European Blockchain Services Infrastructure: A Technical and Architectural Analysis of Trust in Public Administration

1. Executive Summary

The modernization of public administration within the European Union is undergoing a paradigmatic shift, moving from centralized, opaque bureaucratic processes toward decentralized, transparent, and mathematically verifiable infrastructures. At the forefront of this transformation is the **European Blockchain Services Infrastructure (EBSI)**, a cross-border network designed to leverage Distributed Ledger Technology (DLT) for the delivery of trusted public services. This report provides an exhaustive analysis of EBSI's role in reshaping governance, with a specific focus on the cryptographic mechanisms that enable immutable audit trails and privacy-preserving citizen participation.

A critical application of this infrastructure is the "Citizen Draw" or sortition (e.g., *Bürgerrat*), a democratic process where citizens are randomly selected to participate in policy deliberations. The legitimacy of such assemblies rests entirely on the public's ability to verify the fairness of the selection process. However, this creates a fundamental tension: the administration must prove the integrity of the draw without exposing the sensitive personally identifiable information (PII) of the citizens involved.

This analysis explores how **Cryptographic Hashing** (*Kryptografisches Hashing*) serves as the foundational primitive for resolving this paradox. It details the mathematical properties of hash functions—determinism, collision resistance, and the avalanche effect—and demonstrates how they are employed in **Commit-Reveal** schemes and **Verifiable Random Functions (VRFs)**. By utilizing EBSI's Notarization and Verifiable Credential services, public administrations can create an unalterable, time-stamped history of the selection process. This ensures that the result is fixed before it is known, preventing retroactive manipulation, while simultaneously shielding the identities of non-selected participants. The transition of EBSI governance to the **EUROPEUM-EDIC** consortium further solidifies the legal and operational permanence of this infrastructure, marking a decisive step toward a digitally sovereign and transparent European public sphere.

2. The Architecture of Trust: EBSI Governance and Infrastructure

The European Blockchain Services Infrastructure is not merely a technical project; it is a political and legal instrument designed to establish a sovereign digital layer for the European Union. Its evolution reflects the growing necessity for cross-border digital services that adhere to European values of privacy, data protection (GDPR), and environmental sustainability.

2.1 Mandate and Strategic Evolution

Launched in 2018, EBSI began as a joint initiative between the European Commission and the European Blockchain Partnership (EBP), a coalition comprising all EU Member States, plus Norway and Liechtenstein. The mandate was clear: to harness the potential of blockchain technology to create trusted digital audit trails, automate compliance checks, and prove data integrity across borders. Unlike the permissionless blockchains that dominate the cryptocurrency sector (e.g., Bitcoin, Ethereum), EBSI is a **permissioned** network. This means that while the ledger may be publicly verifiable, the validation of blocks and the maintenance of the infrastructure are restricted to trusted public entities, ensuring alignment with EU regulations such as eIDAS.
The strategic importance of EBSI lies in its ability to prevent the reliance on non-European platforms for critical public infrastructure. It is funded by the **Digital Europe Programme (DIGITAL)**, which focuses on bringing digital technology to businesses, citizens, and public administrations, underscoring its role as a public good rather than a commercial venture.

2.2 The Transition to EUROPEUM-EDIC

A pivotal development in the maturity of EBSI is the transition of its governance structure. Initially coordinated by the Commission and the EBP, governance is currently shifting to a new legal entity: **EUROPEUM-EDIC** (European Digital Infrastructure Consortium), established in May 2024.

Feature	Pre-2024 Governance	EUROPEUM-EDIC Governance (2024 onwards)
Legal Status	Loose partnership (EBP) & Commission coordination	Formal Legal Entity (EDIC) under EU law
Membership	All Member States + EEA (Signatories)	Specific Consortium Members (Belgium, Italy, Poland, etc.)
Mandate	Pilot and Definition	Production and Operation
Operational Control	DG CNECT (Owner) / DG DIGIT (Provider)	EUROPEUM-EDIC (Autonomous Management)
Funding	EU Grants / Commission Budget	Member State Contributions + EU Funding

This transition, expected to conclude in 2025, represents the "graduation" of EBSI from a pilot project to a permanent production infrastructure. The founding members of EUROPEUM-EDIC include Belgium, Croatia, Cyprus, Greece, Italy, Luxembourg, Poland, Portugal, Romania, Slovenia, and Spain, with other countries able to join as members or observers. This structure ensures that the infrastructure remains under the sovereign control of Member States, preventing any single entity from monopolizing the network.

