To address the deficiencies in conventional systems, the blockchain-based domain name registrar and management system of the present disclosure stores domain name registration information and DNS records in a blockchain. Since domain names and their corresponding DNS records are all stored on-chain, the blockchain-based domain name registrar uses “wrappper” middleware services to translate and convert traditional nameserver requests into blockchain queries.

[0021] Embodiments describe herein provide many technical advantages and improvements, in addition to addressing the deficiencies of previous solutions. By adhering to ICANN requirements, the blockchain-based domain name registrar and management system uses the security and finality guarantees of blockchain transactions to administer and facilitate the management of domain names, while maintaining their interoperability with traditional systems. For example, using serverless middleware services to translate and convert traditional nameserver requests into blockchain queries abstracts away the complexities of interfacing with smart contracts and provides an interface structured as a standard nameserver. This allows the blockchain-based domain name registrar to be queried using requests structured as if they will be processed by traditional database-based domain name registrars.

[0022] Further, a blockchain-based domain name registrar and management system can provide the benefits of using a blockchain to store DNS records and resolve requests. For example, the decentralized approach to registering and resolving domain names and DNS records using the blockchain-based domain name registrar and management system provides a fully transparent, auditable, and cryptographically secure domain name registration system. A blockchain-based domain name registrar further allows current DNS records, historical versions, and administrative actions to be publicly accessible. Unlike traditional systems, resolving DNS records for on-chain domains does not require access to any centralized services provided by the administering registrar, but can instead be resolved directly by the user themselves. Transacting on top of domains does not require a third party but can be performed directly on-chain due to the composable nature of the assets. The blockchain-based registrar further increases the composable and interoperability of the domain name by issuing it on-chain.

[0023] Further, a blockchain-based domain name can have the benefits of real-time transfers and settlement guarantees. For example, because the registration of a domain name in the domain registrar blockchain is tied to an on-chain asset (e.g., a token or NFT) stored in a digital wallet of the owner of the domain, the domain name can be transferred to another person by transferring the on-chain asset from the digital wallet of the owner to a new owner’s digital wallet. Once the on-chain asset is in the new owner’s digital wallet, only the new owner has administrative capabilities for the domain.

[0024] FIG. 1 illustrates a blockchain-based domain name registrar and management system according to some embodiments described herein. The blockchain-based domain name registrar and management system 100 illustrated in FIG. 1 includes a nameserver 102, a blockchain that operates a domain registrar blockchain 106, and a blockchain that operates a DNS resolver blockchain 112. The domain registrar blockchain 106 is a blockchain that operates a domain registrar and the DNS resolver blockchain 112 is a blockchain that operates a DNS resolver.

[0025] In some embodiments, the nameserver 102 is an authoritative name server configured and authorized to receive requests for domain names and transmit IP addresses in response. In the embodiment of FIG. 1, the nameserver 102 includes registrar middleware 104 and DNS middleware 110 as separate modules/components. In other embodiments, the registrar middleware 104 and the DNS middleware 110 are a single module/component of nameserver 102.

[0026] In one embodiment, registrar middleware 104 is an intermediary configured to receive requests from client devices. In embodiments, the registrar middleware 104 is further configured to translate received requests into blockchain queries for querying the domain registrar blockchain 106. Example requests received by the registrar middleware 104 can include requests to access/retrieve a webpage, send an email, register a new domain name, renew a domain name, transfer a domain name off or onto the domain registrar blockchain 106, etc. In one embodiment, the DNS middleware 110 is an intermediary configured to receive and translate DNS requests into blockchain queries for querying the DNS resolver blockchain 112 and returning IP addresses from DNS records. In embodiments, the purpose of the registrar middleware 104 and the DNS middleware 110 is to maintain the link between the information for a domain name stored off-chain and the information for the domain name in the domain registrar blockchain 106.

[0027] In embodiments, the registrar middleware 104 and DNS middleware 110 allow the blockchain-based domain name registrar and management system 100 to store domain names and DNS records to blockchains (e.g., domain registrar blockchain 106 and DNS resolver blockchain 112) using on-chain assets (e.g., tokens or NFTs). For example, while the domain name registration process involves off-chain operations (e.g., process with ICANN, registering the nameserver with a top-level registry, etc.), when a domain name is registered to the domain registrar blockchain 106, an on-chain asset can be minted and added to a digital wallet of the user who registered the domain name.

[0028] The on-chain asset functions as the controller/master of the domain name for the blockchain-based domain name registrar and management system 100. In embodiments, as the on-set asset is stored in the holder’s digital wallet, possession of the on-chain asset grants the holder of the on-chain asset exclusive administrative control/permission over the domain name corresponding to the on-chain asset. For example, any changes to a DNS record for a domain names stored in the DNS resolver blockchain 112 can only be performed by the owner of the on-chain asset associated with the domain name. To update a DNS record in the DNS resolver blockchain 112, a request can include the owner’s digital wallet address to indicate that the request
is from the owner of the on-chain asset. In contrast, traditional database-based domain registrars have control over domain names stored in their centralized databases, such that while a user can contact a traditional domain registrar to reset credentials, modify DNS records, etc., the traditional domain registrar has ultimate control over whether the perform the request action(s). For example, a request to change a DNS record can be sent to the traditional domain registrar, the traditional domain registrar determines whether to fulfill the request.

