ZK-Disclosure: Privacy-Preserving Information Disclosure for Digital Evidence with C2PA and zk-SNARKs

Johnny Marinho, Eryk Schiller, Arthur Debauge, Noria Foukia
Institute of Industrial and IT Engineering (InTECH)
School of Landscape, Engineering and Architecture (HEPIA)
University of Applied Sciences and Arts of Western Switzerland (HES-SO)
johnny.marinhodamota@master.hes-so.ch
{eryk.schiller, arthur.debauge, noria.foukia}@hes-so.ch

***Abstract*—This paper presents a framework that integrates the Coalition for Content Provenance and Authenticity (C2PA) standard with Zero-Knowledge Proofs (ZKPs), specifically the Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs), to enable verifiable yet privacy-preserving authentication of digital images. Using the ZoKrates toolkit, the system derives a non-revealing fingerprint from the image, generates a succinct proof of integrity, and embeds this proof into C2PA-compliant metadata without exposing the underlying content. The proof can be verified locally or on the Ethereum blockchain using a Groth16 smart contract verifier, providing decentralized and auditably transparent validation. This capability allows journalists, victims, and legal professionals to attest to the existence and integrity of sensitive evidence while deferring its disclosure. Experimental results show that proof verification is highly efficient, requiring approximately 0.01 s, and that the entire workflow is reproducible within containerized environments. The proposed integration of zk-SNARKs with C2PA establishes a practical foundation for secure digital provenance, privacy-preserving evidence management, and strengthened trust in digital media ecosystems.**

1. Introduction

The increasing availability of powerful image-editing tools and the proliferation of synthetically generated media have raised profound concerns about the authenticity and trustworthiness of digital evidence. Frameworks such as the Coalition for Content Provenance and Authenticity (C2PA) address part of this challenge by defining an open standard for embedding cryptographically signed provenance metadata, such as the creator's identity, device, location, and subsequent transformations, directly into media files. These signatures enable a verifier to confirm the declared origin and history of a file, thereby helping to counter disinformation and forgeries [1]. However, provenance alone is insufficient in contexts where the content itself must remain confidential. In judicial, investigative, or ethically sensitive cases, stakeholders may need to attest to the existence and integrity of evidence without disclosing its content until it is legally or strategically required. This tension between the need for verifiable authenticity and the requirement for confidentiality remains unresolved by current provenance standards.

To address this gap, we propose a novel framework that integrates C2PA with Zero-Knowledge Proofs (ZKPs), specifically the the Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARKs). A zero-knowledge proof is a cryptographic protocol that allows one party (the prover) to convince another party (the verifier) that a particular statement is true, such as "this file is authentic and untampered", without revealing any additional information about the underlying data. Among the various ZKP constructions, zk-SNARKs are particularly suitable for practical deployments: they produce very short proofs that can be verified efficiently without repeated interaction between the prover and verifier. These properties make zk-SNARKs appropriate for embedding proofs within C2PA-compliant manifests and for enabling automated, large-scale verification.

In our framework, the system derives a non-revealing fingerprint from the protected content, generates a zk-SNARK proof attesting to its integrity and consistency with the committed fingerprint, and embeds this proof within the C2PA manifest of the file. The proof can be verified in a decentralized manner on a public blockchain, ensuring transparent validation of provenance without disclosing the protected content itself. By unifying provenance and privacy-preserving proof mechanisms, the proposed framework offers a robust foundation for secure, verifiable, and ethically responsible digital evidence management, supporting applications ranging from investigative journalism to judicial proceedings and thereby contributing to the restoration of trust in digital media ecosystems.

The remainder of this paper is organized as follows. Section 2 reviews related work on digital evidence management, provenance standards, and zero-knowledge proofs. Section 3 presents the proposed framework that integrates C2PA with zk-SNARKs for privacy-preserving yet verifiable content authenticity. Section 4 illustrates the applicability of the framework through representative use cases. Section 5 reports preliminary experimental results evaluating the fea-

sibility and efficiency of the approach. Finally, Section 6 concludes the paper and outlines directions for future research.

