Engineering and Technology Journal e-ISSN: 2456-3358

Volume 11 Issue 07 July-2026, Page No.-11029-11041

DOI: 10.47191/etj/v11i07.29, I.F. – 8.595

Privacy-Preserving Session-Bound Mother–Neonate Identity Verification with Permissioned Blockchain Audit Anchoring

Ihsan H. Hussein

Cybersecurity Engineering Technologies Department, College of Computer Engineering and Artificial Intelligence, Northern Technical University

ABSTRACT: Newborn misidentification poses serious patient safety and accountability problems, but errors can be traced through the use of a blockchain to create an audit trail. However, a blockchain storing raw or even hashed biometric templates for individual identities is not acceptable for privacy reasons. This work redefines our prior work (1) to form a privacy-preserving audit protocol that isolates the processes of capturing a biometric and matching it against a database of known identities to an external Service Provider and the processing of the blockchain to a permissioned Ledger that contains only pseudonymous audit commitments related to keyed entries on the Ledger. This work describes an implementation of this protocol in Solidity 0.8.30 and provides metrics for the gas use and latency of the smart contract for 100 iterations of 100 total Enrollment and Verification Workflows each. Twenty Adversarial Functional Tests are also described that attempt to place the system into an invalid state, as well as four additional tests that assess the effect of batched submission to the smart contract of multiple keyed audit commitments. The smart contract processing throughput is also determined for a batch of submissions, finding a maximum local throughput of 60.2 tx/s. A further 50,000 randomized reference-model transitions of the system's internal reference-model were then made (involving a total of 57,345,087 invariant checks, all of which passed), as well as a measurement of the time taken to generate an HMAC-SHA-256-sized commitment for 10,000 iterations (local median time = 0.002 ms). The results of this work provide a solid foundation for the blockchain component of BIBIS, but it is not intended to provide any insights into the accuracy of neonatal biometric matching, the presentation attack resistance of the system, or even the usability of BIBIS by clinical end-users. The results also do not comment on the finality of QBFT-based commits to a blockchain.

KEYWORDS: permissioned blockchain; biometric template protection; session binding; healthcare audit; HMAC commitment; revocation; newborn identification; smart contract security

1. INTRODUCTION

Error in identification of mothers and their newborns in maternity and neonatal healthcare can have severe consequences for clinical processes, such as giving and feeding, for administration of medication and for discharge of mother and child. The number of errors of identification of mothers and their newborns, that occur in maternity and neonatal care, as reported by several studies, does not give any information on the number of actual events of misidentification of mothers and their newborns. For example, the average of 1 event of identification error per 217 live births (range 0–8.9 events/1000 live births) as reported in [2] is based on reported incidents and as such does not provide any information on the probability of a single birth being misidentified. In addition, the risk of a patient being misidentified in the NICU was found to occur for more than half of the days of stay of a patient in the NICU, not due to a mix-up but because of names of patient and/or medical-record numbers of patient being similar. Thus, when designing a security system for the identification of mothers and their newborns in maternity and for audit purposes in maternity and neonatal healthcare, a clear distinction between the risk of misidentification and the risk of a mix-up should be made and safety data from such studies should not be translated into unsupported biometric probabilities.

In the BIBIS paper we presented a conceptual design for a mother–newborn verification workflow which used fingerprint recognition, off-line storage of templates, smart contracts, and a permissioned proof-of-authority blockchain for its back-end. The accuracy of biometric matching, false-accept rates, mix-up rates, and matching delay were presented for the various stages in the workflow. However, no clinical verification was performed. Most importantly, in the paper the plain SHA-256 template hashes were stored on-chain. Such a hash value can be used to detect changes of the stored template data. However, the reference (i.e. the stored hash value) does not have to be renewable (i.e. the reference has to be associated with the same person in different situations), and it does not have to be unlinkable (i.e. the stored hash value is not associated with other template information for the same person). In addition, in case of compromised information, storage of such information on-chain is not

sufficient. In a blockchain, consensus among the nodes in the network is reached with respect to the order and the authorized transitions of the ledger (i.e. the various versions of the data stored in the ledger). However, this consensus does not imply that the sensor captured correct person for whom recognition is performed, and/or that the recognition matcher reached the correct decision.

In summary, what the studied design does is maintaining tamper-evident, session-bound and even revocable audit trails on a consensus-based permissioned blockchain for a verification workflow for mothers and their newly born children, without storing any personal data on the blockchain. The study under review does not analyze any of the issues of the clinical verification process (i.e. fingerprint capturing and fingerprint matching) as if they were parts of the design for a verification workflow on a blockchain which the BIBIS paper introduced. Thus, the study under review does not provide any information on the accuracy of the fingerprint recognition of newborns, on the applicability of different types of fingerprint scanners for this verification process, or on the frequency of mix-ups in a clinical setting.

1.1 Contributions and Research questions

This work has several main contributions.

C1. A data-minimized on-chain/off-chain architecture: We describe an Integrity Anchoring design on the blockchain and data storage off the blockchain, which for the proposed Neonatal and Mother verification solution only stores pseudonymous keyed-commitments on the blockchain and keeps the rest of the data (raw biometric images, derived templates, match scores, clinical information of subjects involved etc. including session information and audit information) off the blockchain.

C2. A session-bound mother–infant verification protocol: Maternal verification has to be successfully done for a mother–infant verification to be admissible in the system. In order to prevent others to use a session for verification, each session is tied to a unique one-time identifier and secret token. After a certain time has passed the session is finalized and cannot be used in a replay attack anymore.

C3. Construct for revocable keyed-commitments: In contrast to simply hashing biometric templates with SHA-256 on the blockchain, HKDF is used for domain separation, HMAC-SHA-256 keyed hashes are created, nonces for each record are stored and even template cancelation on a per epoch basis is supported to allow for commitment rotation and even revocation of templates without exposing sensitive information of the individual.

C4. A Smart Contract for Permissioned-EVM Blockchain Architecture. In this work, we have designed and implemented the architecture proposed in this work using Solidity programming language for a Permissioned-EVM based blockchain architecture. The smart contract represents the finite state machine for the designed architecture for the proposed verification process. The Smart Contract enables the maternal verification, the two-person maternity registration, the template-commitment rotation and the template-commitment revocation and logging of audit-log events.

C5. A reproducible preliminary evaluation. The new architecture was evaluated in a preliminary test of 100 end-to-end workflows (100 mothers and 100 newborn babies) in 4 different concurrency scenarios, along with a reference implementation which was tested with 50,000 state transitions, 10,000 HMACs, and 20 tests, all reproducible in full.