[0029] In one embodiment, the on-chain asset can be transferred from one user to another user (e.g., via on-chain transactions between digital wallets). Once transferred, the new owner of the on-chain asset becomes the controller/master of the domain name and is the only identity that can make changes to the DNS record for the domain name associated with the on-chain asset.

[0030] In one embodiment, when a request for a domain name is received, the request is sent to a registry to identify the owner/nameserver that owns the domain name. Where the owner/nameserver stores the domain names in a domain registrar blockchain 106 and the DNS records in a DNS resolver blockchain 112, the owner/nameserver is configured to read the appropriate blockchain and return whatever data is stored in the blockchain.

[0031] In one embodiment, the domain registrar blockchain 106, or blockchain-based domain registrar, is a blockchain database, or distributed ledger, storing domain name registration data. The domain registrar blockchain 106 can use nameserver smart contracts 108 to build a registrar to store domain names. In embodiments, the nameserver smart contracts 108 can include one or more smart contracts. Smart contracts are state based logical units that operate on-chain. They are programmable agreements that can be used to build decentralized protocols on the domain registrar blockchain 106. In some embodiments, the nameserver smart contracts 108 are used to mint and register new domain names on-chain in the domain registrar blockchain 106. For the domain registrar blockchain 106, the nameserver smart contracts 108 store which domain names are stored on-chain.

[0032] In one embodiment, the DNS resolver blockchain 112 is a blockchain database, or distributed ledger, storing DNS records. The DNS resolver blockchain 112 can use blockchain resolver smart contracts 114 to build a registrar to store and manage DNS records for corresponding domain names. In embodiments, the resolver smart contracts 114 can include one or more smart contracts. Smart contracts are programmable agreements that can be used to build decentralized protocols on the DNS resolver blockchain 112. In some embodiments, the blockchain resolver smart contracts 114 are used to store and resolve DNS records associated with a specific domain name. For the DNS resolver blockchain 112, the blockchain resolver smart contracts 114 store the different DNS records for each domain name.

[0033] FIG. 2 illustrates an exemplary networked system for resolving a domain name using a blockchain-based domain registrar according to an embodiment described herein. The exemplary networked system 200 illustrated in FIG. 2 includes blockchain-based domain name registrar and management system 100, a client device 202, and a web server 204. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104, a domain registrar blockchain 106, DNS middleware 110, and a DNS resolver blockchain 112.

[0034] Examples of client device 202 include computing devices (e.g., laptops, workstations, smartphones, palmtops, mobile phones, tablets, gaming systems, set top boxes, wearable devices, electronic devices, etc.) that are capable of transmitting and/or receiving network traffic. In some embodiments, client device 202 executes a client network application that is capable of transmitting and/or receiving network traffic. For example, client network applications may be a web browser or other application that can access network resources (e.g., web pages).

[0035] In one embodiment, the web server 204 is a device that receives requests (e.g., HTTP/S requests) for resources, such as a webpage. For example, the web server 204 can receive a request for a resource (e.g., a webpage) that includes an address (e.g., IP address) for the resource and either return the requested resource or an error message.

[0036] In one embodiment, a request for a webpage is received at the registrar middleware 104 from a client device 202, at operation 2.1. The request for the webpage can be generated in response to a user selecting a link or URL for the webpage (e.g., in a browser application or client network application). For example, the request can be for the webpage “http://example.com.” In embodiments, to determine the nameserver that owns or manages the requested webpage, a forward lookup process is performed by first sending the request to a local DNS server. If the address of the nameserver for the webpage is not stored in a cache of the local DNS server, the local DNS server performs a recursive query, or recursive lookup, to other DNS servers determine the identity (e.g., IP address) of the nameserver for the webpage. For example, where the request is for “http://example.com,” the local DNS server may first forward the request to a local DNS root server, which returns the address for the “.com” top level DNS server. The local DNS server then forwards the request to the “.com” top level DNS server, which returns the address for the “example.com” nameserver, where the “example.com” nameserver stores DNS records on-chain. The local DNS server then forwards the request to the “example.com” nameserver, on which the registrar middleware 104 is executing.

[0037] In one embodiment, the registrar middleware 104 translates the request for the webpage to a blockchain query, at operation 2.2. For example, the registrar middleware 104 translates the request for the webpage to a blockchain query by decoding the DNS request and performing a recursive query on the associated DNS records. The blockchain query is then sent to the domain registrar blockchain 106 to perform the nameserver lookup request, at operation 2.3. Common DNS requests can include a QName (or target zone) and a QNameType (or requested record type). Each recursive query performs a series of record lookups by asking the smart contract for the records for X QName and X QNameType. The blockchain query is then sent to the domain registrar blockchain 106 to perform the nameserver lookup request, at operation 2.3.

