Using Decentralized Identifiers and InterPlanetary File System to Create a Recoverable Rare Disease Patient Identity Framework

Sean McHale
Department of Computer Science
Vanderbilt University
Nashville, TN
sean.r.mchale@vanderbilt.edu

Lincoln Murr
Department of Computer Science
Vanderbilt University
Nashville, TN
lincoln.d.murr@vanderbilt.edu

Peng Zhang, Ph.D.
Department of Computer Science
and Data Science Institute
Vanderbilt University
Nashville, TN
peng.zhang@vanderbilt.edu

Abstract—This paper presents a novel framework for creating a recoverable rare disease patient identity system using blockchain and smart contracts, decentralized identifiers (DIDs), and the InterPlanetary File System (IPFS). Smart contracts are executable code that can be written into decentralized storage such as blockchains in order to enable tamper-proof transactions of data. DIDs provide a secure, decentralized, and extensible way to create, store, and manage digital identities, while IPFS provides a distributed, immutable, and secure storage system for patient identities. Utilizing these technologies with smart contracts, we created a framework to store persistent medical records of patients. Smart contracts additionally allow account recovery without the use of any centralized authority. The framework enables healthcare providers to securely access a patient's data while maintaining the patient's ownership of their data. The paper explores the advantages of using a decentralized identity system and highlights the potential of this approach to improve the security and universality of medical records for patients with rare diseases.

Index Terms—blockchain, Hyperledger Fabric, smart contracts, decentralized identifiers, verifiable credentials, healthcare attestations, rare disease, InterPlanetary File System, healthcare interoperability, patient identity

I. Introduction

The National Institutes of Health (NIH) estimates that 7,000 rare diseases exist in the United States, impacting between 25 and 30 million Americans [1]. According to the Orphan Drug Act, a disease is defined as a rare disease if it impacts less than 200,000 people in the U.S. [2]. Rare diseases can be challenging to diagnose and often lack effective interoperability among healthcare systems. They are a critical healthcare issue that affects a large population of people yet receive less scientific and commercial attention than other medical conditions. This lack of support can lead to limited access to appropriate healthcare and a longer diagnosis time. Therefore, it is essential to find better methods to manage and thus treat these conditions to improve the lives of those affected.

Approximately 12 million individuals are misdiagnosed yearly, resulting in 40,000 to 80,000 fatalities annually [3]. Diagnosing a rare disease can be a long and complex process involving referrals from multiple specialists. This commonly results in the age-old problem of conventional patient identification matching. Linking patient data from various healthcare providers can prolong the diagnosis period, with the average time to diagnosis being 7.6 years [4]. Patient identification matching is a fundamental component in health data exchange, and existing patient identification matching systems do not provide ample support for patients with rare diseases. A recent survey of rare disease patients revealed that a proper diagnosis typically requires 8 physician visits [4]. The tool to record patient data, Electronic Health Record (EHR) systems, have three main problems: manual reentry of personal information, inaccurate and lengthy matching process of patient data, and the lack of a unified identity management system.

It is estimated that 195,000 deaths annually in the United States are caused by medical errors, with 60% of these being attributed to "wrong patient" errors [5]. Patients with rare diseases are particularly vulnerable to medical errors due to complex care plans, the need for communication between multiple providers, and the increased chance of incorrect diagnoses. In addition, rare disease patients require medical opinions from numerous specialists over an extended period, thus elevating the risk of medical mistakes. The magnitude of this problem is staggering, and it is clear that accurately matching patient data among healthcare providers is a vital and urgent issue that needs to be addressed.

Consequently, the rare disease community faces numerous challenges in accurately and consistently matching patient data, leading to duplicate patient records, inaccurate medical data, and delayed treatments. The current standards for identity management do not meet the demands required for patients with rare diseases. There is no universal system for storing and managing patients' identities with rare diseases. Individual healthcare providers provide patients with unique IDs; however, these are often unable to transfer over directly to new systems [27].

The lack of a universal system hinders patients' medical

National Science Foundation

progress with rare diseases. The correct information is difficult to retrieve without a standardized system and can make diagnosis and treatment challenging for the patient and the healthcare provider. In the era of big data, accessing abundant and accurate information is a powerful tool to improve medical care for rare diseases. Statistical analysis of consistent data can improve diagnosis, treatment, and communication for patients and care providers. With a universal standard, patients and providers can access complete patient histories, which will aid doctors in determining the best possible care they can provide. Furthermore, a universal system will improve communication between patients, care providers, and other stakeholders. Ultimately, a universal system can help develop better treatment and enable more comprehensive care for patients with rare diseases.

In response to this need, this research paper explores using Decentralized Identifiers (DIDs) and the InterPlanetary File System (IPFS) to create a recoverable patient identity framework. DIDs are self-sovereign digital identifiers that enable individuals to control their digital identities and securely access digital services. IPFS is a distributed file system that enables the permanent storage and linking of data on a global peer-to-peer network. We will explore how the combination of DIDs, IPFS, and smart contracts can create a unified, interoperable, and secure identity management system for the rare disease community that overcomes the limitations of existing centralized solutions. Sections II-V will discuss key concepts and existing efforts related to this research, our proposed identity management system, and the critical lessons learned.

II. Background and Related Work

In this section, we discuss the concepts and related technologies of blockchains that form the basis of our research effort, emphasizing the mechanisms that make them appropriate for the patient identity framework. Additionally, we review existing research on decentralized identity management that pertains to our work.