2. Related Work

Researchers have long been concerned with the reliability of digital evidence. [2] proposed that evidence should be encapsulated as self-describing digital objects that include metadata, provenance records, and cryptographically verifiable chains of custody. This approach focuses on preserving authenticity and interpretability, so that future courts can confirm the integrity of evidence even when no living witness is available to testify to its origin. Notably, [3] proposes a robust and privacy-compliant system to securely timestamp digital evidence using a chronological ledger, preventing backdating and enabling integrity verification of a digital file.

[4] argued that ZKPs can reconcile the evidentiary needs of courts with the privacy rights of individuals. They propose that ZKPs allow an investigator or a defendant to prove specific properties of digital evidence, such as its lawful collection or the presence of a particular feature, without revealing the evidence itself, thereby addressing long-standing tensions between privacy and due process. This represents a conceptual transition from merely preserving evidence for long-term verification to enabling privacy-preserving, controlled disclosure in ongoing legal proceedings.

Building on this line of thought, [5] extended the use of ZKPs to protect not only the contents of evidence but also the identity of the individuals providing it. Their framework allows whistleblowers to submit evidence to a private blockchain-based ledger, while supplying ZKPs that convince authorities of the evidence's authenticity and integrity without revealing either the sensitive content or the whistleblower's identity. In this way, the focus of cryptographic research on digital evidence has evolved from ensuring long-term integrity to enabling selective, privacy-preserving disclosure of both content and source.

The proposed system builds upon recent advances in two complementary technologies: the C2PA standard and zk-SNARKs. C2PA [6], supported by major industry players such as Adobe, Microsoft, and Intel, establishes content authenticity by embedding cryptographically signed provenance metadata into digital files, thereby ensuring provenance, integrity, and transparency across the entire lifecycle of a file. zk-SNARKs enable cryptographic proofs of knowledge without disclosing the underlying data [7]. The open-source toolkit ZoKrates [8] facilitates the generation and verification of zk-SNARK proofs on the Ethereum blockchain [9].

Prior work using these technologies has typically focused either on content traceability (C2PA) or on confidentiality (ZKPs). The present project proposes a novel integration of both, creating a secure and verifiable digital proof system that ensures provenance and integrity while also preserving confidentiality.

3. Proposed Solution

This work introduces a cryptographic framework that enables privacy-preserving yet verifiable proof of authenticity for sensitive digital images, particularly suited for evidentiary and judicial applications. The system integrates zk-SNARKs with the C2PA standard. The objective is to allow third parties to verify that an original image existed and has remained unaltered, without revealing the image itself until disclosure is legally required.

3.1. System Design

The system's design, illustrated schematically in Figure 1, transforms a sensitive input image into a *proxy image* that contains no visual content but carries a C2PA-compliant manifest with a zk-SNARK proof. This proxy can be stored, transmitted, or submitted as sealed evidence. When disclosure is required, the verifier can check that the revealed original image corresponds to the previously committed input by verifying the proof embedded in the proxy image. This architecture provides a clear separation between the management of sensitive data and the public verification of its authenticity.

3.2. Implementation Architecture

The implementation is organized as a set of containerized components that isolate the primary cryptographic and provenance operations. One container runs the zk-SNARK engine together with the trusted setup material and the proving key required for generating proofs. A second container integrates with a C2PA-compliant tool to embed the evidence into the metadata of the proxy image. Verification may be carried out either locally, using the verification key generated during the setup phase, or remotely via a smart contract deployed on the Ethereum test network.

A crucial design choice in the demonstrator is the definition of the public and private parameters in the zero-knowledge circuit. From the input image, a representation of the image integrity anchor, such as the Merkle root, is treated as the private witness in the proof circuit. A cryptographic hash of the Merkle root, computed with SHA-256, is exposed as the public input and recorded in the proxy image's C2PA manifest. The zk-SNARK proof therefore certifies that the prover knows a 64-byte block whose SHA-256 digest equals the published value, while revealing neither the block itself nor any other part of the original image. Strengthening the construction to with a Merkle-root [10] commitment extends this guarantee from the slice to the entire image.