[0038] In one embodiment, the nameserver smart contracts 108 are accessed to locate nameserver information, at operation 2.4. In one embodiments, when the blockchain query generated by the registrar middleware 104 hits the domain registrar blockchain 106, the domain registrar blockchain 106 checks the nameserver smart contracts 108 to determine whether the requested domain name is stored on
the domain registrar blockchain 106. In one embodiment, the blockchain query is sent to a public remote procedure code (RPC) node, in which the node carries out the RPC request and returns the corresponding data. In other embodiments, an RPC node is run locally and in sync with the blockchain network. The blockchain query is then sent to the local RPC node, which carries out the RPC request and returns the data. In both cases, the blockchain queries are queries (e.g., non-transactional, interactions with the node and network), where they only read data from the blockchain. As they are used to lookup DNS records, they do not modify the state.

[0039] In one embodiment, the registrar middleware 104 performs a series of recursive queries to determine the zone associated with the DNS request. Because all zones are treated as their own registration, the registrar middleware 104 performs a recursive query from left to right to determine the parent zone that the request corresponds to. For example, a request for “a.b.example.com” could have three possible zones (e.g., “a.b.example.com,” “b.example.com,” and “example.com”). However, each zone may be owned by a different user/entity. For example, a first user/entity owns “example.com” and mints a subdomain, “b.example.com” as an on-chain asset (e.g., tokens or NFTs) to a second user/entity’s digital wallet. The registration link between the domain and the zone is stored in the domain registrar blockchain 106. In such situations, the DNS record for “b.example.com” can be set in the context of the first user’s zone in the resolver blockchain, but the second user’s zone records take precedence. The first user’s DNS records overlapping with the second user’s zone may only be referenced if the second user’s zone is burned. Thus, if a blockchain query is received for “a.b.example.com,” the second user’s zone (e.g., “x.example.com”) is referenced instead of the first user’s zone (e.g., “example.com”).

[0040] Once the registrar middleware 104 identifies the parent zone from the domain registrar blockchain 106, that zone (or node) is then used to query the corresponding DNS records.

[0041] When the requested domain name is determined to be stored on the domain registrar blockchain 106, a nameserver lookup response is returned to the registrar middleware 104, at operation 2.5. The nameserver lookup response can include the nameserver (e.g., an address for the nameserver) for the requested domain name. For example, the nameserver for “example.com” is provided. The nameserver for the requested domain name is then sent to the DNS middleware 110, at operation 2.6.

[0042] In one embodiment, the DNS middleware 110 then generates a DNS record lookup request as a blockchain query using the nameserver for the requested domain name, at operation 2.7. The DNS middleware 110 then resolves the DNS record request by sending the blockchain query to the DNS resolver blockchain, at operation 2.8. In embodiments, the blockchain query is specific to the configuration of the blockchain resolver smart contracts 114 and the schema of the data stored in the blockchain resolver smart contracts 114. In one embodiment, the blockchain resolver smart contracts 114 include a function, “DNSRecord,” that takes in a root zone, a name (or query name), and a resource (e.g., indicating the requested DNS record type). Using the “http://page.example.com” example, the root zone is “example.com,” the name is “page.example.com,” and the resource is an “A record.”

[0043] In one embodiment, the blockchain resolver smart contracts 114 are accessed to locate the DNS record for the requested webpage, at operation 2.9. If the requested domain name is stored on the DNS resolver blockchain 112, a DNS record lookup is performed to determine if the requested domain name delegates control elsewhere. For example, a delegate DNS record type (e.g., a DS record) can be used to point authoritative control to a different IP address or canonical name (CNAME). If the requested domain name does delegate control, the request is forwarded. If the requested domain name does not delegate control, DNS resolver blockchain 112 checks to see if the corresponding record exists. If the record exists, the data is returned, including the nameserver(s) upon which to locate the DNS records. If the record does not exist, a “no data” response is returned. If the requested domain name is not stored on the DNS resolver blockchain 112, the DNS request is denied. For example, continuing the example, the root server for “http://page.example.com” can be queried, and the server that controls “.com” is returned. The “.com” server is then queried for “example.com,” and the server that controls “example.com” is returned. The “example.com” server is then queried for “page.example.com,” and the “page.example.com” record can be returned. In some embodiments, if the “example.com” server has delegated control of the subdomain to another server, the querying process continues until the server that has the record is identified.

[0044] In some embodiments, the data in the blockchain resolver smart contracts 114 is tracked and then stored in an off-chain database. In such embodiments, a bot can be used to monitor new transactions and mirror any modifications made to DNS records on-chain, in the off-chain database. To perform a lookup where the data is stored in an off-chain database, the off-chain database can be simply queried.

[0045] In one embodiment, after locating the DNS record, the DNS record or an IP address determined from the DNS record, is returned, at operation 2.10. In one embodiment, after receiving the DNS record, the DNS middleware 110 directs the request for the webpage to a web server using the IP address indicated by the DNS record, at operation 2.11. In other embodiments, the DNS middleware 110 returns the IP address indicated by the DNS record to the client device 202 directly, and the client device 202 sends the request for the webpage using the IP address to the web server 204.