2.3 The Node Network and Operational Roles

The physical infrastructure of EBSI relies on a distributed network of nodes hosted by public administrations and authorized agencies across Europe. This distribution is the first line of defense against censorship and data manipulation. The network classifies participants into distinct roles with specific Service Level Agreements (SLAs) :

- **Validator Nodes:** These are the critical anchors of the network. They are responsible for

validating transactions, proposing new blocks, and participating in the consensus protocol. To become a Validator Node, an organization must undergo a rigorous endorsement process by EUROPEUM-EDIC, comply with strict security policies, and guarantee high availability.
- **Regular Nodes:** These nodes maintain a full copy of the ledger, allowing them to verify the state of the blockchain and serve read requests. They do not participate in block creation but are essential for the network's resilience and data availability.
- **API and Service Layers:** Above the node layer, EBSI exposes specific APIs (e.g., Timestamping, Notarization) that allow external applications—such as a student information system or a citizen draw platform—to interact with the blockchain without running a node themselves.

3. Core Service Families in Public Administration

EBSI is designed as a modular platform, supporting a variety of "Use Cases" that address specific administrative challenges. These use cases are not isolated applications but interconnected services that can be combined to build complex workflows.

3.1 Notarization and Audit Trails

The **Notarization** service is the cornerstone of EBSI's ability to create immutable audit trails. It allows a public administration or a citizen to create a trusted digital record of a document or event. By submitting a "digital fingerprint" (hash) of a file to the ledger, the system generates a proof that the data existed in a specific state at a specific time.
This service is domain-agnostic. It is used for:
- **Document Authenticity:** Proving that a PDF (e.g., a tax receipt or a court order) has not been altered since its issuance.
- **Process Verification:** Logging the steps of a procurement process or a citizen sortition to ensure compliance with procedural rules.
- **Timestamping:** Providing a legally recognized timestamp that is independent of the local server's clock, derived instead from the consensus of the entire blockchain network.

3.2 Verifiable Credentials (VCs) and Digital Identity

The **Verifiable Credentials** framework represents a shift toward "Self-Sovereign Identity" (SSI). In traditional systems, identity data is stored in central government databases (silos). In the EBSI model, based on W3C standards, the data is issued to the citizen, who stores it in a digital wallet.
- **The Trusted Issuer:** An entity (e.g., a university or municipality) accredited to issue credentials. Their accreditation is recorded on the blockchain (in the Trusted Issuers Registry), allowing anyone to verify their authority.
- **The Holder:** The citizen who receives the credential (e.g., a proof of residency) and stores it in their wallet. They fully control this data.
- **The Trusted Verifier:** An entity (e.g., the organizer of a citizen draw) that requests proof of eligibility.

Crucially, VCs support **Selective Disclosure**. A citizen can prove they are "over 18" and "resident of Berlin" without revealing their exact birth date or street address. This capability is

essential for privacy-preserving administrative processes.

3.3 Diplomas and Educational Credentialing

One of the earliest and most successful pilots of EBSI involves the cross-border verification of educational diplomas. Universities like TU Graz and WU Wien have implemented systems where diplomas are issued as VCs. This eliminates the need for expensive and slow manual verification of paper transcripts, reducing fraud and administrative overhead. The success of this use case provides the operational template for more complex scenarios like the citizen draw: if the infrastructure can securely verify a degree, it can securely verify residency eligibility for a civic lottery.

4. Cryptographic Hashing: The Mathematical Engine of Immutability

To understand how EBSI creates an "immutable audit trail" for a citizen draw, one must move beyond the abstract concept of "blockchain" and examine the specific cryptographic primitive that powers it: **Cryptographic Hashing** (*Kryptografisches Hashing*).