A. Key Concepts

Blockchain. Blockchain is a distributed ledger technology used to transfer value between individuals in a trustless, decentralized manner [15]. Blockchain networks are managed by nodes, each with a copy of the ledger, and reach a consensus about its state at predefined intervals.

In blockchains, private and public keys are critical components for cryptography. They provide users with a secure and reliable manner to control their accounts. Public keys identify the individual making the transaction and are visible to anyone. Private keys to sign the transactions sent to the blockchain network. Private keys are kept confidential to prove ownership of the user's transactions.

Blockchains are public networks that anyone may access and use. In contrast, private blockchains are permissioned networks with access restricted to specific participants. They are designed to be more secure and efficient than typical blockchains. The participants in a private blockchain can be controlled through access control methods such as lists, digital signatures, and other forms of authentication.

Hyperledger Fabric. Hyperledger Fabric is an open-source toolkit developed by IBM in 2014 [23]. This toolkit aims to simplify the creation and maintenance of private blockchains. This platform enables the development of decentralized applications to host secure and efficient business transactions. With Hyperledger Fabric, the organization that deployed the network can create role-based access and set permission levels for different individuals. The main components of Hyperledger Fabric are Hyperledger Fabric Nodes, which store and transmit data within the network. Nodes maintain the ledger, validate transactions, and execute smart contracts.

Smart Contracts. A smart contract is an agreement stored on a distributed ledger like a blockchain and enforced by logic and code. Smart contracts enable the direct exchange of anything of value, including money, data, and property, transparently while not requiring the intervention of a third party.

One of the main benefits of smart contracts is their ability to be self-executing, meaning that they can automatically initialize a task based on pre-set conditions. For example, a smart contract could exchange money between two people on a specific date. Blockchain-based smart contracts are also immutable, meaning they cannot be changed after creation.

DID Documents and Verifiable Credentials. DID documents are the first form of DIDs, initially introduced by the World Wide Web Consortium (W3C) in 2016. These self-sovereign digital documents allow individuals to manage their digital identities securely and verifiably [18]. They can contain personally identifiable information (PII), such as a name, address, or identification number, in an encrypted format. Moreover, DID documents utilize cryptography and cryptographic keys to prove the authenticity and identity of the document's owner.

Polygon ID is a decentralized self-sovereign identification system that utilizes zero-knowledge cryptography to preserve privacy and is one of the first solutions adopting W3C's DID standard [20]. Its four main properties are its blockchain-based ID, zero knowledge protocol utilization, scalable and private on-chain verification, and adherence to existing standards.

To validate themselves with the service, users download the Polygon ID app onto their phones, store their seed phrase, then complete third-party verification through Jumio, a centralized digital identity verification platform. Once this data is verified, the Polygon ID wallet related to the user, with no further communication with Jumio, can now prove its status as belonging to a human.

DID Documents are JSON-LD objects with specific fields defining them as valid documents. These fields include the DID itself, an authentication method such as public keys, the cryptographic protocols used for authentication, service endpoints describing how to interact with the document, and timestamps. Other optional fields include a JSON-LD signature and a controller authorized to make changes. The modu-

larity of the DID Document specification makes it applicable to a wide variety of use cases.

Verifiable credentials are a digital identification method that can prove specific attributes about an individual without relying on a third party for constant verification [19]. They incorporate cryptography, specifically digital signatures, which allow verifiers to ensure that credentials are issued by the entity that says they issued them. There are four roles in the verifiable credential ecosystem:

1) Holders are entities possessing verifiable credentials and can create a verifiable presentation, which can be used to authenticate ownership.
2) A subject is an entity about which the verifiable credential makes a claim.
3) The issuer is the organization or entity in charge of creating verifiable credentials and claims about a subject.
4) Verifiers can use verifiable presentations to prove that a holder owns a credential.

Verifiable credentials have many of the same properties as physical credentials. For example, the credential's authenticity is only as valid as the issuer, and there could be fraudulent verifiable credentials that come from a malicious issuer. However, the cryptographic guarantees provided by verifiable credentials provide more protection against tampering and fraud after issuance. In addition, unlike their physical counterpart, verifiable credentials can be easily and quickly shared over the internet.

Like DID Documents, verifiable credentials can be stored on a distributed ledger to ensure that the holder has tangible ownership over their credential and its data, preventing any central organization from making an unauthorized claim on a credential. Moreover, decentralized identifiers can be used with verifiable credentials to make claims about an individual and their actions. For example, a verified credential can prove that a specific DID, which would be associated with an individual, has a driver's license, vaccination record, or diploma from a given institution.

Healthcare Attestations. In our use case, verifiable credentials provide the framework for healthcare attestations. Healthcare attestations are documents verifying that a patient received a healthcare service. The healthcare providers who performed the service sign the attestation. The attestation will typically include a description of the services, the date of service, and any appropriate diagnosis. Healthcare attestations are essential as they provide an authenticated record of the patient's treatment and can be used to recommend future care.

InterPlanetary File System. The InterPlanetary File System (IPFS) is a peer-to-peer file system that uses content addressing to store data in an immutable and decentralized manner [24]. When a file is uploaded, or pinned, to IPFS, it is assigned a hash which is then used to identify the file. The global network of IPFS nodes stores the file and its associated hash. Filecoin is a blockchain network designed to be the decentralized payment facilitator for IPFS [25]. Users can pay in Filecoin to have their content hosted on servers, leading to more redundancy and reducing the possibility of all nodes deciding to no longer host the content. Other services, such as Pinata, abstract the cryptocurrency payment processing out of Filecoin and offer to store files for fiat payment, leading to a more seamless experience for end users at the cost of decentralization. Our solution utilizes Pinata's APIs to upload and pin DID Documents. It may be beneficial to use a more decentralized service in the future. However, Pinata provides an easy onramp and simple APIs that make it ideal for the current iteration of our healthcare identity management solution.