C6. A clear statement of what the ledger layer is guaranteeing (i.e. ordering, integrity, accountability) and what it is not guaranteeing (i.e. biometric matching accuracy, presentation attack detection, sensor quality, clinical identity proofing, biological family relationship).

This evaluation was structured around three research questions that formed the evaluation's core structures. RQ1 sought to force the correct authorisation, dual control, one-time session consumption and subsequent expiry/rotation/revocation for a range of tested transition sequences through the proposed contract. RQ2 evaluated the EVM execution cost and local processing latency of the proposed audit protocol. RQ3 distinguished the security and privacy guarantees provided by the proposed protocol from those of the gateway, the biometric subsystem, the validator governance and the set of clinical procedures currently in place.

2. BACKGROUND AND RELATED WORK

2.1 Newborn identification and biometric limits

Finger print recognition of newborns and infants within the first months of life is often carried out by using wristbands. These wristbands have a tendency to detach and the wristband number to change during a hospital stay of a few days [4]. In order to be able to study the quality of the finger print images as well as the recognition of the finger print patterns of infants over time the first study, namely Infant-ID, focused on finger print recognition of infants. The study revealed the high value of age-aware high-quality finger print acquisitions by using high-resolution gray-scale finger print sensors. Subsequently a prospective clinical study was carried out in which the quality of the finger print acquisition by a non-contact finger print sensor was studied. The results of this study showed strong age-dependent effects on the quality of the acquired finger print patterns [6]. Follow-up study was carried out during a period of 4 months in which finger print images of 43 infants were collected directly after birth. The finger print images were collected at three different resolutions, i.e. 500 ppi, 1000 ppi and 3000 ppi. The results clearly showed that for very young children the highest quality finger print images were obtained at the highest resolution of 3000 ppi [7].

2.2 Biometric template protection

The protection of information of value has three objectives: confidentiality, integrity, and the ability to revoke (to

withdraw permission and reissue to another person) previously compromised information. The first approaches to cancelable biometrics were first introduced by Ratha et al. [10] in 2001. Although the hashing of databases containing biometric information for the purpose of storage protection is not sufficient to protect information in storage, a number of methods have been proposed for this purpose (database hashing). Later, various points of attack were analyzed in the entire biometric processing pipeline. Some of the approaches to protect information, so far in the literature, have already been used for the protection of biometric information. This class of approaches also includes methods for biometric cryptosystems, for secure sketches, for fuzzy commitments and for so-called fuzzy vaults. In this work, we are not interested in new approaches that can replace the approaches already listed for the protection of information in general and of biometric information in particular. Instead, we assume the existence of an off-chain system, in which so far un-compromised, protected, and so far un-cancelled information of value for a user is stored. Here, a new contribution is presented in the form of the ledger commitment, which in an uncontrolled fashion not only binds the active information representation of a user to a specific epoch, but also anchors it to an HMAC value of opaque nature.

2.3 Blockchain audit in healthcare

Many previous healthcare-blockchain studies detail the integrity, provenance tracking and audit-tracking benefits of implementing a blockchain-based system across institutions. However, there are many challenges in scaling such a system, including creating and managing a governance structure, ensuring privacy of medical information while sharing, and interoperability with currently used systems. MedRec put the metadata for who was allowed to read and write specific data on the blockchain, but the rest of the medical data was off-chain [18]. Most other prototypes for securing integrity of electronic health records also use some form of contracts on virtual machines (VMs) such as the Ethereum Virtual Machine (EVM) with the actual clinical data kept off-chain [19]. A remaining problem for such systems is to verify information that had been provided by a participant (healthcare provider) who was authorized to submit the information in the first place, i.e. verify the accuracy of the information as it was entered.

2.4 Research gap

There is a large gap of existing work in the intersection of biometric protection, clinical workflow, and ledger governance. While BIBIS from earlier this year [1] is based on direct template hashes and provides a theoretical analysis for FAR and timing for a very broad Proof-of-Authority setting, it is not realizable in practice. Studies [5–8] for biometric systems characterize their capture processes and matching performances. However, these studies do not extend to a consortium-level audit-governance framework. For template-protection in biometrics, a large set of studies [9–15] achieve the goals of revocability and non-invertibility of protected templates. A clinically meaningful session model is missing, i.e., mother-first verification for verifying mothers before verifying their children in follow-up sessions. Many healthcare systems [16–20] use blockchain for ensuring record-integrity and for storing access metadata. However, most of these systems do not have a biometric-specific revocation mechanism, and more importantly, they lack a set of clearly defined non-goals for such a system [34]. In this work, we bridge these gaps. A combination of keyed and of revocable commitments, session-bound mother-first verification, and of opaque audit-anchors embedded in an executable state machine with explicitly-scoped claims for a set of clearly defined goals and non-goals is presented. Future work includes an evaluation of the consensus layer and clinical validation.

3. SYSTEM MODEL, TRUST ASSUMPTIONS, AND DESIGN GOALS

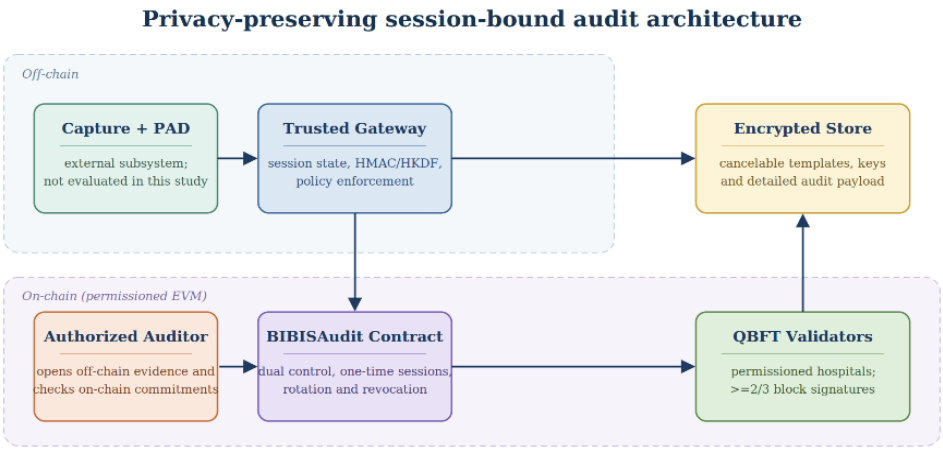

Figure 1. Privacy-preserving session-bound audit architecture.

3.1 Entities

Table 1. Entities and their responsibilities in the proposed architecture.