Once generated, the proof is serialized in JavaScript Object Notation (JSON) format and incorporated into the C2PA manifest of a blank proxy image. This embedding allows the proof to be carried within a standardized provenance structure and accessed by any compliant tool without exposing sensitive pixels of the original content. By combining privacy-preserving proofs with provenance metadata, the system bridges the gap between traditional integrity-oriented

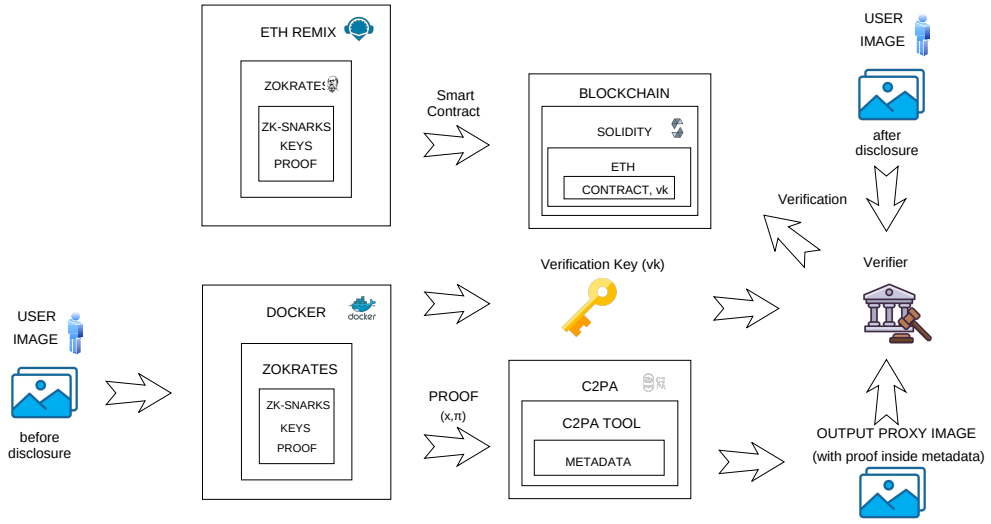

Figure 1: Architecture of the privacy-preserving image provenance system. zk-SNARK proofs are generated and embedded in a proxy image via C2PA metadata, allowing local or blockchain-based verification without revealing the original image.

provenance approaches and the need for confidential information disclosure.

To guarantee transparent and tamper-resistant verification, a Solidity smart contract implementing the Groth16 verification algorithm [11] is deployed on the Ethereum test network. The contract validates the proof against the public hash and requires no access to the original image. This on-chain verification provides decentralization and auditability, while the local verification option enables use in settings where blockchain access is impractical.

3.3. Workflow and Usage Scenario

In a typical usage scenario, an individual such as a journalist, a victim, or a legal professional processes a sensitive image through the system to produce a proxy image that carries the embedded proof. The proxy image can be stored, exchanged, or submitted as sealed evidence without revealing the protected content. At a later stage, for instance during judicial proceedings, the original image is disclosed and its integrity is confirmed by checking that the zk-SNARK proof embedded in the proxy image validates against the hash of the disclosed image. This workflow ensures that the existence and integrity of the evidence are guaranteed from the outset while preserving confidentiality until disclosure becomes appropriate.

3.4. Proof Specification

We model the zk-SNARK as a Groth16 scheme [11] for a circuit C that enforces a binary relation $R(x, w) = 1$ over a public input x and a private witness w.

Let $H : \{0,1\}^* \to \{0,1\}^{256}$ denote SHA–256.

Let $I \in \{0,1\}^L$ be the original image file and let $\mathrm{Slice}_{64}(I)$ denote a fixed 64-byte block derived from I by a publicly known rule (e.g., first or last 64 bytes).

The circuit is:

$$R(x, w) = 1 \iff H(w) = x, \quad \text{with } x \in \{0,1\}^{256},\ w \in \{0,1\}^{512}.$$

In the implementation, w encodes $\mathrm{Slice}_{64}(I)$ and x is recorded in the C2PA manifest as a public digest. The Groth16 algorithms are instantiated as usual: $\mathsf{Setup}(C) \to (pk, vk)$, $\mathsf{Prove}(pk, x, w) \to \pi$, $\mathsf{Verify}_{\mathrm{Groth16}}(vk, x, \pi) \in \{\mathsf{accept}, \mathsf{reject}\}$. Zero-knowledge and succinctness follow from Groth16 and the pairing-based assumptions on the chosen curve.