B. Related Work

In this section, we will explore related work in three distinct categories. First, we will explore existing centralized solutions to manage patients with rare diseases. Second, we discuss the use of DIDs and IPFS in identity management systems. Finally, we review existing blockchain-based solutions to manage and recover patient identities.

Centralized Methods for Managing Healthcare Identity. Currently, the most widely used patient identity management (PIM) approaches are centralized solutions that revolve around a universal patient identifier (UPI). A UPI is defined as a "unique, non-changing alphanumeric key for each patient" which is linked to the patient's medical record and healthcare data [6]. Generally, the healthcare industry welcomes the use of UPI as it increases interoperability, reducing errors and inefficiencies in matching and maintaining patients. However, it has been criticized for its possible adverse effects on patient privacy and data security [7].

In the United States, the National Database for Autism Research utilizes a global unique identifier for every patient [8]. This database de-identifies human subject data and aggregates this data across hundreds of research projects. The data is integrated into a common standard and available to qualified researchers. Another example is the Society of Thoracic Surgeons Database which uses unique identifiers for patients, surgeons, and hospitals [22]. They combine their UPIs with data from the Social Security Death Master File to manage patients over a long duration [9].

The Epic medical software, created by Epic Systems Corporation, is a medical record system used to manage electronic medical records (EMR) and electronic health records (EHR). Epic is used to store records of over 300 million patients [21]. It is well known for its interoperability across compatible EHRs, which allows it to facilitate the transfer of over 100 million records per month [21]. It is also a comprehensive platform with tools to manage various clinical, financial, and administrative tasks, including scheduling, electronic prescribing, and billing.

The Rare Diseases Clinical Research Network (RDCRN) Contact Registry is a registry of volunteer rare disease research participants that stores and maintains patient information on a cloud service. There is a challenge in linking one participant's data across multiple databases and sources, which RDCRN is addressing by proposing to use NCATS GUID to encrypt

patient data and link it to one unique identifier across all databases [10].

The European Unified Patient Identity Management (EUPID) provides secure identity management for cancer research among children. They use encryption and hashing to allow pseudonymization sharing of data. Additionally, in Sweden, all permanent residents are given a personal identity number (PIN). This PIN is used nationwide to link patients among different national patient registries [11].

DIDs and IPFS in Identity Management Systems. uPort is a decentralized platform for self-sovereign identity, meaning identity owners are in control rather than a centralized authority [12]. This framework enhances security by removing incentives from harmful agents trying to gain access to centralized databases that store millions of identities. This identity system allows users to control their data, selectively disclose it to counterparties, and interact with decentralized applications with their proven identity. uPort relies on the uPort identifier, a 20-byte hexadecimal string, which acts as a unique identifier. Then the user, through a series of smart contracts, can interact with applications or manage their identity. The identities are tracked in a Registry contract that maps the uPort identifier to a profile in a decentralized storage system called IPFS. Storing data on a blockchain is costly; IPFS offers a better alternative.

The World Wide Web Consortium (W3C) has developed a set of standards and best practices for creating and managing digital identities (DID). There are specific syntaxes and properties that all DIDs must follow. These identifiers allow entities to prove ownership using cryptographic proof such as digital signatures.

Blockchain Based Solutions for Managing Healthcare Data. MediLinker is a blockchain-based EMR system created by a team at the University of Texas at Austin [16]. It utilizes the Hyperledger Indy self-sovereign identity framework to store patient DIDs and verifiable credentials. User data, including medical history, is stored in a web application.

FHIRChain uses the HL7 Fast Healthcare Interoperability Resources (FHIR) standard to exchange medical information on a blockchain-based framework. The framework includes a token-based permission model that provides role-based authentication to the patients [13]. This project additionally has a decentralized application that analyzes subsequent collaborative decision-making from the secure and scalable data sharing the blockchain allows.

MedicalChain is a decentralized platform that uses a dual blockchain structure for storing and sharing electronic health records (EHR) [14]. It allows an ecosystem of interconnected healthcare applications to interact with each other securely. It is built with Hyperledger Fabric to link an on-chain hashed patient identifier to an off-chain EHR.

C. Differentiating Proposed Research from Related Work

All centralized solutions for patient identity management have two main disadvantages stemming from their centrality: they are less secure, and the individual does not own their identity [17]. Hosting millions of identities with valuable data in a centralized database incentivizes harmful agents to attempt to steal the information. Additionally, once in a centralized database, the patient is unaware of how their data is being used and shared. The previous works explored do not meet our standards of decentralization, do not meet the demands of identity management for rare diseases, or do not emphasize interoperability through W3C standards or IPFS.

Even though Epic is one of the largest EHRs, it is not without issues. For example, since Epic can only interoperate with other compatible EHRs, a small rare disease practice may not benefit from this feature if they do not have the funding to support an Epic-based EHR system. Additionally, data is centralized at the medical system level, such as in a hospital system, and faces all of the same issues as any centralized database: data breaches, physical destruction of the server, and control by a single entity.