Entity	***Responsibility***
Registrar	Creates an enrollment proposal after clinical identity-proofing has been approved.
Supervisor	- an Operator (distinct from the Subject), e.g. who approved the enrollment of a Subject, who withdraws a previously given commitment, or who revokes a Subject.
Biometric gateway	The back end operator interfaces with capture/PAD/matcher components, manages the session secret, and submits authorized ledger transactions.
Protected store	Holds encrypted cancelable templates, nonces, detailed audit information and other key information off the blockchain.
QBFT validators	permissioned hospital or authority nodes that order and finally confirm the off-line transactions on the ledger.
Auditor	Off-line record (e.g. hospital) is checked to open to the on-line commitment; event is checked for authorized chronology.

3.2 Trust assumptions

• Enrollment procedure is based on the approved clinical process under which the mother–neonate pair is enrolled for the protocol (i.e., they are enrolled for the protocol and recorded but it does not necessarily prove that the biological mother of the neonate is enrolled for the protocol).

• All of the gateway's signing key, the ledger-commitment key, the session secrets and the transform seeds are stored in an HSM, a TEE or equivalent protection which is part of the operational environment.

• The biometric subsystem records a decision and corresponding PAD result for the enrolled template according to the matcher's policy that it has been configured with. The correctness of the ledger with respect to the protocol's objectives does not imply the correctness of the matcher with respect to its own policy.

• Safety and liveness of QBFT for n validators as a network follows from safety and liveness of the network of n validators as a set (i.e. the set of n validators configured to form the network) under the assumption that fewer than one third of the validators in the network are Byzantine. Insertion of a block into the blockchain of a network of validators requires a super-majority signature of the set of all validators in the network [22].

• As two authorized people are assumed to be malicious, the dual control for fraudulent enrollment of them by governance is under the control of governance and therefore under control of governance.

3.3 Threat model

Table 2. Threat model summary: adversarial objectives grouped by category (T1–T9).

Threat category	*Threat*	*Adversarial objective*
Protocol misuse	T1–T3	An adversary invokes contract functions without authorization (T1), replays a prior session token, identifier, or finalization transaction (T2), or combines a maternal success with a child step from another session (T3).
Data exposure	T4, T6	Protected templates or transform parameters are disclosed (T4), or an observer links audit entries to a patient or across contexts (T6).
Evidence manipulation	T5, T7	A participant modifies, removes, or reorders committed evidence (T5), or an authorized insider enrolls a false pair or submits a false result (T7).
Infrastructure attacks	T8, T9	A validator coalition attempts censorship or conflicting finality (T8), or a presentation artifact targets the capture/matcher path (T9).

3.4 Security goals and non-goals

Table 3. Security goals (G1–G6) and explicit non-goals (NG1–NG2) with their verification status.

Property	***Definition***	***Status***
G1 Authorization	Only role holders can invoke privileged transitions.	Implemented/tested
G2 Dual control	Enrollment requires two distinct operators.	Implemented/tested
G3 Freshness	A session identifier is unique, bounded by expiry, and consumed once.	Implemented/tested
G4 Audit integrity	Changes to off-chain evidence are detectable against the keyed on-chain anchor.	Cryptographic assumption
G5 Revocability	A compromised protected template can move to a new epoch and commitment.	Implemented/tested

G6 Data minimization	No raw biometrics, templates, scores, tokens, or clinical identifiers are on-chain.	Design property
NG1 Biometric accuracy	FAR, FRR, TAR, EER, age stability, and PAD efficacy.	Not evaluated
NG2 Clinical efficacy	Reduction in errors, workflow usability, or proof of family relationship.	Not evaluated

4. CRYPTOGRAPHIC CONSTRUCTION AND PROTOCOL

4.1 Notation and data classification

Table 3. Notation summary: symbols grouped by function.

Category	*Symbols*	*Meaning*
Identity	FID, R_s	The clinical family id of this family member is not written to the ledger as a identified entity but a pseudonymised subject reference is generated based on a secret context key.
Template protection	T_M, T_C, e, N_M, N_C	These protected cancelable templates for mother and child are stored off-chain. The following information is stored in the protected record: when these templates expire (rotation epoch for the templates) and a random number for each of the templates (per-template nones).
Template protection	K_ref, K_led, K_e^M, K_e^C	The key(s) for the subject reference(s) to the ledger and/or the corresponding ledger commitment(s) plus the epoch- and role-separated derived key(s) (using HKDF).
Template protection	sid, τ, C_enr, C_req, C_res, π	Random public session identifier and secret off-chain token; enrollment, request, and result commitments; and the versioned policy hash covering algorithms, roles, timeouts, and governance.

The subject reference R_s is derived inside a defined hospital or consortium context; it is never computed directly from a public identifier:

$$R_s = HMAC\text{-}SHA\text{-}256(K_{ref}, context \,||\, FID) \quad (1)$$

Keys for each role and epoch are separated through HKDF [27]:

$$K_{e^r} = HKDF(K_{master}, salt = R_s \,||\, e, info = "BIBIS/template/" \,||\, r) \quad (2)$$

Each protected-template anchor is both keyed and randomized:

$$C_r = HMAC\text{-}SHA\text{-}256(K_{e^r}, T_r \,||\, N_r \,||\, \pi), \; r \in \{M,C\} \quad (3)$$

A single enrollment commitment then conceals the two template anchors while binding the active policy:

$$C_{enr} = HMAC\text{-}SHA\text{-}256(K_{led}, R_s \,||\, e \,||\, C_M \,||\, C_C \,||\, \pi) \quad (4)$$

For message authentication under a secret key HMAC [26] is used. The digest algorithm SHA-256 is specified by FIPS 180-4 [25]. In no case the raw digest of the biometric template is published. This is NOT anonymization, but it keeps on-chain disclosures of protected data to a minimum while an authorized auditor can verify integrity of associated records using the provided Opening Data / and / or Keys.

4.2 Dual-control enrollment

Algorithm 1. Enrollment commitment and ledger activation

```
INPUT: FID, protected templates T_M and T_C, policy π
1  R_s ← HMAC(K_ref, context || FID)
2  e ← 1; sample N_M, N_C ← {0,1}^256
3  derive K_e^M and K_e^C with HKDF domain separation
4  C_M ← HMAC(K_e^M, T_M || N_M || π)
5  C_C ← HMAC(K_e^C, T_C || N_C || π)
6  C_enr ← HMAC(K_led, R_s || e || C_M || C_C || π)
7  registrar → contract.proposeEnrollment(R_s, C_enr, H(π))
8  distinct supervisor verifies off-chain record and calls approveEnrollment(R_s)
9  store {FID,T_M,T_C,N_M,N_C,e,π} encrypted off-chain
OUTPUT: active subject reference R_s at epoch e
```

A subject reference that has already been proposed is not accepted by the contract code, and the account which proposed the reference is stored in the contract code. This account is also not able to approve the reference which it proposed itself. On-chain storage of the contract code consists of R_s, C_enr, H(π), the current epoch, the current state and the event metadata.