We adopt the standard adversarial model where the prover may be malicious but polynomial-time, the verifier is

honest, and the hash function H is collision- and preimage-resistant. Soundness is computational and holds provided the Common Reference String (CRS) generated by Setup is produced via a secure multi-party ceremony and the toxic waste is destroyed. Under these assumptions, any π convincing an honest verifier for input x implies the existence of a 64-byte witness w such that $H(w) = x$. Nevertheless, the binding property achieved by R is limited to the 64-byte witness. More precisely, the proof certifies existence of some $w \in \{0,1\}^{512}$ with $H(w) = x$, which could lead to an undetectable tampering attack if we do not consider all the image (cf. Section 3.5)

3.5. Undetectable Tampering Attack

As we keep only the 64 bytes of an image to build the proof, an adversary $\mathcal{A}$ could rebuild another image with the same 64 bytes leading to the same proof, as explain below:

3.5.1. Adversarial Model. An adversary $\mathcal{A}$ has access to:

- The original image I
- The public hash

$$x = H(Slice_{64}(I)) = H(w)$$

- The valid proof π such that $\text{Verify}_{\text{Groth16}}(x, \pi) = 1$

$\mathcal{A}$ aims to produce $I' \neq I$ such that $\text{Verify}_{\text{Groth16}}(x, \pi) = 1$ still holds.

3.5.2. Insecurity of 64-Byte Binding. There exists an efficient adversary $\mathcal{A}$ that outputs $I' \neq I$ with $\text{Verify}_{\text{Groth16}}(x, \pi) = 1$.

Let $\mathcal{A}$ construct:

$$I' = Slice_{64}(I) \parallel (I[64:] \oplus M)$$

where $M \neq 0^{n-64}$ is any non-zero modification of the remaining bytes of the picture avoiding pixel erasure. Then:

1) $Slice_{64}(I') = I'[0{:}64] = Slice_{64}(I)$ *i.e.* $w' = w$
2) $H(w') = H(w) = x$
3) The witness w = $Slice_{64}(I)$ still satisfies $R_{\text{slice}}(w, x) = 1$
4) $\text{Verify}_{\text{Groth16}}(x, \pi) = 1$ (same proof)
5) But $I' \neq I$ and semantic content is altered

Thus, integrity is not preserved. This attack is undetectable and requires no cryptographic break.

3.5.3. Merkle-Tree Witness. To obtain small-witness proofs that bind to the *entire* image, we replace the simple slice-hash relation R with a Merkle-bound relation. The process is the same as the one we adopted in Horodocs tree that we implemented in [3]. We partition the image I into $m = \lceil L/64 \rceil$ blocks $B_0, \ldots, B_{m-1}$ of 64 bytes each and compute for every block a leaf hash $h_i = H(B_i)$. A binary Merkle tree is then built over these leaves using the same hash function H, yielding a single root $\mathsf{MerkleRoot}(h_0, \ldots, h_{m-1})$ that commits to the entire image. The public value recorded in the C2PA manifest is the hash of this root,

$$x = H\big(\mathsf{MerkleRoot}(h_0, \ldots, h_{m-1})\big),$$

so that the hash function is effectively applied twice: once to derive the Merkle leaves from the chunks and again to compress the root into a single published commitment. The strengthened relation is therefore

$$R_{\text{merkle}}(x, w) = 1 \iff H\big(\mathsf{MerkleRoot}(h_0, \ldots, h_{m-1})\big) = x.$$

Under the collision-resistance of H, any modification of any block of the image changes x except with negligible probability, thus binding the proof to the *entire* image while keeping the private witness minimal.

4. Use Case: Legal Context

In a legal setting, the ability to provide reliable, timestamped, and tamper-proof evidence is a fundamental requirement for maintaining the chain of custody and upholding the fairness of judicial proceedings. In many situations, however, the premature disclosure of the evidence itself may compromise the strategy of either the prosecution or the defense. A typical example involves a party that possesses a sensitive image, such as a compromising photograph or a strategically significant document, that they wish to declare and certify at a given date, while refraining from revealing its content until it becomes procedurally appropriate, for instance, during a hearing or upon judicial order.