An analysis of uPort reveals the lack of digital identity standards and failure to function for a rare disease use case. The uPort identifiers do not meet the W3C standards and thus would not be ideal for building a highly interoperable rare disease identity management system. Additionally, this platform assumes that the user is physically capable of managing their identity. However, in many instances, the diseased individual cannot access the platform themself. Thus they need a representative (parent, relative, or a trusted individual) to access the application and perform actions for them. These changes will be discussed in our proposed identity management system.

While MediLinker utilizes blockchain and decentralized identifiers, a couple of issues with its model need to be addressed. First, it uses Amazon Web Services, a centralized entity, as its host for Hyperledger Indy due to its HIPAA compliance. This reliance on one data source calls into question whether or not this solution provides decentralized identifiers. Additionally, in an eight-task usability test, three of the tasks saw multiple users fail while attempting to onboard. There is also no public documentation of MediLinker being used in industry.

Rare diseases present particular challenges that need to be incorporated into the framework's design. For example, some patients may have impaired motor function or cognition and require a delegate to make and approve decisions on their behalf. Moreover, given the rarity of the disease that patients may have, interoperability among as many systems as possible is critical, as many of the healthcare facilities patients attend may be small and not interoperable with something like Epic's standards.

In our delegate system, we incorporated enhanced security features to help prevent possible adversarial attacks on patient identities. Although it only takes one delegate to initiate an account recovery (or change of public key), a majority vote of delegates is needed to incorporate such a change. Additionally, after the initial onboarding process, the patient or their primary representative can only add or delete delegates once per week. If a patient's private key (their phone) were compromised, the other delegates would have ample time to change the patient's public key before the adversary could

add many adversary delegates or revoke the voting privileges of legitimate delegates.

By incorporating IPFS into our design, the DIDs are stored independently from the blockchain. This provides greater immutability, interoperability, and future-proofing, as the underlying ledger could one day be modified or upgraded to fit some future healthcare blockchain standard while ensuring that the data is intact. Additionally, since the DIDs are distributed across thousands of IPFS nodes, the problems of centralization and potential data loss are mitigated significantly.

III. Current Challenges

Creating a rare disease patient identity framework poses challenges due to the uniqueness of the situation. As previously stated, many diseases require consultation from multiple specialists, each with their own patient identity system, leading to the potential for inaccurate data across healthcare providers and fragmented records. A solution where the patient owns and custodies their digital identity creates its own set of problems, such as creating this consistent medical record that the patient owns and controls and making it interoperable among healthcare providers.

Moreover, there could be situations where patients cannot correctly authenticate themselves and their digital identity. This could be for numerous reasons, including the loss of verification devices such as a phone or computer. In other situations, patients may be unable to act on their behalf or consent to treatment, such as if they are a minor or have a disease that inhibits their communication or mental state [26].

Healthcare regulations and the proper storage of protected health information (PHI) must also be considered. Since anyone can access a file on IPFS if they have the content identifier (CID), PHI cannot be stored directly on IPFS. Even if it was encrypted and unpinned by the owner as soon as its encryption was broken, it may still be vulnerable to exploitation. This makes IPFS an inappropriate storage location for any sensitive data or PHI.

IV. Decentralized Identity Recovery Framework

A. Technical Overview

Smart contracts act as computer programs stored on the blockchain. They have a specific state that can be updated or changed depending on their underlying code. Our recovery mechanism consists of four smart contracts, similar to UPort [12]. Our implementation deviates by providing greater ease of use, specifically for patients who require another person to manage their account, along with additional security for account recovery. Some rare diseases limit the ability for an individual to use our application so they must have a person who represents them and thus be able to perform all actions an abled patient can do. Our framework allows a delegate entity called "representative" to have this authority.

The smart contracts, shown in Fig. 1, aim to provide a mechanism for a patient or a representative to access the main application where they can control their identity, interact with healthcare providers, and allow consumers to access their data. Additionally, they provide patients the ability to recover their identity if their access to their account is lost or compromised. To do this, we need to create a process for the patient and representative to interact with the main application. Secondly, we must have a recovery mechanism that will provide the patient with a consistent and reliable method to recover their identity.