4.1 Defining the Cryptographic Hash Function

A cryptographic hash function is a mathematical algorithm that maps data of arbitrary size (the "input" or "message") to a bit string of a fixed size (the "hash" or "digest"). Common algorithms include SHA-256 (used in Bitcoin and many EBSI components), which produces a 256-bit output regardless of whether the input is a single word or the entire Encyclopedia Britannica. Unlike standard database indexing hashes, cryptographic hash functions must satisfy four rigorous properties to be considered secure for public administration:

4.1.1 Determinism

A given input must *always* generate the exact same hash output. There is no element of randomness in the hashing process itself.

- **Implication:** If a citizen's eligibility record is hashed today, and verified by an auditor ten years later, the hash must match exactly. This guarantees consistency in long-term audit trails. If the hash changes, it is absolute proof that the record has been altered.

4.1.2 The Avalanche Effect

This property dictates that a microscopic change in the input should result in a massive, unpredictable change in the output. Ideally, flipping a single bit in the input data should flip approximately 50% of the bits in the output hash.

- **Administrative Relevance:** This prevents "approximate" forgery. A corrupt official cannot change a citizen's name from "Schmidt" to "Schmitt" without completely altering the resulting hash. The discrepancy would be immediately obvious to the network, as the new hash would bear no resemblance to the original notarized hash.

4.1.3 Pre-Image Resistance (One-Way Function)

It must be computationally infeasible to reverse-engineer the original input from the hash value.

- **Privacy Implication:** This is the bedrock of privacy-preserving verification. A public observer can see the hash of a citizen's ID on the ledger but cannot derive the ID number or name from that hash. It allows the administration to publish the "list of participants" in a format that is verifiable (by checking against the original data) but readable only to those who possess the original data.

4.1.4 Collision Resistance

It must be computationally infeasible to find two distinct inputs that produce the same hash output.

- **Security Implication:** This ensures uniqueness. No two citizens can accidentally share the same "digital fingerprint." More importantly, it prevents a "collision attack," where a malicious actor generates a fake identity that produces the same hash as a legitimate selected citizen, attempting to swap them out. While theoretical collisions exist (due to the pigeonhole principle), the probability of finding one with SHA-256 is so infinitesimally small (requiring 2^{128} operations) that it is physically impossible with current or foreseeable computing power.

4.2 Hashing vs. Anonymization: The Necessity of Salting

While pre-image resistance prevents reversing a hash, it does not prevent "brute-force" or "rainbow table" attacks if the input space is small.

- **The Vulnerability:** If an administration simply hashes a Social Security Number (SSN), an attacker can generate the hashes for all possible SSNs (which are finite and structured) and compare them to the ledger to de-anonymize the citizens.
- **The Solution: Cryptographic Salt/Nonce.** To secure the audit trail, the administration adds a random, high-entropy string (a "Salt" or "Nonce") to the citizen's data before hashing. Because the attacker does not know the random salt, they cannot pre-compute the hashes. This technique allows the administration to commit to the citizen pool publicly without exposing the identities of the individuals.

5. Building the Immutable Audit Trail

The "Immutable Audit Trail" is not a single file stored on a server; it is a chronological chain of cryptographically linked events stored on the EBSI network. This structure ensures that once a result is recorded, it cannot be changed without destroying the integrity of the entire system.

5.1 The Chaining Mechanism

Blockchain achieves immutability by grouping transactions (such as the registration of a citizen pool) into blocks. Each block contains the cryptographic hash of the *previous* block.

1. **Block N** contains a hash of **Block N-1**.
2. If an attacker attempts to delete a name from **Block N-1**, the content of that block

changes.
3. Consequently, the hash of **Block N-1** changes (due to the Avalanche Effect).
4. This breaks the link to **Block N**, which contains the *old* hash of N-1. The validation nodes will reject Block N as invalid.
5. To successfully alter the record, the attacker would have to re-compute the hashes for Block N, Block N+1, and every subsequent block in the chain.

5.2 Distributed Consensus as a Time Anchor

Re-computing hashes is computationally expensive, but theoretically possible for a powerful entity. However, EBSI adds a layer of **Distributed Consensus**. The ledger is maintained by dozens of nodes across different Member States. To alter the history, a corrupt actor would need to convince the majority of these sovereign nodes to accept the altered chain.

- **Practical Impossibility:** In the context of EUROPEUM-EDIC, where nodes are operated by the governments of Belgium, Italy, Poland, etc., colluding to rewrite history is politically and operationally impossible.
- **Timestamping:** Each block is timestamped. This provides a "Proof of Existence" for the citizen draw data at a specific moment in time. This prevents the "stuffing the ballot box" attack, where names are added to the pool *after* the selection criteria are known.