[0046] In one embodiment, the retrieved webpage is transmitted to the client device 202, at operation 2.12.

[0047] FIG. 3 illustrates an exemplary networked system for transmitting an email message using a blockchain-based domain registrar according to some embodiments described herein. The exemplary networked system 300 illustrated in FIG. 3 includes blockchain-based domain name registrar and management system 100, a client device 302, a mail server 308 and a domain owner server 310. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104, a domain registrar blockchain 106, DNS middleware 110, and an MX records blockchain 304.

[0048] In one embodiment, the MX records blockchain 304 is a blockchain database, or distributed ledger, storing mail exchanger (MX) records. An MX record is a record indicating how an email message should be routed to arrive at a mail server. The MX records blockchain 304 can use MX records smart contracts 306 to build a registrar to store and managed MX records for corresponding domain names.
Smart contracts are programmable agreements that can be used to build decentralized protocols on the MX records blockchain 304. In some embodiments, the MX records smart contracts 306 are used to store and resolve MX records associated with a specific domain name. For the MX records blockchain 304, the MX records smart contracts 306 store the different MX records for each domain name.

[0049] In one embodiment, the mail server 308 is a device configured to receive and send email messages between computing devices. For example, the mail server 308 can receive an email from a sending computing device (e.g., client device 302) to a recipient computing device (e.g., via a domain owner server 310).

[0050] In one embodiment, a client device 302 sends an email message directed to a recipient at a domain that is received at the registrar middleware 104, at operation 3.1. For example, the email message can be directed to the email address: “mail@example.com,” where “example.com” is the domain of the recipient. In embodiments, the email message can be sent using a client application (e.g., an email application). In embodiments, to determine the nameserver that owns or manages the requested email domain, a forward lookup process is performed, as described with respect to FIG. 2.

[0051] In one embodiment, the registrar middleware 104 translates a nameserver lookup request for the domain to a blockchain query, at operation 3.2. In some embodiments, the registrar middleware 104 translates the request for the webpage to a blockchain query, as described with respect to FIG. 2. The blockchain query to perform the nameserver lookup request is then sent to the domain registrar blockchain 106, at operation 3.3.

[0052] In one embodiment, the nameserver smart contracts 108 are accessed to locate nameserver information, at operation 3.4. In one embodiment, when the blockchain query generated by the registrar middleware 104 hits the domain registrar blockchain 106, the domain registrar blockchain 106 checks the nameserver smart contracts 108 to determine whether the requested domain name (e.g., “example.com”) is stored on the domain registrar blockchain 106.

[0053] When the requested domain name is determined to be stored on the domain registrar blockchain 106, a nameserver lookup response is returned to the registrar middleware 104, at operation 3.5. The nameserver lookup response can include the nameserver (e.g., an address for the nameserver) for the requested domain name. For example, the nameserver for “example.com” is provided. The nameserver for the requested domain name is then sent to the DNS middleware 110, at operation 3.6.

[0054] In one embodiment, the DNS middleware 110 then generates an MX record lookup request as a blockchain query using the nameserver for the requested domain name, at operation 3.7. The DNS middleware 110 then resolves the MX record request by sending the blockchain query to the MX records blockchain 304, at operation 3.8.

[0055] In one embodiment, the MX records smart contracts 306 are accessed to locate the MX record for the requested domain, at operation 3.9. If the requested domain name is stored on the MX records blockchain 304, a MX record lookup is performed to determine if the requested domain name delegates control elsewhere. If the requested domain name does delegate control, the request is forwarded. If the requested domain name does not delegate control, MX records blockchain 304 checks to see if the corresponding record exists. If the record exists, the data is returned, including the nameserver(s) upon which to locate the MX records. If the record does not exist, a “no data” response is returned. If the requested domain name is not stored on the MX records blockchain 304, the MX request is denied.

[0056] In one embodiment, after locating the MX record, the MX record or an IP address determined from the MX record, is returned, at operation 3.10. In one embodiment, after receiving the MX record, the DNS middleware 110 directs the email message to a mail server 308 using the IP address indicated by the MX record, at operation 3.11. In one embodiment, the mail server 308 then transmits the email message to the domain owner server 310 for delivery to a recipient client device, at operation 3.12.

[0057] FIG. 4 illustrates an exemplary networked system for performing a registration process for a new domain name using a blockchain-based domain registrar according to some embodiments described herein. The exemplary networked system 400 illustrated in FIG. 4 includes blockchain-based domain name registrar and management system 100 and a client device 402. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104 and a domain registrar blockchain 106.

[0058] In one embodiment, the client device 402 sends a registration request to register a new domain for a user, at operation 4.1. For example, the client device 402 can send a registration request to register “newexample.com.” The registration request is received by the registrar middleware 104. In embodiments, to determine the nameserver that owns or manages the new domain, a forward lookup process is performed, as described with respect to FIG. 2.