B. Proposed Smart Contract Design

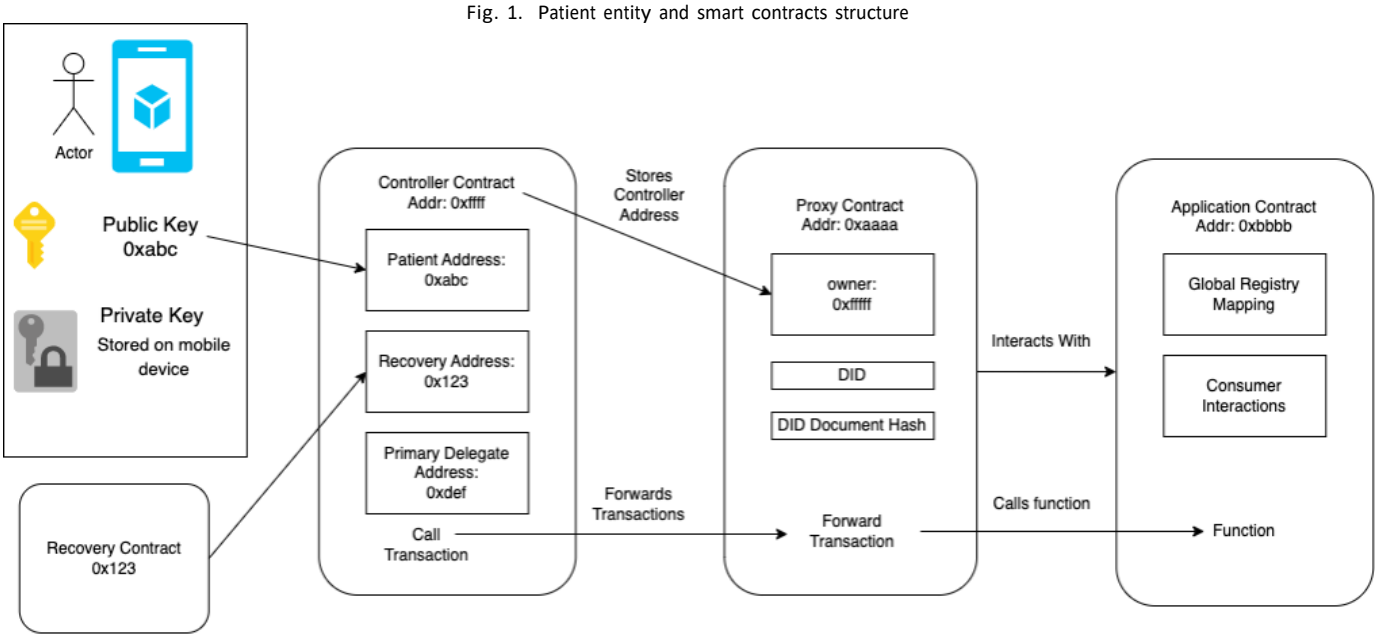
The proposed system revolves around two entities: the patient with the rare disease and the patient's representative. If the patient does not need/have a representative, the system will function the same. The patient entity will be assigned a public and private key. The private key is stored locally on the device and will not be shared. It is used to create a digital signature and authenticate transactions created by its owner. The patient and representative also have a public key, which is visible to everyone in the system. When a user sends a transaction, their public key is sent with the transaction for authentication.

To interact with the main application the patient entity will have a Controller Contract. The Controller Contract is a program that stores the public keys of the patient entity. When the patient entity wants to interact with the main application, they will send a transaction to the Controller Contract, which contains the functionality to interact with the application. For security, the Controller Contract will check that the public key of the sender of a transaction is either the patient address or the representative address stored in the contract. If it is not, the transaction will not process.

Attached to the Controller Contract is a Recovery Contract. When a patient is onboarded into the system, they will provide a list of delegates they trust. The public key addresses of these delegates will be stored in the Recovery Contract. The Recovery Contract has access to a function in the Controller Contract that can change the patient or representative address stored in the Controller Contract. If a patient or representative loses their private key, they can contact their delegates off-chain and provide them with a new public key. The Recovery Contract has the logic for any delegate to create a proposal to initiate the change of the public address stored in the Controller Contract. The delegates will vote on the proposal, and if the proposal receives a majority vote (or satisfies other conditions outlined later), the proposed change will take effect. The address stored in the Controller Contract will change, allowing the user to access the application with their new device.

When a user sends a transaction in the Controller Contract their transaction will be forwarded through the Proxy Contract before calling the Application Contract. The Proxy Contract is an intermediary between the Controller and the Application Contract. The Proxy Contract is a persistent data object to the user. It stores the Controller Contract's address along with digital identification documents discussed in the next section of the paper. Only the Controller Contract can call functions within the Proxy Contract.

Fig. 1. Patient entity and smart contracts structure



The Proxy Contract directly interacts with the Application Contract. The Application Contract contains the programs for the majority of the health care application. For example, in the Application Contract, patients and representatives are registered in registries, can update their profiles, and interact with other roles on the blockchain.

C. Smart Contracts

This section provides a detailed overview of the four smart contracts: Controller Contract, Recovery Contract, Proxy Contract, and Application Contract.

1) Controller Contract: The Controller Contract stores the public key of the patient and their representative (if they have one). The contract's constructor is called on deployment, which will set these stored variables. Additionally, the constructor deploys a Proxy and Recovery Contract unique to the patient entity. The constructor also calls the account creation function, which assists with creating a DID and linking it to the account. This is explored in detail in this section.

The patient entity can create transactions that go through the Proxy Contract to the application. The Controller Contract can change the public key associated with the patient and representative. This is necessary for the recovery mechanism since it is the only way for the patient's or representative's public key to change. Note that neither the patient nor the representative can change their public keys; only the Recovery Contract can. This is to enhance security and prevent a single entity compromise from undermining the ownership of the contract.

If the functionality of the Controller Contract needs to be changed it can be easily updated. First, a new Controller Contract is developed and published. Then, once complete, the original Controller Contract can call a function to change the owner of the Proxy Contract to the new Controller Contract.

Controller Contract Overview:

1) Stores the patient and representative public keys for authentication.
2) Calls the Proxy Contract to interact with the main application.
3) Has a function to change the public keys associated with the patient and representative. Only the Registry Contract (governed by the delegates) can call this transaction.

2) Recovery Contract: The Recovery Contract enables account recovery without requiring a new decentralized identifier if the patient loses their device. This process is illustrated in Fig. 2. The Recovery Contract stores all the delegates of the patient. Delegates can create and vote on proposals to change the public address of the patient. Each delegate consists of a struct with three variables: their public key, a boolean signifying if they voted or not, and a boolean if they are permitted to vote. The patient and representative can add delegates to the mapping, and each delegate is permitted to vote. The patient entity can revoke the right to vote from any delegate. For security purposes, they can only perform this action once a week.