6. Citizen Sortition (Bürgerrat): The Privacy-Verification Paradox

Sortition—the selection of public officials or decision-makers by lottery—has deep historical roots, most notably in ancient Athens, where it was considered the truest form of democracy. Today, it is experiencing a renaissance in Europe through "Citizens' Assemblies" (e.g., *Bürgerrat* in Germany, *Convention Citoyenne* in France) to address complex issues like climate change.

6.1 The Challenge of Trust in Modern Sortition

In traditional modern implementations, the selection process is often a "black box." A government agency runs a script on a private server to select 100 citizens from the population registry. The public must trust:

1. The code was fair (unbiased).
2. The input list was complete (no exclusions).
3. The result was not altered (no "preferred" candidates swapped in). Research suggests that this opacity contributes to a decline in trust in public institutions. Critics may argue that the assembly was rigged to produce a specific policy outcome.

6.2 The Requirements for a Trusted Draw

To restore trust, the process must move from a "Black Box" to a "Glass Box." However, this creates a conflict between two fundamental requirements :

1. **Public Verification:** Anyone should be able to mathematically prove that the selection was random and the winners came from the eligible pool.
2. **Individual Privacy:** The list of all eligible citizens (the pool) and the identities of those *not* selected must remain private to protect against harassment and comply with GDPR. Even

the winners' identities might need to be protected until they consent to participate. EBSI solves this paradox by using **Cryptographic Hashing** to decouple the *verification of the process* from the *visibility of the data*.

7. Technical Workflow: Creating the Unalterable Draw

The following section details the step-by-step technical workflow for conducting a privacy-preserving, verifiable citizen draw using EBSI. This workflow integrates **Verifiable Credentials**, **Commit-Reveal Schemes**, and **Verifiable Random Functions**.

Phase 1: Identity and Eligibility (The Pool Formation)

Before the draw can occur, the pool of eligible candidates must be established.

- **Step 1:** Citizens utilize their EBSI-compliant Digital Wallets to prove eligibility. They present a **Verifiable Presentation** containing claims (e.g., "Resident of Region X", "Age > 18") derived from their official IDs.
- **Step 2:** The administration validates these credentials against the **Trusted Issuers Registry** (TIR) on EBSI to ensure they were issued by legitimate authorities.
- **Step 3:** For each eligible citizen, the administration generates a unique, random **Salt** (S).
- **Step 4:** The system computes the **Commitment Hash** (C) for each citizen: The use of the Salt ensures that even if the list of C_i is public, no one can brute-force the DIDs to identify the citizens.

Phase 2: The On-Chain Commitment (The "Sealed Envelope")

The administration must now "lock in" the pool to prevent future manipulation.

- **Step 1:** The list of all Commitment Hashes (C_1, C_2,... C_N) is aggregated. For large populations, this is done using a **Merkle Tree**, where hashes are paired and hashed iteratively until a single **Merkle Root** remains.
- **Step 2:** This Merkle Root is submitted to the **EBSI Notarization Service** via a transaction.
- **Step 3:** EBSI returns a transaction receipt containing the block number and timestamp.
    - **Crucial Outcome:** The pool is now immutable. The administration cannot add or remove a citizen without changing the Merkle Root, which would not match the notarized record on the blockchain. This is the "Pre-commitment".

Phase 3: The Random Selection (Verifiable Randomness)

Standard Random Number Generators (RNGs) are opaque and can be manipulated (e.g., run multiple times until a desired result is found). EBSI architectures utilize **Verifiable Random Functions (VRFs)** to guarantee fairness.

- **The Mechanism:** A VRF takes a **Seed** and a **Private Key** to produce a **Random Value** (R) and a **Proof** (\pi).
- **The Seed:** To ensure the administration cannot predict the outcome, the Seed is defined as the hash of a future block on the EBSI chain (e.g., "The hash of the first block mined after 12:00 PM next Friday"). Since no one can predict a block hash in advance, the seed is truly random and impartial.