4.3 Session-bound verification

Once the maternal matcher succeeds under the proposed policy for matching maternal DNA to a profile, the gateway generates a random session id sid, a secret token τ, a nonce n_s for the new session and an expiry time t_exp for the session. The secret token τ is never sent to the server but is instead passed to the local component which continues to service this workflow. The request commitment is created that binds the maternal decision d_M to the current epoch and does not reveal the actual decision d_M made by the maternal matcher:

$$C\_req = HMAC(K\_s, sid \,||\, R\_s \,||\, e \,||\, t\_exp \,||\, n\_s \,||\, H(\tau) \,||\, d\_M) \quad (5)$$

The request commitment is created by first deriving a key K_s from the gateway's gateway master key. This key is then used to lock the canonical record d_M of the maternal decision. A new ledger session is then created for the random session id sid, the current subject reference R_s, the request commitment C_req and the time until which the secret token τ is valid t_exp. The secret token τ and the session id sid are then handed off to the corresponding local component of the gateway local component to continue the workflow locally. The child decision is then made and - afterwards - the canonical result record d_res is committed to the ledger by the gateway for the session id sid.

$$C\_res = HMAC(K\_s, sid \,||\, C\_req \,||\, d\_M \,||\, d\_C \,||\, PAD\_M \,||\, PAD\_C \,||\, t\_fin) \quad (6)$$

Algorithm 2. Mother-first session-bound verification

INPUT: R_s, current epoch e, maternal and child live samples
1 verify subject state = ACTIVE and load protected templates
2 run maternal capture, PAD, and matcher under policy π
3 if maternal result fails: write local audit record and STOP
4 sample sid, τ, n_s; set t_exp ← now + Δt
5 compute C_req and call openSession(sid,R_s,C_req,t_exp)
6 require presented token τ, same sid, and now ≤ t_exp
7 run child capture, PAD, and matcher under policy π
8 compute C_res over both decisions and policy evidence
9 call finalizeSession(sid,C_res); contract consumes sid
10 erase live samples, τ, and transient derived data
OUTPUT: immutable event anchor plus encrypted detailed record

4.4 Rotation and revocation

Compromise of a protected template, transform seed, or policy triggers a new cancelable representation and a new epoch. The off-chain service computes fresh role keys and nonces. The service then computes C_enr′ and submits a rotateCommitment message. All previous epochs' commitments are included in the audit history for the subject, but are not sufficient to authorize a new session for the current epoch. As opposed to revocation which puts a subject into a finalized state where no more sessions may be created, rotation of a commitment gives the subject cryptographic renewability. However, all previous ledger entries for the subject are still included in the audit history for the subject.

Algorithm 3. Commitment rotation, revocation, and stale-session invalidation

INPUT: active subject reference R_s, current epoch e, action a ∈ {ROTATE, REVOKE}
1 supervisor verifies authorization and reads the current subject state
2 require subject state = ACTIVE
3 if a = ROTATE:
4 create new cancelable representations T_M′ and T_C′ off-chain
5 set e′ ← e + 1; sample fresh nonces N_M′ and N_C′
6 derive role-separated keys K_e′^M and K_e′^C with HKDF
7 compute C_enr′ under the new epoch and policy hash H(π′)
8 call rotateCommitment(R_s,C_enr′,H(π′))
9 atomically activate the new encrypted off-chain record
10 else if a = REVOKE:
11 compute opaque reason commitment C_reason; keep the reason off-chain
12 call revokeSubject(R_s,C_reason)
13 reject finalization of every open session whose epoch/state is now stale
OUTPUT: rotated ACTIVE subject at e′ or terminal REVOKED subject; audit event

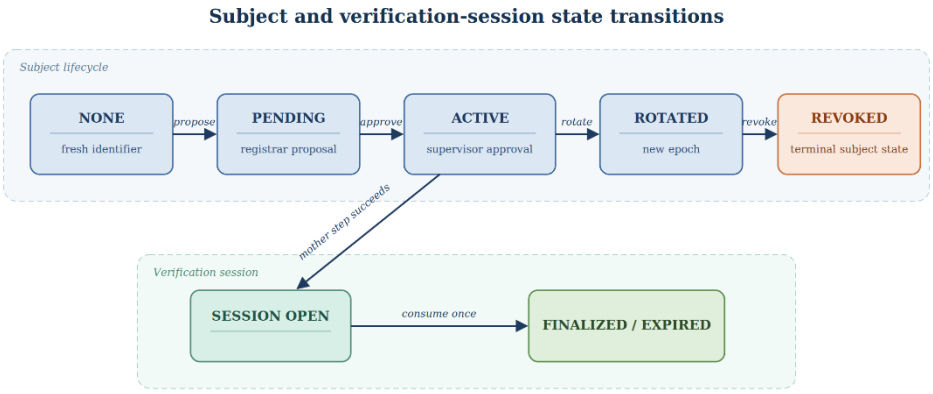

Figure 2. Contract state transitions for subjects and sessions.

5. SMART-CONTRACT IMPLEMENTATION

For the benchmark we have used the reference implementation BIBISAudit.sol of this thesis written in Solidity 0.8.30. The explicit role mappings have been included in the benchmark for better reproducibility of the results. These role mappings however are not recommended to be used in production, instead a reviewed access-control contract should be used in combination with multisignature administration, delayed execution of privileged functions, regular key rotation, monitored permissioning and an independent contract audit. A role-based model as provided by OpenZeppelin is suitable for scenarios where different accounts have separated rights and well-defined roles [30].

Table 4. Contract functions, authorized callers, and security effects.

Function	***Caller***	***Security effect***
setRole	Admin	Grant or remove registrar, supervisor, or gateway role.
proposeEnrollment	Registrar	Create pending opaque enrollment commitment.
approveEnrollment	Supervisor	Activate proposal; approver must differ from proposer.
rotateCommitment	Supervisor	Increment epoch and replace active commitment.
revokeSubject	Supervisor	Move active subject to revoked state.
openSession	Gateway	Create a unique session after maternal success; expiry ≤10 min.
finalizeSession	Gateway	Consume an open, unexpired session and emit result commitment.
expireSession	Any account	Mark an elapsed open session as expired.

5.1 On-chain/off-chain boundary

Table 5. Data placement across storage locations with privacy rationale.