When a patient or representative loses access to the device that stores their public key, they must contact a delegate (Fig. 2 Step 3). The user must have a contact method for each delegate added to the recovery contract, such as a phone number or email. When a delegate receives a notification that their user has lost their private key, they must verify that the claim is legitimate. After doing so, they can create a proposal in the Recovery Contract to give the patient a new key association

(Fig. 2 Step 5).

Once a proposal is created, other delegates can view and vote on the proposal (Fig. 2 Step 6). Delegates may either vote to support or oppose the proposed change. Each proposal has an expiration date of one week. Once the expiration date has passed, if yes votes are 50% or more of the total number of votes, the proposal is accepted, and the Recovery Contract will call a function in the Controller Contract to change the public key (Fig. 2 Step 8). A proposal is rejected if over 50% of delegates vote no. If neither vote count is a majority, a direct comparison takes place, with a tie accepting the proposal. This process ensures that if delegates are inactive a proposal can still pass. Any attack on the mechanism would require over 50% of the active delegates to be compromised.

Several security features are in place to prevent adversarial attacks against the patient's identity. First, the security of the patient relies on the delegates who can change the public key upon an off-chain request. Thus we can expect an adversary would either want to add many of their own "delegates" or revoke the ability to vote for actual delegates. Both of these attacks are mitigated by time constraints. After the onboarding process, a patient entity can only add or revoke a delegate once per week. This feature allows the delegates to initiate and process a change in the patient's public key before most of the delegate pool is adversarial.

Recovery Contract Overview:

1) Patients and representatives can add delegates whose public keys are stored in the contract.
2) Any delegate can create a proposal to change the patient's or the representative's public key.
3) Delegates can vote on the proposal. If it passes, the public key in the controller contract is changed. If it does not pass, nothing will change.

3) Proxy Contract: The Proxy Contract is the intermediary between the Controller Contract and the Application Contract. It receives forwarded transactions from the Controller, makes low-level calls to the Application Contract, and stores the DID Document associated with a given patient. Low-level calls allow the Proxy to call any function in the Application Contract. Additionally, the Proxy Contract's existence allows for the Application Contract's upgradeability without needing to change the Controller Contract. If a bug is found or a new feature is proposed, a new Application Contract can be created with the same information as the previous and expanded function definitions that the Proxy can call.

The Proxy Contract is the most crucial aspect of this framework as it is persistent for the patient entity. Even in the case of a social recovery or account migration, the Proxy Contract stays associated with a patient, permanently linking the DID Document to their blockchain public-private keypair. As such, it is also used to store verifiable credentials and is the main link between the blockchain and the DID Documents stored on IPFS. These are publicly visible, so anyone with access to the blockchain can verify the patient's authenticity.

Proxy Contract Overview:

1) Acts as a proxy between the Controller Contract and Application Contract.
2) Contains the patient's DID and any verifiable credentials
3) Is persistent for the user (even if the Controller Contract is updated).

4) Application Contract: The Application Contract will host interactions between the patient and other roles such as consumers, verifiers, and delegates. Each role has a corresponding global registry. The Patient Registry is a mapping from DID to Proxy Contract. This allows verifiers to search patients by DID and add attestations to the patient's DID Document in their Proxy Contract. The other roles are a mapping from public key to contract containing general information (these identities do not need the Recovery, Controller, Proxy structure since they are managed by the health care provider). The healthcare provider can assign roles to the public keys of consumers, verifiers, and delegates.