[0059] In one embodiment, the registrar middleware 104 verifies the availability of the new domain and reserves the new domain for the user. In some embodiments, the registrar middleware 104 first verifies that the new domain name is unique and performs an off-chain registration prior to issuing the domain name on-chain. For example, the registrar middleware 104 performs a verification process with the Internet Corporation for Assigned Names and Numbers (ICANN) for the new domain name. This can include providing identifying contact information for the registrant, among others.

[0060] In some embodiments, instead of performing an off-chain registration, all domains can be stored on the domain registrar blockchain 106, as a top-level domain. In such embodiments, the registrar middleware 104 does not interface with any other off-chain systems to verify if a new domain name is available, as the registrar middleware 104 can just check the domain registrar blockchain 106.

[0061] In one embodiment, after the registrar middleware 104 reserves the new domain, the registrar middleware 104 sends a blockchain transaction to mint the new domain name to the domain registrar blockchain 106, at operation 4.3. The blockchain transaction may include an authorization signature which is used by the registrar smart contract to authorize the mint. Funds may be collected on-chain or off-chain. The nameserver smart contracts 108 are accessed and the new domain name is minted as a new on-chain asset to the user’s digital wallet, at operation 4.4. For example, the on-chain asset is assigned to the address of the user’s digital wallet. In some embodiments, this includes storing a mapping in a specific smart contract that maps the new domain name to
the user’s wallet address. The mapping stores all domain names that are registered on the domain registrar blockchain 106 and their owners. This allows for lookups for registered domains, and if registered, to whom they are registered.

[0062] In some embodiments, a notification message can be transmitted back to the client device 402 confirming the registration of the new domain name. In some embodiments, the newly registered domain can be returned to the client device 402 as an NFT.

[0063] FIG. 5 illustrates an exemplary networked system for performing an off-chain renewal process for a domain name using a blockchain-based domain registrar according to some embodiments described herein. The exemplary networked system 500 illustrated in FIG. 5 includes blockchain-based domain name registrar and management system 100 and a client device 502. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104 and a domain registrar blockchain 106.

[0064] In one embodiment, the client device 502 sends a renewal request for a domain, at operation 5.1. For example, the client device 502 can send a renewal request to be performed off-chain for the renewal of the domain “example.com.” The renewal request is received by the registrar middleware 104. In embodiments, to determine the nameserver that owns or manages the domain, a forward lookup process is performed, as described with respect to FIG. 2.

[0065] In one embodiment, the registrar middleware 104 updates the ICANN registration for the domain name off-chain. In one embodiment, after the registrar middleware 104 updates the ICANN registration for the domain name to reflect the renewal, the registrar middleware 104 sends a blockchain transaction to update the domain expiration date to the domain registrar blockchain 106, at operation 5.3. The nameserver smart contracts 108 are accessed and the registration expiration date for the domain name is updated to the user’s wallet address, at operation 5.4. For example, an expiration metadata variable associated with the existing on-chain asset in the domain registrar blockchain 106 is updated.

[0066] In some embodiments, a notification message can be transmitted back to the client device 502 confirming the renewal of the domain name, including the updated expiration date of the domain.
a non-blockchain registrar to a blockchain-based domain registrar according to some embodiments described herein. The exemplary networked system 800 illustrated in FIG. 8 includes blockchain-based domain name registrar and management system 100, a client device 802, and a registrar database 804. The registrar database 804 can be a database storing domain names for a domain name registrar. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104 and a domain registrar blockchain 106.

[0076] In one embodiment, the client device 802 requests to transfer a domain name on a non-blockchain-based registrar (e.g., registrar database 804) to a domain registrar blockchain (e.g., domain registrar blockchain 106), at operation 8.1. In such embodiments, the client device 802 can further provide a wallet address to the registrar database 804 for transferring the domain name to the domain registrar blockchain 106.

[0077] In one embodiment, the registrar database 804 sends a request to transfer the domain to the domain registrar blockchain 106, at operation 8.3. In embodiments, to determine the nameserver that owns or manages the domain, a forward lookup process is performed, as described with respect to FIG. 2. The registration request is received by the registrar middleware 104 and the transfer event is identified, at operation 8.4. The emitting of the transfer event indicates that a transfer transaction has been successfully executed to transfer the domain name from the domain registrar blockchain to a non-blockchain registrar. This allows the transfer event to be a transparent and auditable event. Further, the emitting of the transfer event indicates that the transfer should be replicated in off-chain. In one embodiment, the registrar middleware 104 sends a blockchain transaction to transfer the domain to the domain registrar blockchain 106, at operation 8.3. The domain registrar blockchain 106 mints the domain name to the wallet address received from the client device 802, at operation 8.6.

[0078] In some embodiments, a notification message can be transmitted back to the client device 802 confirming the transfer of the domain from to the domain registrar blockchain 106.

[0079] FIG. 9 illustrates an exemplary networked system for setting a DNS record for a domain name with a blockchain-based domain registrar according to some embodiments described herein. The exemplary networked system 900 illustrated in FIG. 9 includes blockchain-based domain name registrar and management system 100 and a client device 902. The blockchain-based domain name registrar and management system 100 includes registrar middleware 104 and a domain registrar blockchain 106.