The use of ZKPs, and in particular zk-SNARKs, provides a technically rigorous means to address this tension between verifiability and confidentiality. A zk-SNARK allows one party (the prover) to convince another party (the verifier) that they have a specific digital item, and that this item has not been modified since its commitment, without revealing the item itself. In this way, two requirements that are often seen as conflicting in legal evidence management, namely the need for verifiable availability of evidence and the need to preserve the confidentiality of its content, are reconciled.

The system developed in this work builds on the capabilities of zk-SNARK technology in combination with the C2PA standard. A cryptographic fingerprint of the evidence, obtained as the SHA-256 hash of the Merkle root, is recorded as a public parameter. The zk-SNARK proof attests that the private witness is consistent with the committed public hash, thereby demonstrating both possession and integrity without revealing the actual image. This proof is embedded within the C2PA manifest of a blank proxy image, which can then be submitted as sealed evidence. The manifest thus serves as a tamper-evident container for the proof, providing a standardized and interoperable means of preserving provenance and authenticity.

5. Experimental Evaluation

The experimental evaluation demonstrates that the strengthened Merkle-bound relation R_{merkle} can be efficiently integrated into a zk-SNARK workflow for binding cryptographic commitments to full-resolution digital images while preserving confidentiality. In contrast to the initial slice-hash prototype, which only authenticated a fixed 64-byte region, the new system commits to the entire image, regardless of its size or content. This allows C2PA manifests to embed a cryptographic commitment that is both privacy-preserving and globally image-binding.

5.1. Setup

We generated four synthetic, high-entropy Portable Network Graphics (PNG) images of increasing resolution, each filled with random Red-Green-Blue (RGB) pixel values. Their file sizes were:

- 64×64 px: 12 420 bytes,
- 128×128 px: 49 353 bytes,
- 256×256 px: 197 173 bytes,
- 512×512 px: 788 089 bytes.

For each image, we compute a Merkle tree over 64-byte blocks of the file. Each leaf is defined as $h_i = H(B_i)$ using SHA-256, and the binary Merkle tree is built by hashing pairs of children, with the last node duplicated when a level has an odd number of hashes. The resulting root MerkleRoot commits to the entire image. During proof generation, the prover supplies this Merkle root (i.e., MerkleRoot) in packed format as a private witness, while the public value x is the SHA-256 hash (i.e., H) of the root. The ZoKrates circuit then verifies the relation

$$H(\text{merkleRoot}) = x,$$

thereby anchoring the entire image through a single short proof while revealing neither image content nor any intermediate Merkle structure.

All proofs were generated and verified inside a Docker[1] container running `zokrates/zokrates:latest` (pulled on 15 November 2025). The host machine was equipped with an Intel(R) Core(TM) Ultra 9 185H CPU, 32 GB of RAM, and a 512 GB SSD. For each image size, we generated and verified the proof five times, which allowed us to compute mean values and standard deviations for both operations. Importantly, Merkle tree construction occurs entirely outside the zk-SNARK system and is therefore not included in the proof-generation or verification timings. This means that the measured proof times reflect only the zk-SNARK computation, not the preprocessing needed to compute the Merkle root.

1. https://www.docker.com/

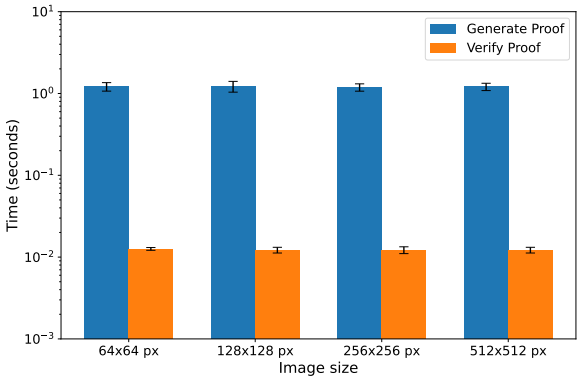

Figure 2: Mean proof generation and verification times.