- **The Selection:** The Random Value R is used to select an index from the sorted list of Commitments (e.g., Index = R \mod N).
- **Verification:** Any observer can use the **Public Key** and the **Proof** to verify that R was generated correctly. This proves that the administration did not "cherry-pick" the random number.

Phase 4: The Reveal and Public Audit

Once the winners are selected, the administration contacts them privately using the link between the hash and their internal database. If a citizen accepts, the result is published for audit.

- **The Reveal:** The administration publishes the **Citizen DID** and the **Salt** for the winner.
- **The Public Check:** An independent auditor (e.g., a journalist or NGO) performs the following check:
    1. Take the published DID and Salt.
    2. Compute H = \text{SHA-256}(\text{DID} + \text{Salt}).
    3. Verify that H matches the Commitment Hash stored at the winning index in the Merkle Tree notarized in Phase 2.
- **Conclusion:** If the match is valid, it proves mathematically that:
    1. The citizen was in the original pool (Audit Trail).
    2. The citizen was selected by the unbiased random number (VRF).
    3. The administration did not alter the result (Immutability).

8. Why the Result Cannot Be Changed (Resilience Analysis)

The user's query specifically emphasizes understanding *why* the result is unchangeable after the fact. This resilience is not due to a single feature but the interlocking of three layers of cryptographic and systemic security.

8.1 Layer 1: The Pre-Commitment (The "Temporal Handcuff")

The most common form of lottery fraud is altering the pool *after* the selection criteria are known, or altering the selection criteria *after* the pool is known.

- **EBSI Solution:** The "Commitment" phase (Phase 2) separates the definition of the pool from the generation of the randomness. The pool is cryptographically locked (hashed and notarized) at Time T_1. The Random Seed is determined by a blockchain event at Time T_2.
- **Why it works:** Because T_1 < T_2, the administration is forced to commit to the candidates *before* they know who will win. Conversely, once the random seed at T_2 is revealed, they cannot go back and change the pool at T_1 because the blockchain is immutable.

8.2 Layer 2: Deterministic Verification (The "Mathematical Handcuff")

In a manual draw, an official could claim "Alice won" when actually "Bob" was drawn, relying on the opacity of the process.

- **EBSI Solution:** The selection process is deterministic. The VRF output R points to a specific index in the list. That index contains a specific hash.
- **Why it works:** If the administration tries to swap the winner for someone else, the hash of the new person will not match the hash at the selected index. Since the list of hashes is public (on-chain), any observer would immediately detect the fraud. The administration cannot "fake" the math because the hash function is collision-resistant.

8.3 Layer 3: The Distributed Ledger (The "Consensus Handcuff")

Even if an administration tampered with their local server, they cannot tamper with the EBSI ledger.

- **EBSI Solution:** The data is replicated across 29+ nodes in different countries.
- **Why it works:** To retroactively change the winner, the administration would need to rewrite the blockchain history (a "51% attack"). In the EUROPEUM-EDIC governance model, this would require conspiring with the governments of Italy, Belgium, Poland, and others to simultaneously corrupt their nodes. This provides a level of security that a single national database can never achieve.

9. Comparative Analysis: EBSI vs. Traditional Methods

Feature	Traditional Government Draw	EBSI-Enabled Trusted Draw
Data Storage	Centralized SQL Database	Distributed Ledger (Blockchain)
Transparency	"Black Box" (Process hidden)	"Glass Box" (Process verifiable)
Privacy	High (but reliant on admin honesty)	High (Cryptographically guaranteed via Hashing/Salting)
Auditability	Internal logs (mutable by admin)	Immutable Audit Trail (Publicly verifiable)
Randomness	System RNG (Opaque)	Verifiable Random Function (VRF) (Provable)
Trust Model	"Trust us, we are the government"	"Don't trust, verify" (Mathematical certainty)

10. Future Outlook and Strategic Implications

The operationalization of these capabilities through EBSI marks a significant maturation of European digital infrastructure. With the **Early Adopters Programme** already piloting projects across 15 countries and 25 distinct use cases , the transition from theory to practice is well underway.

10.1 Expansion of Use Cases

While this report focused on citizen sortition, the same "Notarization + Hashing" architecture is being applied to:

- **Supply Chain Traceability:** Projects like **TRACE4EU** use EBSI to timestamp seafood catch certificates, creating an audit trail from "net to plate" that prevents food fraud.
- **Asylum Process Management:** Facilitating the cross-border management of asylum

demands by proving the existence and timing of applications without exposing refugee data.