[0080] In one embodiment, the client device 902 sends a request to set a DNS record associated with a domain name stored on the domain registrar blockchain 106, at operation 9.1. The request can include, at least, information identifying the domain name and an address for the DNS record (e.g., a new or updated IP address). In some embodiments, the request can be to change or update a DNS record. In embodiments, to determine the nameserver that owns or manages the domain, a forward lookup process is performed, as described with respect to FIG. 2. The request is received by the registrar middleware 104 and the request is translated to a blockchain transaction, at operation 9.2. In one embodiment, the registrar middleware 104 sends the blockchain transaction to set the DNS record for the domain to the domain registrar blockchain 106, at operation 9.3. The domain registrar blockchain 106 updates the DNS records for the domain name, at operation 9.4.

[0081] FIG. 10 is a flow diagram that illustrates exemplary operations for a blockchain-based domain name registrar according to an embodiment. The operations of FIG. 10 will be described with reference to the exemplary embodiment of the other figures. However, it should be understood that the operations of FIG. 10 can be performed by embodiments of the invention other than those discussed with reference to the other figures, and the embodiments discussed with reference to the other figures can perform operations different than those discussed with reference to FIG. 10. The operations of FIG. 10 are described as being performed by the nameserver 102 of a blockchain-based domain name registrar and management system 100.

[0082] At operation 1010, a nameserver receives a first request from a client device, the first request to access a webpage. In some embodiments, the request for the webpage can be generated in response to a user selecting a link or URL for the webpage (e.g., in a browser application or client network application). In embodiments, to determine the nameserver that owns or manages the requested webpage, a recursive forward lookup process is performed to identify the nameserver that is the authoritative nameserver for the requested webpage.

[0083] At operation 1015, the nameserver generates a first blockchain query based on the first request, the first blockchain query for querying a domain registrar blockchain. In one embodiment, the nameserver includes middleware services (e.g., register middleware) that is configured to translate request messages from a first request type to a second request type. For example, the register middleware can translate a nameserver lookup request structured for identifying nameserver (NS) records from a centralized database into the first blockchain query structured for querying data in a blockchain.

[0084] At operation 1020, the nameserver sends the first blockchain query to the domain registrar blockchain, the domain registrar blockchain storing domain name registration data in smart contracts in the domain registrar blockchain.

[0085] At operation 1025, the nameserver receives a nameserver identifier from the domain registrar blockchain. In some embodiments, when the blockchain query generated by the registrar middleware hits the domain registrar blockchain, the domain registrar blockchain checks the nameserver smart contracts to determine whether the requested domain name is stored on the domain registrar blockchain. When the requested domain name is determined to be stored on the blockchain, a nameserver lookup response is returned to the registrar middleware. The nameserver lookup response can include a nameserver identifier (e.g., an address for the nameserver) for the requested domain name. For example, the nameserver for “example.com” is provided. The nameserver for the requested domain name is then sent to the DNS middleware 110, at operation 2.6.

[0086] At operation 1030, the nameserver retrieves a DNS record for the webpage using the received nameserver identifier. After receiving the nameserver identifier from the domain registrar blockchain, the nameserver identifier for the requested domain name can then be sent to middleware (e.g., DNS middleware). In some embodiments, the registrar
middleware and the DNS middleware can be a single component of the nameserver. In one embodiment, the DNS middleware generates a DNS record lookup request as a second blockchain query using the nameserver identifier for the requested domain name. In some embodiments, a DNS resolver blockchain storing DNS records in smart contracts is designated by the domain registrar blockchain. For example, the blockchain-based domain name registrar can set the DNS resolver blockchain as a default smart contract controlled by the blockchain-based domain name registrar. In embodiments, when the blockchain-based domain name registrar updates the smart contract for subsequent registrations, users with existing registrations can either migrate to the new smart contract or keep the existing registration. In one embodiment, migration to the new smart contract includes changing the address of the resolver smart contract corresponding to a zone in the registrar smart contract. In some embodiments, migration can further include setting the same DNS records in the new resolver smart contract. The DNS middleware then resolves the DNS record request by sending the second blockchain query to the DNS resolver blockchain. In one embodiment, blockchain resolver smart contracts are accessed to locate the DNS record for the requested webpage. If the record exists, the DNS record is returned back to the nameserver.

[0087] At operation 1035, the nameserver provides information from the retrieved DNS record to the client device for accessing the webpage. For example, the nameserver can extract or retrieve an IP address for the requested webpage from the retrieved DNS records. In some embodiments, the nameserver can contact the web server hosting the webpage using the IP address and retrieve the requested webpage on behalf of the client device.