5.2. Quantitative Results

Across all four image sizes and all twenty proof-generation runs in total, the average proof generation time was

$$\mu_{\text{gen}} = 1.22 \text{ s} \quad \text{with standard deviation} \quad \sigma_{\text{gen}} = 0.13 \text{ s}.$$

Proof verification was significantly faster and essentially independent of the image size, with an average of

$$\mu_{\text{ver}} = 0.012 \text{ s} \quad \text{and} \quad \sigma_{\text{ver}} = 0.001 \text{ s}.$$

Figure 2 visualizes these results. The plot shows, for each resolution (64×64, 128×128, 256×256, and 512×512 pixels), the mean wall-clock time required to generate a proof and to verify a proof, together with error bars representing one standard deviation over the five runs per configuration. The y-axis is logarithmic and ranges from 1 ms to 2 s, which makes the large separation between generation and verification times clearly visible: proof generation consistently takes on the order of one second, whereas verification remains around 10–15 ms across all image sizes. Because the Merkle tree construction is performed externally and not inside the circuit, both proof generation and proof verification behave essentially as constant-time operations with respect to the input image size.

5.3. Discussion

These results confirm that the Merkle-root-based commitment integrates seamlessly into a zk-SNARK workflow without compromising practicality. While Merkle tree construction scales linearly with the size of the image file, that computation is deliberately performed outside the zk-SNARK circuit. The circuit itself only verifies a single SHA-256 relation between the Merkle root and its public hash. Consequently, the cost of zk-SNARK proof generation and verification remains almost constant across all image sizes, as the complexity of the circuit does not depend on the depth or shape of the Merkle tree. At the same time, confidentiality of the underlying image is fully preserved. Neither the pixel data nor any Merkle path is revealed to the verifier; only the

public hash x and the succinct proof are disclosed. This construction therefore provides a cryptographically sound and operationally efficient way to bind a C2PA manifest to an entire image while supporting privacy-preserving, delayed-disclosure scenarios such as legal evidence handling.

6. Conclusion

This paper has presented a complete exploration of integrating zero-knowledge proof systems into the C2PA provenance standard for privacy-preserving authentication of digital images. Our work comprised the design, formal analysis, and implementation of two image authentication schemes, followed by an empirical evaluation of their performance.

We first specified a slice-based authentication mechanism in which the prover supplies a zk-SNARK proof attesting that the first 64 bytes of an image match a published SHA-256 digest. While this construction provides a minimal-witness proof and demonstrates the feasibility of embedding zk-SNARK attestations inside C2PA manifests, we formally identified an undetectable tampering attack: an adversary can arbitrarily modify all bytes outside the committed slice while preserving the public hash, the witness, and the validity of the proof. This analysis establishes that slice-based relations do not offer cryptographic binding to the full image under the adversarial model considered.

To address this limitation, we designed and implemented a Merkle-tree authentication scheme that binds the proof to the entire image. By partitioning the file into 64-byte blocks, hashing each block, and constructing a binary Merkle tree, the prover commits to all blocks through a single root. The public value embedded into the C2PA manifest is the hash of this Merkle root, while the private witness consists only of the packed Merkle root supplied to the circuit. Under the collision resistance of SHA-256, any modification of any block in the image changes the Merkle root except with negligible probability, thereby providing full-image integrity with a small and constant-size witness. This resolves the security gap of the slice-based approach while maintaining succinctness.

Both constructions, i.e., the slice-hash and the Merkle-bound relations, were fully implemented in ZoKrates. The evaluation focused on the Merkle-based variant, as it alone satisfies the integrity requirements of our adversarial model. Using randomly generated PNG images ranging from 64×64 to 512×512 pixels and running five trials per image, we measured the performance of zk-SNARK proof generation and verification on an Intel(R) Core(TM) Ultra 9 185H system within a Dockerized ZoKrates environment. Results show that proof generation requires on average 1.22 s with a standard deviation of 0.13 s, while verification remains extremely lightweight, averaging 0.012 s with a standard deviation of 0.001 s. Because Merkle tree construction occurs outside the circuit and only a single SHA-256 hash is verified inside the zk-SNARK, both proof generation and verification behave effectively as constant-time operations with respect to image size.