10.2 The Role of eIDAS 2.0

The upcoming widespread adoption of **EU Digital Identity Wallets** (EUDI Wallets) under the eIDAS 2.0 regulation will act as a force multiplier for EBSI. It will provide every citizen with the toolset (Keys, VCs) needed to participate in these cryptographic workflows natively, without needing to understand the underlying blockchain complexity.

10.3 Conclusion

The European Blockchain Services Infrastructure provides the necessary technical foundation to rebuild trust in public administration. By leveraging **Cryptographic Hashing** to create immutable audit trails, and **Verifiable Random Functions** to ensure fair selection, EBSI allows public institutions to conduct sensitive democratic processes like citizen draws with absolute transparency. This architecture proves that in the digital age, privacy and public verification are not mutually exclusive; through the clever application of cryptography, they can be mutually reinforcing. As governance transitions to the **EUROPEUM-EDIC**, this infrastructure is set to become a permanent pillar of European digital sovereignty, ensuring that the "truth" of public records is guaranteed not by the authority of a seal, but by the certainty of mathematics.

Quellenangaben

1. About us - EBSI - - European Commission, https://ec.europa.eu/digital-building-blocks/sites/spaces/EBSI/pages/474513483/About+us 2. European blockchain services infrastructure | Shaping Europe's digital future, https://digital-strategy.ec.europa.eu/en/policies/european-blockchain-services-infrastructure 3. A European Blockchain technology for the use in accreditation and certification, https://european-accreditation.org/wp-content/uploads/2024/03/EBSI-presentation-to-EAs_Feb-2024.pdf 4. Timestamp API | EBSI hub, https://hub.ebsi.eu/apis/pilot/timestamp 5. Piloting Domibus integration with CEF EBSI (blockchain): Building and Configuring - European Commission, https://ec.europa.eu/digital-building-blocks/sites/download/attachments/323290750/%28ISA2%29.%28eDelivery%29.%28Piloting%20Domibus%20integration%20with%20CEF%20EBSI%29.%28Building%20and%20Configuring%29.%28v1.0%29.pdf?api=v2 6. European Blockchain Service Infrastructure - Trust Alliance New Zealand, https://trustalliance.co.nz/wp-content/uploads/2024/05/EBSI-European-Blockchain-Service-Infrastructure-Explained-Daniel-Du-Seuil.pdf 7. The European Blockchain Services Infrastructure is coming, and the ECA has a role to play, https://medium.com/ecajournal/the-european-blockchain-services-infrastructure-is-coming-and-the-eca-has-a-role-to-play-68b53395c788 8. Piloting Domibus integration with CEF EBSI (blockchain): Implementation Overview - European Commission, https://ec.europa.eu/digital-building-blocks/sites/download/attachments/323290750/%28ISA2%29.%28eDelivery%29.%28Piloting%20Domibus%20integration%20with%20CEF%20EBSI%29.%28Implementation%20Overview%29.%28v1.0%29.pdf?api=v2 9. EBSI Verifiable Credentials - European Commission, https://ec.europa.eu/digital-building-blocks/sites/spaces/EBSI/pages/600343491/EBSI+Verifiable