D. DIDs

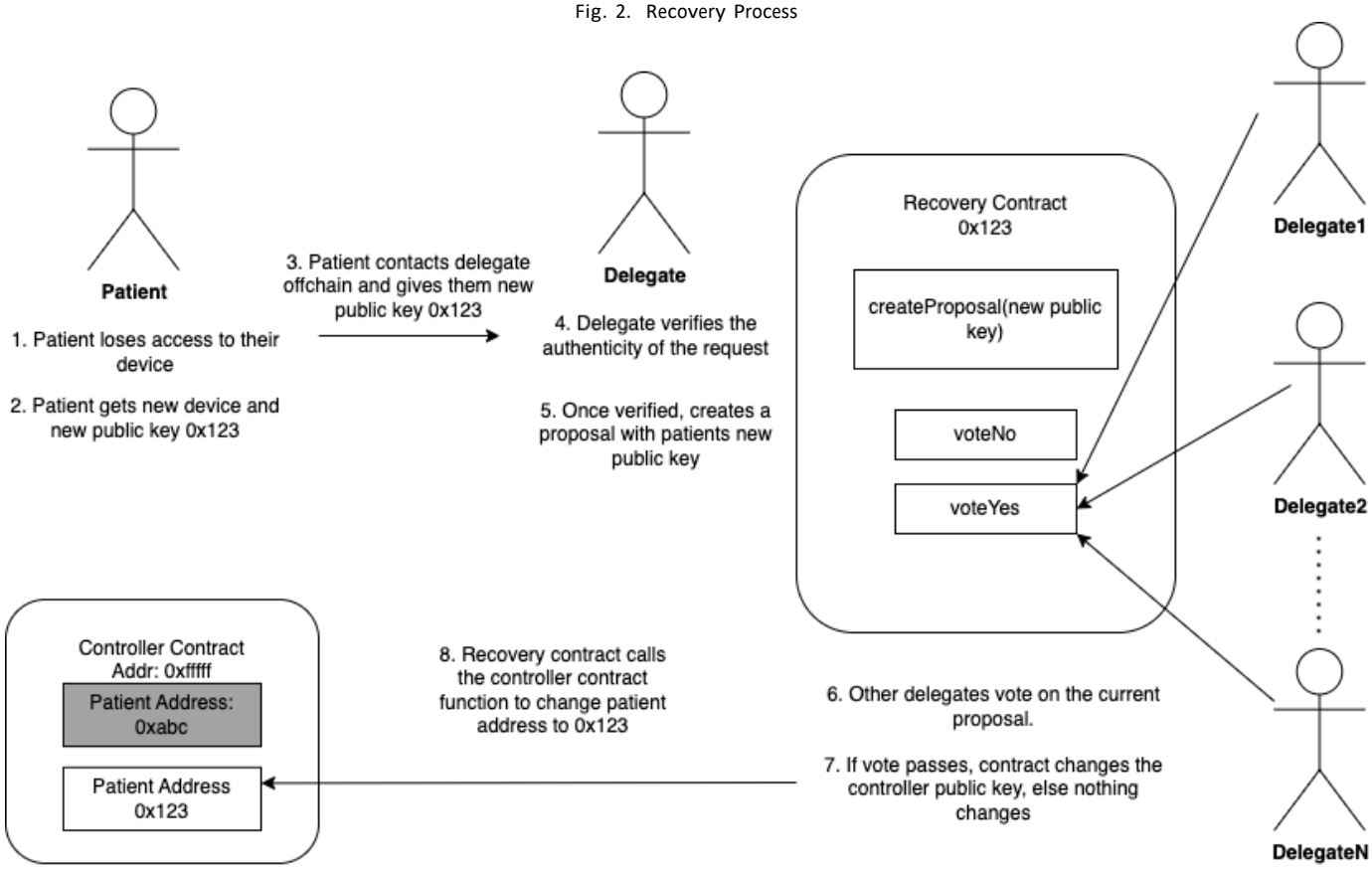
Decentralized Identifiers, or DIDs, are a form of digital identity that is distributed, verifiable, and can be owned by individuals. Unfortunately, current online identifiers suffer from many of the same problems as the healthcare industry, such as centralization, potential censorship or compromise, and a lack of efficiency. Through their distributed nature, DIDs can be independently verified by anyone, owned by the individual whose information is recorded, and distributed across thousands of nodes for a redundant and censorship-resistant identity experience.

The content identifier (CID) of the DID Document associates the IPFS-hosted DIDs with the smart contracts. This CID is the direct access link to the DID Document and is effectively a hashed version of the file. Thus, smart contracts can manage identity ownership on the distributed ledger by associating user wallets with a CID. Since each association of a DID with an address will be recorded on the blockchain, there is no way for a DID to be double-associated with two addresses without being noticed, nor is it possible for someone to make false claims related to their association with DIDs. Additionally, since information about who owns the DID is only stored on the permissioned blockchain ledger, the DID Document does not contain any compromising information about its owner, meaning that only those the identity holder approves can see patient data.

V. Discussion and Conclusions

Through the research presented in this manuscript, IPFS is utilized to create an immutable and censorship-resistant decentralized identifier that can be permanently accessible to patients and their delegates. The decentralized identifiers are the backbone of the system and provide an interoperable solution to the problem of associating a digital object with an individual. The InterPlanetary File System provides a distributed and neutral place to store these files, helping to preserve their autonomy and authenticity. The DID Document and its association with patients cannot be changed without

Fig. 2. Recovery Process



authorization. Therefore, they will likely not be lost due to the redundancy and distribution of the information across IPFS storage providers. The blockchain-based smart contracts create a system of ownership of these identities by their respective individuals and provide solutions for delegation and social recovery while still upholding security. Through this model, patients are, for the first time, able to own their digital identity and healthcare information.

For future work, we plan to explore zero knowledge technology and how it could provide greater security for patients and enable them to share properties about their medical care without revealing sensitive underlying information. This may also make it possible to store healthcare information on a public blockchain in a HIPAA-compliant manner, though much more research is needed in this area. Additionally, we plan to expand our identity system into other fields needing digital identification, such as a government database or a unique nurse identifier. This identification system for the nursing industry may help improve patient outcomes and interventions by creating a more transparent and efficient way to document and store care-related information.

Acknowledgment

This research was supported by National Science Foundation under the CISE-CRII program (project 2153232). The authors would like to thank Dr. Kelly Aldrich from Vanderbilt University and COMBINEDBrain (https://combinedbrain.org/) for providing domain expertise to this research.

References

[1] Genetic and Rare Diseases Information Center. [Online]. Available: https://rarediseases.info.nih.gov/. [Accessed: 31-Dec-2022].

[2] R. C. Griggs, M. Batshaw, M. Dunkle, R. Gopal-Srivastava, E. Kaye, J. Krischer, T. Nguyen, K. Paulus, P. A. Merkel, "Clinical research for rare disease: opportunities, challenges, and solutions," Molecular genetics and metabolism, vol. 96, no. 1, pp. 20–26, 2009.

[3] M. L. Graber, "The incidence of diagnostic error in medicine," BMJ Quality & Safety, vol. 22, no. Suppl 2, pp. ii21–ii27, 2013.