These findings demonstrate that the combination of C2PA manifests and zk-SNARK proofs can provide strong cryptographic guarantees for full-image authenticity while preserving confidentiality. The approach further supports practical workflows such as the secure registration of evidence, where images can be committed to at capture time without prematurely revealing their content. This capability is highly relevant in domains such as legal proceedings, journalism, and disinformation mitigation, where trustworthy but privacy-preserving digital provenance is essential.

Future work includes extending proxy images to support selective masking, such as hiding a child's face, while still enabling verification that the unmasked content matches the original image. Achieving this will require extending the zero-knowledge circuits so that the proof can attest to the correctness of both the mask and the underlying unmasked regions. We also plan to reinforce the commitment layer by moving beyond simple hash-based commitments and exploring stronger cryptographic commitments, such as, but not limited to, Pedersen commitments [12], which provide improved hiding and binding guarantees. Finally, we aim to develop a mobile application that automates Merkle-tree construction, proof generation, and C2PA manifest embedding, and to investigate blockchain interoperability. These directions further enhance usability, trust, and the robustness of privacy-preserving provenance systems.

Acknowledgement

We would like to thank the Institute of Industrial and IT Engineering (InTECH) and the School of Engineering, Architecture and Landscape of Geneva (HEPIA), and University of Applied Sciences and Arts Western Switzerland (HES-SO) for supporting this work and enabling the publication of its results.

References

[1] J. Sedlmeir, A. Rieger, T. Roth, and G. Fridgen, "Battling disinformation with cryptography," *Nature Machine Intelligence*, vol. 5, pp. 1056–1057, Oct. 2023.

[2] H. M. Gladney, "Trustworthy 100-year digital objects: Evidence after every witness is dead," *ACM Trans. Inf. Syst.*, vol. 22, p. 406–436, July 2004.

[3] D. Jaquet-Chiffelle, L. Pfeiffer, L. Brocard, E. Benoist, and N. Foukia, "Horodocs: A scalable, sustainable, robust and privacy compliant system to securely timestamp digital evidence and documents," *Forensic Science International: Digital Investigation*, vol. 53, no. 3, 2025. Available at: https://www.sciencedirect.com/science/article/pii/S2666281725000526.

[4] I. Majdoub and K. Atmani, *Privacy Paradigm Shift: Zero Knowledge Proofs in Criminal e-Evidence Collection*, pp. 151–175. Cham: Springer Nature Switzerland, 2025.

[5] B. Mbimbi, D. Murray, and M. Wilson, "Preserving whistleblower anonymity through zero-knowledge proofs and private blockchain: A secure digital evidence management framework," *Blockchains*, vol. 3, no. 2, 2025.

[6] C2PA Consortium, "C2PA specification 2.2," 2024. Available at: https://spec.c2pa.org/specifications/specifications/2.2/index.html.

[7] A. Nitulescu, "A survey of zk-SNARKs," tech. rep., ENS Paris, 2019. Available at: https://www.di.ens.fr/ nitulesc/files/Survey-SNARKs.pdf.

[8] ZoKrates Team, "ZoKrates documentation," 2024. Getting Started. Available at: https://zokrates.github.io/gettingstarted.html.

[9] ZoKrates Contributors, "ZoKrates: Toolbox for zk-SNARKs on Ethereum," 2024. Available at: https://github.com/Zokrates/ZoKrates.

[10] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Advances in Cryptology — CRYPTO '87* (C. Pomerance, ed.), (Berlin, Heidelberg), pp. 369–378, Springer Berlin Heidelberg, 1988.

[11] J. Groth, "On the size of pairing-based non-interactive arguments," in *EUROCRYPT: Theory and Applications of Cryptographic Techniques*, Springer, 2016. Available at: https://eprint.iacr.org/2016/260.pdf.

[12] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Advances in Cryptology – CRYPTO '91*, vol. 576 of *Lecture Notes in Computer Science*, pp. 129–140, Springer, 1991.