+Credentials 10. Explained Series - EBSI - - European Commission, https://ec.europa.eu/digital-building-blocks/sites/spaces/EBSI/pages/659621351/Explained+Series 11. Issuer Trust Model | EBSI hub, https://hub.ebsi.eu/vc-framework/trust-model/issuer-trust-model-v3 12. Public Administrations Interoperability - EBSI - - European Commission, https://ec.europa.eu/digital-building-blocks/sites/display/EBSI/Public+Administrations+Interoperability 13. Examples of JSON-LD contexts and Verifiable Credentials used by EBSI4Austria - GitHub, https://github.com/danubetech/ebsi4austria-examples 14. Verification of Education Credentials on European Blockchain Services Infrastructure (EBSI): Action Research in a Cross-Border Use Case between Belgium and Italy - MDPI, https://www.mdpi.com/2504-2289/7/2/79 15. Everything about Cryptographic hash functions | by Pruthvik Reddy | Nerd For Tech, https://medium.com/nerd-for-tech/everything-about-cryptographic-hash-functions-e2cd892e2a87 16. Cryptographic hash function - Wikipedia, https://en.wikipedia.org/wiki/Cryptographic_hash_function 17. Cryptography Hash Functions - GeeksforGeeks, https://www.geeksforgeeks.org/competitive-programming/cryptography-hash-functions/ 18. Hash functions: Theory, attacks, and applications - Microsoft, https://www.microsoft.com/en-us/research/wp-content/uploads/2005/11/hash_survey.pdf 19. No, hashing still doesn't make your data anonymous - Federal Trade Commission, https://www.ftc.gov/policy/advocacy-research/tech-at-ftc/2024/07/no-hashing-still-doesnt-make-your-data-anonymous 20. Cryptographic hash functions | IBM Quantum Learning, https://quantum.cloud.ibm.com/learning/courses/quantum-safe-cryptography/cryptographic-hash-functions 21. Commit-Reveal scheme in Solidity. What is it? | by Srinivas Joshi | Coinmonks | Medium, https://medium.com/coinmonks/commit-reveal-scheme-in-solidity-c06eba4091bb 22. Sheet1 - Hugging Face, https://huggingface.co/datasets/shahadalkhalifa/Crypto_Whitepaper_Labeled/resolve/main/Clean_crypto_text.xlsx 23. What Is Blockchain? | IBM, https://www.ibm.com/think/topics/blockchain 24. The Ultimate Guide to Immutable Audit Trails - HubiFi, https://www.hubifi.com/blog/immutable-audit-log-basics 25. Sortition - Wikipedia, https://en.wikipedia.org/wiki/Sortition 26. Marta POBLET | Professor | Professor | RMIT University, Melbourne | RMIT | Graduate School of Business and Law | Research profile - ResearchGate, https://www.researchgate.net/profile/Marta-Poblet 27. Rebuilding European Democracy: Resistance and Renewal in an Illiberal Age 9780755639717, 9780755639755, 9780755639748 - DOKUMEN.PUB, https://dokumen.pub/rebuilding-european-democracy-resistance-and-renewal-in-an-illiberal-age-9780755639717-9780755639755-9780755639748.html 28. Secure MPC-Sortition: Consolidating Innovations in Democracy and Cryptography - WouterMaas.com, https://www.woutermaas.com/philosophy/Secure%20MPC-Sortition%20-%20Consolidating%20Innovations%20in%20Democracy%20and%20Cryptography%20-%20Wouter%20Maas.pdf 29. FTSOv2: more data feeds and faster updates to the FTSO - Flare Network, https://flare.network/wp-content/uploads/FTSOv2-White-Paper.pdf 30. Request For Proposal - Government Navigator, https://media.governmentnavigator.com/media/bid/1749650220_RFP-1602123430_2025_Security_Operations_Center_as_a_Service_SOCaaS.pdf 31. Verifiable Random Function (VRF) - Explained - Chainlink, https://chain.link/education-hub/verifiable-random-function-vrf 32. Verifiable random function - Wikipedia, https://en.wikipedia.org/wiki/Verifiable_random_function 33. hash - How might one create a system of election by lot (sortition,

https://crypto.stackexchange.com/questions/63858/how-might-one-create-a-system-of-election-by-lot-sortition-that-is-both-secure 34. Algorand Releases First Open-Source Code: Verifiable Random Function - Medium, https://medium.com/algorand/algorand-releases-first-open-source-code-of-verifiable-random-function-93c2960abd61 35. Anonymizing Commit and Reveal Transactions in Decentralized OracleSolutions | by Gorka Irazoqui Apecechea | The Witnet Oracle Blog | Medium, https://medium.com/witnet/anonymizing-commit-and-reveal-transactions-in-decentralized-oracle-solutions-e61067833cd9 36. Immutable Audit Trails with Blockchain | RecordsKeeper.AI, https://www.recordskeeper.ai/immutable-audit-trails/ 37. TRACE4EU Seafood Tracing - EBSI - - European Commission, https://ec.europa.eu/digital-building-blocks/sites/spaces/EBSI/pages/716149132/TRACE4EU+Seafood+Tracing