Data	***Location***	***Rationale***
Raw images and live samples	Off-chain/transient	Erase after operation; never submitted to EVM.
Protected templates and nonces	Encrypted off-chain store	Needed for matching, rotation, and authorized audit opening.
Detailed decisions, scores, PAD evidence	Encrypted off-chain store	May be sensitive clinical/security evidence.
FID and demographic data	Clinical system only	Not accepted by contract interface.
R_s, epoch, state	On-chain	Pseudonymous and linkable within context; treat as personal data.
C_enr, C_req, C_res, H(π)	On-chain	Opaque keyed commitments and policy anchors.
τ and gateway keys	HSM/TEE memory	Never logged or placed on-chain.

5.2 Permissioned consensus profile

Our envisioned deployment is to a private Hyperledger Besu network running in QBFT configuration. QBFT is recommended for private enterprise networks, such as this, because it allows others to add blocks once a sufficient number of the total number of predefined validator signatures (in this case, two thirds) have been provided as if signed by that validator. However, the governance value of such a network is heavily dependent on who is left to manage the individual nodes / validators of the network. If all nodes are

managed by the same administrator, then what one is left with is replicated append-only records (i.e. a distributed database) — as opposed to a distributed institutional control scenario. Thus, the items of network permissioning, TLS, validator-key custody, backups, monitoring, and procedures for replacing individual validators are outside the scope of the smart contract's Solidity code and will be part of the broader deployment design for the network.

6. SECURITY ANALYSIS

6.1 Authorization and dual control

In addition to the authorization of msg.sender's role for the privileged transitions of proposing enrollment and of approving enrollment, and of rotating or of revoking a subject, the proposals for enrollment stored in the ledger of Enrollment proposals by subjects for which they are sponsors, also limit the action of any "insider" to that which the "insider" is authorized to perform. Because any of the other accounts can approve of any of the proposals for enrollment for which they are sponsors (including the case where the account that proposed the enrollment for a subject is also one of the other sponsors for that subject's enrollment), it is possible that two "insiders" that are also authorized could collude to perform actions that are not authorized for a single "insider".

6.2 Replay and cross-session substitution

The smart contract tracks four session states: None (the session has not yet been opened), Open (the session is currently active), Finalized (the session has a valid proof), or Expired (the session ended without a valid proof). As before, sid and duplicate finalization of the same protected payload are forbidden. Note that the secret token τ is not stored on chain, which means only τ_hash is stored in the commitment C_req to enrollment. The local offchain gateway's gateway_key computes a key from τ and input to the child step (i.e., verifying biometrics) is therefore τ. In summary, as long as sid and τ are unpredictable, and gateway_key is kept secret, every session will expire on its own, and gateway_key will remain secure. Furthermore, no evidence from a different session than the one currently verified by the gateway (with its session secret token) will be allowed as proof in the smart contract, unless HMAC-SHA-256 is forged or the entire canonical payload is exactly recreated.

6.3 Audit integrity

Insecure storage of enrollment templates aside, under the assumption of HMAC-SHA-256 being existentially unforgeable under a chosen message attack (EUA-CMA) for K_led and K_s, it is infeasible for an attacker to alter the protected enrollment and result messages that are relayed to the contract, and also compute the correct commitments for said messages. In QBFT, for as long as the number of compromised parties does not exceed a pre-determined threshold during verification, the contract can generate and finalize a log of events that are in turn ordered chronologically, and pass said log to the parties for verification. Note that such a log is not sufficient to verify the correctness of a gateway's decision regarding a given biometric authentication event.

6.4 Template compromise and revocation

As T_M and T_C are not even passed to the smart contract during enrollment, when a cancellable template becomes compromised, a completely new set of data T, τ, and Ek can be generated to create a new C that will have a different commitment to the smart contract. However, the previously created anchor R_s for the same smart contract stored on the ledger as part of the corresponding enrollment event cannot be deleted for historical audit purposes. For systems requiring stronger unlinkability between epochs, instead of simply creating new subject in the off-line process, a new subject reference is created and on an authorized off-line basis this new subject reference is mapped to the corresponding historical subjects on the enrollment events that were used to create the respective anchors.

6.5 Residual risks

Table 6. Residual risks with their consequences and required compensating controls.

Residual risk	***Consequence***	***Required control***
Compromised gateway	May submit a false decision with a valid transaction.	HSM/TEE, signed matcher attestations, dual gateway approval, monitoring.
Colluding operators	Can fraudulently enroll a pair.	Independent organizational roles, witness evidence, periodic audit.
Biometric spoof/injection	Occurs before ledger submission.	Evaluate PAD to ISO/IEC 30107-3 and secure capture-to-matcher channel [23,24].
Key compromise	Allows commitment forgery or linkage.	Key hierarchy, HSM, rotation, incident response, split knowledge.
Validator coalition	May censor; excessive compromise threatens safety.	Independent validators, QBFT sizing, governance and monitoring.
Metadata linkage	R_s and event timing remain observable to members.	Context-specific references, batching where feasible, access-limited consortium.

Immutable personal-data anchor	May conflict with retention or erasure requirements.	DPIA, minimal commitments, off-chain deletion, legal basis and retention policy.

7. PROTOTYPE EVALUATION

7.1 Evaluation scope and method

Our set of experiments is aimed at testing the contracts out and the single steps of the authentication protocol. Note that we are not testing any biometric subsystem here. We just fix the way it interacts with the contracts in the Solidity code for the contract binary which we create by compiling the Solidity source code for solc 0.8.30 with 200 optimizer runs. We then deploy it on a local testbed for the JavaScript EVM, i.e. Ganache 7.9.2, by using ethers 6.15.0. The enrollment proposal, the approval of the enrollment proposal, the session open and the session finalization steps have been tested on 100 different subjects. For the commitment rotation and the revoke operation, 25 and 25 different subjects have been tested respectively. For each of the single steps in the authentication protocol we have measured the time to complete the corresponding transaction. Note that here we are talking about a local EVM testbed, i.e. we do not have any remote validators. Also we are not using any QBFT (i.e. the validators always agree in one round). Therefore we are basically measuring the time to complete a transaction on a testbed.

Table 7. Experimental environment and configuration.