[0088] FIG. 11 illustrates a block diagram for an exemplary data processing system 1100 that may be used in some embodiments. One or more such data processing systems 1100 may be utilized to implement the embodiments and operations described with respect to the nameserver 102, the domain registrar blockchain 106, and the DNS resolver blockchain 112, or other computing devices. The data processing system 1100 is a computing device that stores and transmits (internally and/or with other computing devices over a network) code (which is composed of software instructions and which is sometimes referred to as computer program code or a computer program) and/or data using machine-readable media (also called computer-readable media), such as machine-readable storage media 1110 (e.g., magnetic disks, optical disks, read only memory (ROM), flash memory devices, phase change memory) and machine-readable transmission media (also called a carrier) (e.g., electrical, optical, radio, acoustical or other form of propagated signals—such as carrier waves, infrared signals, digital signals, etc.). In addition, such computing devices typically include a set of one or more processors coupled to one or more other components, such as one or more storage devices, user input/output devices (e.g., a keyboard, a touchscreen, and/or a display), and network connections. The coupling of the set of processors and other components is typically through one or more busses and bridges (also termed as bus controllers). The storage device and signals carrying the network traffic respectively represent one or more machine-readable storage media and machine-readable communication media. Thus, the storage device of a given computing device typically stores code and/or data for execution on the set of one or more processors of that computing device. Of course, one or more parts of an embodiment of the invention may be implemented using different combinations of software, firmware, and/or hardware.

[0093] In the preceding description, numerous specific details are set forth. However, it is understood that embodiments of the invention may be practiced without these specific details. In other instances, well-known circuits, structures, and techniques have not been shown in detail in order not to obscure the understanding of this description. Those of ordinary skill in the art, with the included descriptions, will be able to implement appropriate functionality without undue experimentation.

[0094] References in the specification to “one embodiment,” “an embodiment,” “an example embodiment,” etc.,
indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an embodiment, it is submitted that it is within the knowledge of one skilled in the art to effect such feature, structure, or characteristic in connection with other embodiments whether or not explicitly described.

[0095] In the preceding description and the claims, the terms “coupled” and “connected,” along with their derivatives, may be used. It should be understood that these terms are not intended as synonyms for each other. “Coupled” is used to indicate that two or more elements, which may or may not be in direct physical or electrical contact with each other, co-operate or interact with each other. “Connected” is used to indicate the establishment of communication between two or more elements that are coupled with each other.

[0096] While the flow diagrams in the figures show a particular order of operations performed by certain embodiments of the invention, it should be understood that such order is exemplary (e.g., alternative embodiments may perform the operations in a different order, combine certain operations, overlap certain operations, etc.).

[0097] While the invention has been described in terms of several embodiments, those skilled in the art will recognize that the invention is not limited to the embodiments described, can be practiced with modification and alteration within the spirit and scope of the appended claims. The description is thus to be regarded as illustrative instead of limiting.

1. A method, comprising:
   performing an off-chain registration process for a domain name with an off-chain domain registrar, the off-chain registration process including verifying and reserving the domain name with the off-chain domain registrar accredited by the Internet Corporation for Assigned Names and Numbers (ICANN);
   responsive to the off-chain registration process, performing an on-chain registration process for the domain name with a domain registrar blockchain, wherein an on-chain asset for the domain name is stored at the domain registrar blockchain, and wherein the on-chain asset enables a controller relationship over the domain name;
   receiving a first request from a client device, the first request to access a webpage using the domain name;
   determining that a domain name associated with the webpage is stored in the domain registrar blockchain;
   generating a first blockchain query based on the first request, the first blockchain query for querying the domain registrar blockchain;
   sending the first blockchain query to the domain registrar blockchain, the domain registrar blockchain storing domain name registration data in smart contracts in the domain registrar blockchain;
   receiving a nameserver identifier from the domain registrar blockchain;
   retrieving a DNS record for the domain name associated with the webpage using the received nameserver identifier; and
   providing information from the retrieved DNS record to the client device for accessing the webpage.

2. The method of claim 1, wherein retrieving the DNS record for the webpage using the received nameserver identifier comprises:
   generating, by the nameserver, a second blockchain query including the received nameserver identifier;
   sending the second blockchain query to a DNS resolver blockchain, the DNS resolver blockchain storing DNS records in smart contracts in the DNS resolver blockchain;
   and
   receiving the DNS record for the webpage from the DNS resolver blockchain.

3. The method of claim 2, wherein the DNS resolver blockchain for retrieving the DNS record for the webpage is designated by the domain registrar blockchain.

4. The method of claim 1, wherein generating the first blockchain query based on the first request comprises:
   translating a nameserver lookup request of a first request type to the first blockchain query of a second request type.

5. The method of claim 1, wherein the nameserver receives the first request responsive to a forward lookup process identifying the nameserver as an authoritative nameserver for the domain name.

6. The method of claim 1, wherein performing the on-chain registration process for the domain name with the domain registrar blockchain further comprises:
   sending a blockchain transaction to the domain registrar blockchain, wherein the blockchain transaction causes the domain registrar blockchain to mint the on-chain asset for the domain name, store the on-chain asset to a digital wallet associated with a user, and store a mapping between the domain name and the digital wallet in a smart contract.