[4] C. Hendriksz, 'Rare Disease Impact Report: Insights from patients and the medical community', 06 2013.

[5] J. L. Fernandez-Alem ´ an, I. C. Se ´ nor, P. ˜ A. O. Lozoya, and A. Toval, "Security and privacy in electronic health records: A systematic literature review," Journal of biomedical informatics, vol. 46, no. 3, pp. 541–562, 2013

[6] R. Hillestad, J. Bigelow, B. Chaudhry, P. Dreyer, M. Greenberg, R. Meili, S. Ridgely, J. Rothenberg, R. Taylor, Identity Crisis: An Examination of the Costs and Benefits of a Unique Patient Identifier for the U.S. Health Care System. Santa Monica, CA: RAND Corporation, 2008.

[7] National Committee on Vital and Health Statistics (NCVHS); Subcommittee on Standards and Security. Hearing Minutes. Chicago, Il: Jul 20-21, 1998.
[8] S. Johnson, G. Whitney, M. McAuliffe, H. Wang, E. McCreedy, L. Rozenblit, C. Evans, Using global unique identifiers to link autism collections. J Am Med Inform Assoc. vol. 17, no. 6, pp. 689–95. 2010.
[9] J. Jacobs , C. Haan, F. Edwards, R. Anderson, F. Grover, J. Mayer, W. Chitwood, The rationale for incorporation of HIPAA compliant unique patient, surgeon, and hospital identifier fields in the STS database. Ann Thorac Surg. vol. 86, no. 3, pp. 695–698. 2008.
[10] Y. R. Rubinstein and P. McInnes, "Nih/ncats/grdr® common data elements: A leading force for standardized data collection," Contemporary clinical trials, vol. 42, pp. 78–80, 2015.
[11] J. Ludvigsson, P. Otterblad-Olausson, B. Pettersson, A. Ekbon, The Swedish personal identity number: possibilities and pitfalls in healthcare and medical research. Eur J Epidemiol. vol. 24, no. 11, pp. 659–67. 2009.
[12] N. Naik and P. Jenkins, 'uPort Open-Source Identity Management System: An Assessment of Self-Sovereign Identity and User-Centric Data Platform Built on Blockchain', in 2020 IEEE International Symposium on Systems Engineering (ISSE), 2020, pp. 1–7.
[13] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, "Fhirchain: applying blockchain to securely and scalably share clinical data," Computational and structural biotechnology journal, vol. 16, pp. 267–278, 2018
[14] Medicalchain, [Online] Available: https://medicalchain.com/en/
[15] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," International Journal of Web and Grid Services, vol. 14, no. 4, pp. 352–375, 2018.
[16] A. Khurshid, C. Holan, C. Cowley, J. Alexander, D. Harrell, M. Usman, I. Desai, J. Bautista, E. Meyer, 'Designing and testing a blockchain application for patient identity management in healthcare', JAMIA Open, vol. 4, no. 3, 02 2021.
[17] I. Yaqoob, K. Salah, R. Jayaraman, and Y. Al-Hammadi, 'Blockchain for healthcare data management: opportunities, challenges, and future recommendations', Neural Computing and Applications, vol. 34, no. 14, pp. 11475–11490, Jul. 2022.
[18] "Decentralized identifiers (dids) v1.0," W3C. [Online]. Available: https://www.w3.org/TR/did-core/. [Accessed: 09-Jan-2023].
[19] "Verifiable Credentials Data Model v1.1," w3.org, Mar. 3, 2022. W3C. [Online]. Available: https://www.w3.org/TR/vc-data-model/. [Accessed Jan. 9, 2023].
[20] C. Shetty, "What is Polygon ID?," polygon.technology, Jun. 24, 2002. [Online]. Available: https://support.polygon.technology/support/solutions/articles/82000889067-what-is-polygon-id-. [Accessed Jan. 9, 2023].
[21] Epic, "About Us," epic.com, [Online]. Available: https://www.epic.com/about. [Accessed Jan. 9, 2023].
[22] H. Gaissert, F. Fernandez, M. Allen, W. Burfeind, M. Block, J. Donahue, J. Mitchell, P. Schipper, D. Raymond, E. David, Patterson GA, "The Society of Thoracic Surgeons General Thoracic Surgery Database: 2016 Update on Research" Ann Thorac Surg, 2016. [Abstract]. Available: https://pubmed.ncbi.nlm.nih.gov/27772572/ [Accessed January 9, 2023].
[23] Hyperledger Foundation, "Hyperledger Fabric," hyperledger.org,. [Online]. Available: https://www.hyperledger.org/use/fabric. [Accessed Jan. 9, 2023].
[24] Protocol Labs, "IPFS Powers the Distributed Web," ipfs.tech,. [Online]. Available: https://ipfs.tech/. [Accessed Jan. 9, 2023].
[25] "What is Filecoin,", filecoin.io,. [Online]. Available: https://legacy-docs.filecoin.io/about-filecoin/what-is-filecoin/. [Accessed Jan. 9, 2023].
[26] F. Rafid, E. Benissan, L. Murr, I. Ermakov, K. Oikonomou, and P. Zhang, A Decentralized Identity System for Accelerating Medical Communications Within Rare Disease Communities, 2022 IEEE International Conference on Blockchain, Smart Healthcare and Emerging Technologies (SmartBlock4Health), October 24-25, 2022, Bucharest, Romania.
[27] B. Healthcare, "Patient matching peril: Why unique patient identifiers are a unique problem for hospitals," 2016.