Item	***Configuration***
CPU	AMD EPYC 9V74 80-Core Processor
Logical CPUs / memory	9 / 15.93 GiB
OS / architecture	linux 6.12.47 / x64
Runtime	Node v24.14.0; ethers 6.17.0
Compiler	0.8.30+commit.73712a01.Emscripten.clang
EVM	Ganache 7.9.2, in-process, immediate mining
Repetitions	100
Contract bytecode	4157 bytes deployed

7.2 Gas and local latency

Figure 3 provides box plots for the medians of total gas usage and local execution latency for all 6 contract operations. For Session Open the median gas consumption is 141,513 gas with a p95 of 207,083 gas and maximum of 72,044ms latency (35.735ms median latency). The operation with the 2nd highest amount of gas and latency is Enrollment Proposal with a median gas usage of 115,856 gas and 30.37ms latency (p95 of <45ms and max of 39.699ms). In stark contrast to the above two operations the lowest gas consumption and latency was for Subject Revocation with a median gas usage of 31,596 gas and 18.401ms of latency (p95 of <45ms and max of 30.370ms). Note that the above figures are for local EVM executions and do not account for any additional latency added by the network. Furthermore, QBFT transactions finalize after all members have propagated the transaction. Therefore there is not a direct correlation between gas usage and local execution latency as the runtime also incurs additional computational costs not reflected in the gas figures.

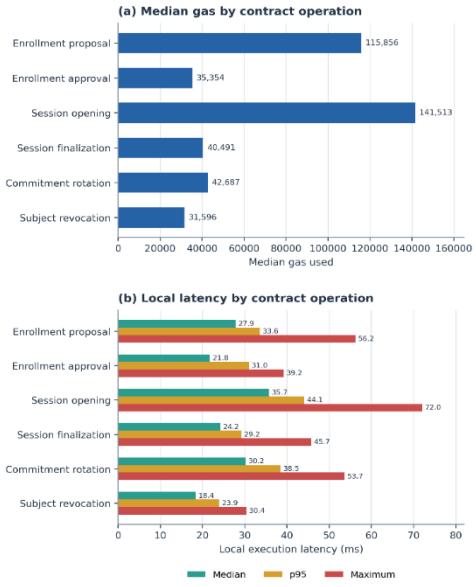

Figure 3. Gas and Local Latency by Contract Operation

7.3 Cryptographic microbenchmark

Here is a simple Node HMAC-SHA-256 microbenchmark for a 256-byte payload for 10,000 iterations: 0.002ms median, p95 0.003ms, max 0.127ms. Note that even in this tiny space for commit construction compared to the rest of the system's costs for capture, matching, I/O for storage, transaction processing, and consensus, this is a completely stand-alone benchmark: no Hardware Security Module (HSM), no remote key-management service.

7.4 Adversarial functional tests

All 20 of the previously described adversarial tests are running either successfully (for the tested paths of execution) or failing as expected (for the untested paths of execution). Tests include those that attempt to perform actions for which the caller is not authorized, self-approval, creation of multiple subjects and/or session IDs, finalization of a session by an unauthorized caller multiple times, finalization of a session after expiration of the session, and many others. The tests for a session opened by a revoked subject not functioning, and for all sessions opened by a committing or revoked subject to be invalidated (including those open at the time of the revocation or end of the current commitment epoch) confirm that the tested code is behaving as intended and following the correct state transition behavior. However, these test cases give no assurance about code outside of the tested paths of execution, or about untestable vulnerabilities. They are not a formal security proof of anything.

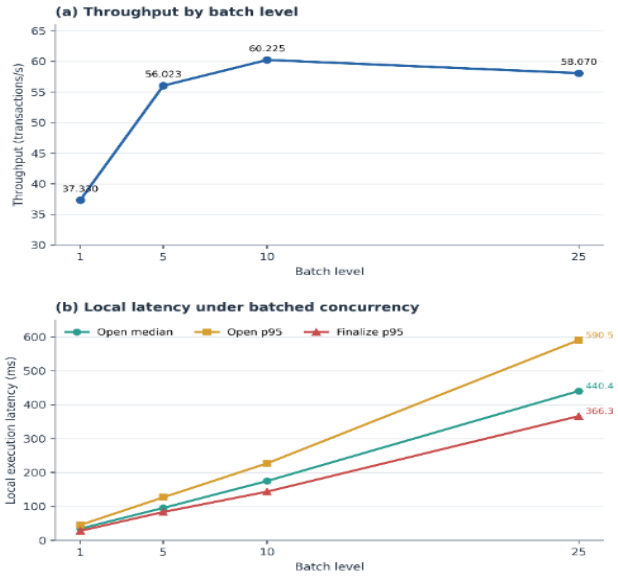

Figure 4. Batched Concurrency Performance by Batch Level

7.5 Batched concurrency and model-based exploration

Algorithm 4. Deterministic randomized reference-model exploration

INPUT: seed σ = 0xB1B15EED, transition count N = 50,000
1 initialize empty subject and session maps in the reference model
2 for i ← 1 to N:
3 select subject, session, and operation using seeded pseudorandom input
4 choose propose, approve, rotate, revoke, open, finalize, expire, or replay
5 evaluate the operation preconditions in the reference state machine
6 if valid then apply exactly one transition; otherwise record rejection
7 check session state ∈ {OPEN, FINALIZED, EXPIRED}
8 check FINALIZED implies state/epoch validation occurred at finalization
9 check no finalized session returns to OPEN and no identifier is reused
10 accumulate accepted/rejected transitions and invariant checks
OUTPUT: transition counts, invariant-check count, and observed violations
NOTE: this explores a JavaScript reference model; it is not Solidity execution or formal proof

Figure 4 provides further information on the local test setup with increasing amounts of concurrency up to 25 for 200 transactions each. The throughput for 200 transactions first increases up to a peak of 60.225 tx/s (level 10) and then slightly decreases towards the saturation point at level 25 to 58.070 tx/s. The median latency for session opening a session for 200 transactions increases from 34.453 ms up to 440.410 ms for 200 transactions. The p95-values for opening and for

finalizing a session are reaching up to 590.496 ms and 366.335 ms respectively. Level 10 seems to be the best trade-off between high throughput and acceptable latency for the setup of local tests. Note that the numbers do not represent the throughput for QBFT nor the finality on the network. The numbers provided stem from local EVM with single process and with immediate mining and thus are subject to queueing.

7.6 Threats to validity

• Internal validity: timing (JavaScript provider, signing, EVM execution, receipt handling) vs. background scheduling (and JavaScript fallback engine).

• External validity: The testbed under single-process control does not take into account e.g. network delay, validator voting, block period, permissioning, TLS, database access, HSM calls and other operational contention.

• Construct validity: the gas and transition tests test for the overhead of the ledger as well as the functionality of the state-machine under test, as opposed to the accurate identification of new born children, the clinical error reduction, the privacy compliance and the PAD of the solution being under test.

• Reproducibility: The report uses a single compiler, a single optimizer configuration, a fixed set of dependencies, a fixed contract code, a fixed set of test scripts, fixed JSON result data and a fixed number of test iterations. All of this can be reproduced from the supplementary distribution package for this report.