7. A non-transitory machine-readable storage medium that provides instructions that, when executed by a processor, cause said processor to perform operations comprising:
   performing an off-chain registration process for a domain name with an off-chain domain registrar, the off-chain registration process including verifying and reserving the domain name with the off-chain domain registrar accredited by the Internet Corporation for Assigned Names and Numbers (ICANN);
   responsive to the off-chain registration process, performing an on-chain registration process for the domain name with a domain registrar blockchain, wherein an on-chain asset for the domain name is stored at the domain registrar blockchain, and wherein the on-chain asset enables a controller relationship over the domain name;
   receiving a first request from a client device, the first request to access a webpage using the domain name;
   determining that a domain name associated with the webpage is stored in the domain registrar blockchain;
   generating a first blockchain query based on the first request, the first blockchain query for querying the domain registrar blockchain;
   sending the first blockchain query to the domain registrar blockchain, the domain registrar blockchain storing domain name registration data in smart contracts in the domain registrar blockchain;
   receiving a nameserver identifier from the domain registrar blockchain;
   retrieving a DNS record for the domain name associated with the webpage using the received nameserver identifier; and
   providing information from the retrieved DNS record to the client device for accessing the webpage.
retrieving a DNS record for the domain name associated with the webpage using the received nameserver identifier; and providing information from the retrieved DNS record to the client device for accessing the webpage.

8. The non-transitory machine-readable storage medium of claim 7, wherein retrieving the DNS record for the webpage using the received nameserver identifier comprises:
generating, by the nameserver, a second blockchain query including the received nameserver identifier;
sending the second blockchain query to a DNS resolver blockchain, the DNS resolver blockchain storing DNS records in smart contracts in the DNS resolver blockchain; and receiving the DNS record for the webpage from the DNS resolver blockchain.

9. The non-transitory machine-readable storage medium of claim 8, wherein the DNS resolver blockchain for retrieving the DNS record for the webpage is designated by the domain registrar blockchain.

10. The non-transitory machine-readable storage medium of claim 7, wherein generating the first blockchain query based on the first request comprises:
translating a nameserver lookup request of a first request type to the first blockchain query of a second request type.

11. The non-transitory machine-readable storage medium of claim 7, wherein the nameserver receives the first request responsive to a forward lookup process identifying the nameserver as an authoritative nameserver for the domain name.

12. The non-transitory machine-readable storage medium of claim 7, wherein performing the on-chain registration process for the domain name with the domain registrar blockchain further comprises:
sending a blockchain transaction to the domain registrar blockchain, wherein the blockchain transaction causes the domain registrar blockchain to mint the on-chain asset for the domain name, store the on-chain asset to a digital wallet associated with a user, and store a mapping between the domain name and the digital wallet in a smart contract.

13. An apparatus, comprising:
a processor; and
a non-transitory machine-readable storage medium coupled with the processor, the non-transitory machine-readable storage medium storing instructions that, when executed by the processor, cause said processor to perform the following operations:
perform an off-chain registration process for a domain name with an off-chain domain registrar, the off-chain registration process including verifying and reserving the domain name with the off-chain domain registrar accredited by the Internet Corporation for Assigned Names and Numbers (ICANN); responsive to the off-chain registration process, perform an on-chain registration process for the domain name with a domain registrar blockchain, wherein an on-chain asset for the domain name is stored at the domain registrar blockchain, and wherein the on-chain asset enables a controller relationship over the domain name;
receive a first request from a client device, the first request to access a webpage using the domain name; determine that a domain name associated with the webpage is stored in the domain registrar blockchain;
generate a first blockchain query based on the first request, the first blockchain query for querying the domain registrar blockchain;
send the first blockchain query to the domain registrar blockchain, the domain registrar blockchain storing domain name registration data in smart contracts in the domain registrar blockchain;
receive a nameserver identifier from the domain registrar blockchain;
retrieve a DNS record for the domain name associated with the webpage using the received nameserver identifier; and provide information from the retrieved DNS record to the client device for accessing the webpage.

14. The apparatus of claim 13, wherein retrieving the DNS record for the webpage using the received nameserver identifier further causes said processor to:
generate, by the nameserver, a second blockchain query including the received nameserver identifier;
send the second blockchain query to a DNS resolver blockchain, the DNS resolver blockchain storing DNS records in smart contracts in the DNS resolver blockchain; and receive the DNS record for the webpage from the DNS resolver blockchain.

15. The apparatus of claim 14, wherein the DNS resolver blockchain for retrieving the DNS record for the webpage is designated by the domain registrar blockchain.

16. The apparatus of claim 13, wherein generating the first blockchain query based on the first request further causes said processor to:
translate a nameserver lookup request of a first request type to the first blockchain query of a second request type.

17. The apparatus of claim 13, wherein the nameserver receives the first request responsive to a forward lookup process identifying the nameserver as an authoritative nameserver for the domain name.

18. The apparatus of claim 13, wherein performing the on-chain registration process for the domain name with the domain registrar blockchain further causes said processor to perform the following:
send a blockchain transaction to the domain registrar blockchain, wherein the blockchain transaction causes the domain registrar blockchain to mint the on-chain asset for the domain name, store the on-chain asset to a digital wallet associated with the user, and store a mapping between the domain name and the digital wallet in a smart contract.