8. PRIVACY AND REGULATORY CONSIDERATIONS

The use of personal data for the purpose of unique identification of a natural person in processing activities pursuant to points (a) and (b) of Article 9(2) GDPR shall be treated as requiring special protection under Article 9(1) GDPR. A keyed commitment or pseudonymous subject reference (e.g. a hash value of the subject or a subject handle) is not considered to be anonymous data even if its value is not transparent for casual observation. The European Data Protection Board published final Guidelines on the use of Distributed Ledger/Blockchain technology in 2019 which, among others, outlines under which circumstances the processing of personal data on a distributed ledger is even necessary and that data can only be processed in so far as is strictly necessary. It also outlines which architecture is the most appropriate and who should govern the distributed ledger(s) in question. The main challenge for any such processing of personal data on a distributed ledger will be the balance between the immutability of the processing on the distributed ledger and the data subjects' rights. The proposed design for a contract anchor outlined in this chapter does not, in itself, satisfy all of these requirements but does keep raw data and direct identifiers off-chain.

In addition to the use of the anchor numerous further measures need to be put in place in order to fulfill the obligations of the GDPR. For example a written record of the processing with a statement of the reasons for the processing including a statement of the condition or grounds of lawfulness under Article 6(1) of the GDPR together with a description of the purposes for which the personal data are to be processed and a Data Protection Impact Assessment, details of the roles and responsibilities of any controller and any processor(s) together with a schedule for the storage, processing and disposal of any personal data and what is to happen in the event of a data breach. It will also be necessary to justify the use of a distributed ledger as opposed to an simple append-only log, such as a SQL database, and to provide evidence/proof that all instances of the related biometric data have been deleted from all the ledgers held by all the validators. As stated above authorized deletion of all data (as defined above) stored off-chain by the anchor (used to store the template and any other data used to open a commitment) for a given anchor will render that anchor unusable. However the historical entries for all instances of that anchor will still exist on all the ledgers held by all the validators and therefore, whilst such deletion be described as full erasure it is not a complete erasure of all instances of the related biometric data.

NIST SP 800-63B-4 treats biometric data as an activation factor in multi-factor authentication (as opposed to a secret used in isolation from other secrets). As such, SP 800-63B-4 requires evaluation of controls for retry limits, data replay, and derived biometric data. Additionally, the NIST publication evaluated for remote capture of evidence and unattended capture of evidence requires presentation-attack detection (PAD) for the specific evidence type at the capture layer, the matching layer, and the operational layer of remote capture or unattended capture of evidence. A contract, however, can only include a reference to current evidence and current policy (i.e. a reference to required controls for a current version of evidence or to current policy) and not to the implementation of controls for current evidence or policy.

9. DISCUSSION

This study has a more modest claim than the BIBIS paper. We provide an executable contract that enforces dual control, bounded unique sessions, one-time consumption, periodic rotation and revocation of all matcher's, all within a few hundred EVM instructions. An immutable distributed ledger (IDL) has a few fundamental limitations: An IDL cannot make an unvalidated sensor (e.g. camera, sensor, etc.) or matcher (e.g. face recognition, etc.) reliable. An IDL can only make an authorized assertion reliable by making it difficult to change after finality. But it does not say whether the assertion was correct when it was submitted. Auditing is a matter of keyed commitments, i.e., an auditor retrieves a pseudonymized and audited record of a single processing run (of any length), recomputes the commitment for that record under a set of managed access keys and compares it against the history of the various entries on the ledger. Note that the other members of the set of validator members do not ever get to see the raw templates, even in the form of template

digests. Hence, they cannot match them against any other databases. The only linkable items between the pseudonymized references and their corresponding timestamps on the various databases of the consortium of database owners, is the evidence submitted by the compromised gateways, which is, of course, cryptographically authentic, but factually wrong. In order to address this, there are three alternative solutions with different costs, all of which, however, require increased trust in additional parties: (1) threshold gateway signatures; (2) trusted-execution attestation; (3) zero-knowledge proofs that a given contract (on the blockchain) was processed in accordance with a given policy. All of these are considered future work.

In terms of a more detailed security assessment, we would execute the contract on a multi-node test network of Besu nodes running QBFT consensus. We would test out a number of different administrative domains for the different validator nodes. We would vary a number of different parameters to test the contract's performance: the number of different validator nodes, the block interval, the number of transactions per second, the transaction payload size, the amount of storage used by the contract on disk, and the contract's ability to recover from a failure of one or more of the nodes in the network. In addition, we would also test out a number of different faults, including crash faults and Byzantine faults. We would measure the system in terms of the following criteria: the system's throughput; the system's median and p95 finality time; the amount of storage used by the system on disk; and the system's ability to recover from a failure. As mentioned above, once the contract has been approved under an appropriate protocol for capture and matching, the resulting system can be integrated into that protocol. For now, though, we are not in a position to draw any conclusions as to how the system might perform in terms of recognizing new borns, and this will require a high-resolution data set as well as ethics approval.

10. LIMITATIONS, FUTURE WORK AND

This study did not use any newborn biometric data, human subjects or an FS80H device. Therefore, this study does not test the biometric accuracy or the clinical performance of identifying newborns by using the FS80H device. The performance testing of the system was carried out on a local EVM on a local machine rather than on a multi-validator QBFT network. Hence, the results on latency and throughput obtained in this study do not give any information on the finality of transactions performed on the network. In addition, 20 successful functional tests as well as 50,000 randomly generated state transitions were tested for the rules implemented in this study. However, the results of the above tests do not replace formal verification, thorough vulnerability testing and/or an independent security audit.

The future work will test BIBIS on a multi-node Besu QBFT network to gain insights on latency and on throughput. The rules followed by BIBIS would benefit from a formal security analysis, from a thorough vulnerability testing and from an independent security audit. Finally, reducing the amount of trust that is put on the gateway (e.g. by using more advanced cryptography) is an area that is worth to investigate in the future as well as the performance of the biometric system on the newborn and the clinical tests (if any) – as previously stated all of these are subject to the ethical approval.

BIBIS is a privacy-preserving audit protocol on blockchain. It keeps the biometric data off the blockchain while enabling dual-control biometric enrollment of patients, session-bound verification of patients, commitment rotation and revocation. We provide a reproducible proof-of-concept for the design in the scope of security, but not for biometric accuracy, clinical usability or performance of QBFT on a network.

REFERENCES

1. I. H. Hussien, A. Z. Ablahd, A. O. Najim, and A. A. Shamaileh, "BIBIS: A Blockchain-Biometric Framework for Tamper-Proof Newborn Identification in Healthcare Facilities," National Journal of Antennas and Propagation, vol. 8, no. 1, pp. 191–202, 2025, doi: 10.31838/NJAP/08.01.20.
2. S. C. Wallace, "Newborns Pose Unique Identification Challenges," Pennsylvania Patient Safety Advisory, vol. 13, no. 2, pp. 42–49, 2016.
3. J. E. Gray et al., "Patient misidentification in the neonatal intensive care unit: Quantification of risk," Pediatrics, vol. 117, no. 1, pp. e43–e47, 2006, doi: 10.1542/peds.2005-0291.
4. M. C. Covas, B. Salvatierra, V. Velázquez, and E. R. Alda, "Use of the identification bracelet in the newborn: A safe method?" Archivos Argentinos de Pediatría, vol. 116, no. 1, pp. 72–76, 2018, doi: 10.5546/aap.2018.72.
5. J. J. Engelsma et al., "Infant-ID: Fingerprints for Global Good," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 7, pp. 3543–3559, 2022, doi: 10.1109/TPAMI.2021.3057634.
6. T. Kalisky et al., "Biometric recognition of newborns and young children for vaccinations and health care: A non-randomized prospective clinical trial," Scientific Reports, vol. 12, art. 22520, 2022, doi: 10.1038/s41598-022-25986-6.
7. M. R. A. Sumi, M. H. Imtiaz, and S. Schuckers, "A Longitudinal Study on Fingerprint Recognition in Infants, Toddlers, and Children," in Proc. 46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 1–4, 2024, doi: 10.1109/EMBC53108.2024.10781797.
8. A. K. Jain, S. S. Arora, K. Cao, L. Best-Rowden, and A. Bhatnagar, "Fingerprint Recognition of Young Children," IEEE Transactions on Information

Forensics and Security, vol. 12, no. 7, pp. 1501–1514, 2017, doi: 10.1109/TIFS.2016.2639346.
9. ISO/IEC 24745:2022, Information Security, Cybersecurity and Privacy Protection—Biometric Information Protection, ISO, 2022.
10. N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," IBM Systems Journal, vol. 40, no. 3, pp. 614–634, 2001, doi: 10.1147/sj.403.0614.
11. N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, "Generating Cancelable Fingerprint Templates," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 4, pp. 561–572, 2007, doi: 10.1109/TPAMI.2007.1004.
12. A. K. Jain, K. Nandakumar, and A. Nagar, "Biometric Template Security," EURASIP Journal on Advances in Signal Processing, vol. 2008, art. 579416, 2008, doi: 10.1155/2008/579416.
13. A. Juels and M. Wattenberg, "A Fuzzy Commitment Scheme," in Proc. 6th ACM Conference on Computer and Communications Security (CCS), pp. 28–36, 1999, doi: 10.1145/319709.319714.
14. Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data," in EUROCRYPT 2004, LNCS 3027, pp. 523–540, 2004, doi: 10.1007/978-3-540-24676-3_31.
15. L. Ballard, F. Monrose, and D. P. Lopresti, "Biometric Authentication Revisited: Understanding the Impact of Wolves in Sheep's Clothing," in Proc. USENIX Security, pp. 29–41, 2006.
16. C. C. Agbo, Q. H. Mahmoud, and J. M. Eklund, "Blockchain Technology in Healthcare: A Systematic Review," Healthcare, vol. 7, no. 2, art. 56, 2019, doi: 10.3390/healthcare7020056.
17. T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, "Blockchain distributed ledger technologies for biomedical and health care applications," Journal of the American Medical Informatics Association, vol. 24, no. 6, pp. 1211–1220, 2017, doi: 10.1093/jamia/ocx068.
18. A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using Blockchain for Medical Data Access and Permission Management," in Proc. IEEE OBD, pp. 25–30, 2016, doi: 10.1109/OBD.2016.11.
19. J. K. Adeniyi et al., "A blockchain-based smart healthcare system for data protection," iScience, vol. 28, no. 4, art. 112109, 2025, doi: 10.1016/j.isci.2025.112109.
20. M. S. B. Kasyapa and C. Vanmathi, "Blockchain integration in healthcare: A comprehensive investigation of use cases, performance issues, and mitigation strategies," Frontiers in Digital Health, vol. 6, art. 1359858, 2024, doi: 10.3389/fdgth.2024.1359858.
21. S. Pongnumkul, C. Siripanpornchana, and S. Thajchayapong, "Performance Analysis of Private Blockchain Platforms in Varying Workloads," in Proc. 26th International Conference on Computer Communication and Networks (ICCCN), pp. 1–6, 2017, doi: 10.1109/ICCCN.2017.8038517.
22. Hyperledger Besu, "Configure QBFT Consensus," Private Network Documentation, accessed July 15, 2026.https://docs.besu-eth.org/private-networks/how-to/configure/consensus/qbft
23. NIST, Digital Identity Guidelines: Authentication and Authenticator Management, SP 800-63B-4, 2025, doi: 10.6028/NIST.SP.800-63B-4.
24. ISO/IEC 30107-3:2023, Information Technology—Biometric Presentation Attack Detection—Part 3: Testing and Reporting, ISO, 2023.
25. NIST, Secure Hash Standard, FIPS PUB 180-4, 2015, doi: 10.6028/NIST.FIPS.180-4.
26. H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," RFC 2104, 1997, doi: 10.17487/RFC2104.
27. [27] H. Krawczyk and P. Eronen, "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)," RFC 5869, 2010, doi: 10.17487/RFC5869.
28. European Parliament and Council, Regulation (EU) 2016/679 (General Data Protection Regulation), Official Journal of the European Union, 2016.
29. European Data Protection Board, Guidelines 02/2025 on Processing of Personal Data through Blockchain Technologies, final version, July 7, 2026.
30. OpenZeppelin, "Access Control," OpenZeppelin Contracts 5.x Documentation, accessed July 15, 2026. https://docs.openzeppelin.com/contracts/5.x/access-control
31. M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance," in Proc. OSDI, pp. 173–186, 1999.
32. Futronic Technology Co., "FS80H/FS81H USB2.0 Fingerprint Scanner: Product Specifications," accessed July 15, 2026. https://www.futronic-tech.com/pro-detail.php?pro_id=1543
33. L. F. P. Southier et al., "A Systematic Literature Review on Neonatal Fingerprint Recognition," ACM Computing Surveys, vol. 57, no. 12, art. 312, pp. 1–34, 2025, doi: 10.11
34. M. G. Ayoub, "AI-based ACL exercises recognition system using wearable multi-sensor data fusion," *NTU Journal of Engineering and Technology*, vol. 4, no. 1, Mar. 2025, doi: 10.56286/t9nzzg67.