Dataset Protocol and Evidence Preservation for Detecting Cyber Incident Screenshot Manipulation: Data Structure, Tamper Recipes, and Chain of Custody

Muhamad Haikal Abdussalam[1*], Rahes Restu Sadewa[2]
[1]Master's Program in Informatics Engineering, Postgraduate Program, Pamulang University, South Tangerang
[2]Information Systems Study Program, Faculty of Computer Science, Pamulang University, South Tangerang
[1.2]Jalan Raya Puspiptek No. 46, Buaran, Kecamatan Serpong, Kota Tangerang Selatan, Banten 15310
Email: muhamad2haikal99@gmail.com[1], rahesrestu2901@gmail.com[2]

Screenshots are widely used in cybersecurity and digital forensics as preliminary evidence of incidents such as phishing pages, website defacement, and SIEM/IDS dashboard captures; however, their ease of manipulation through overlay, cropping, splicing, copy-move, and recompression undermines evidentiary reliability and complicates investigation triage. This study aims to design a standardized dataset protocol for cyber incident screenshots that strengthens digital evidence preservation and supports reproducible analysis workflows. The proposed protocol defines acquisition documentation, SHA-256 hashing, and chain-of-custody recording, alongside a structured folder hierarchy, evidence naming conventions, labeling schemes for binary and multi-class classification tasks, acquisition metadata, documented manipulation procedures via a tamper_recipe, and case_id-based data splitting to prevent leakage of derived manipulations across dataset partitions. As an implementation reference for triage modules, a lightweight analytical framework using GLCM texture features and classical classifiers is specified to demonstrate practical integration without positioning the work as a performance benchmark. The resulting outputs include a comprehensive, auditable protocol specification, standardized metadata and labeling templates, and a reproducible data management workflow tailored for cyber incident screenshots. The study concludes that formalizing acquisition, provenance, and splitting practices improves evidentiary integrity, reduces contamination risk across data partitions, and enhances the utility of screenshots for early-stage forensic triage while remaining compatible with resource-constrained operational settings.

Keywords: Digital Forensics; Screenshot Evidence; Passive Image Forensics; SHA-256; Dataset Protocol; Evidence Preservation.

Corresponding Author:
Muhamad Haikal Abdussalam
Master's Program in Informatics Engineering, Postgraduate Program, Pamulang University, South Tangerang
Jalan Raya Puspiptek No. 46, Buaran, Kecamatan Serpong, Kota Tangerang Selatan, Banten 15310
muhamad2haikal99@gmail.com

1. Introduction

In cybersecurity practice, incident evidence conventionally consists of structured and semi-structured artifacts such as system logs, network traffic captures, endpoint artifacts, and audit records generated by security monitoring tools. These data sources are considered primary evidence in Digital Forensics and Incident Response (DFIR) because they are machine-generated, time-stamped, and can be correlated across systems to reconstruct attack timelines[1][2]. However, in many real-world scenarios particularly during the early phase of incident reporting or within organizations with limited forensic readiness such comprehensive artifacts may not yet be available or easily accessible. In such conditions, screenshots are frequently used as preliminary or even primary evidence because they provide an immediate visual snapshot of an observed anomaly or security incident.

Screenshots are valued for their ability to capture contextual information that may not be explicitly recorded in logs, such as visible URLs, phishing page layouts, defacement messages, warning banners, or alert notifications displayed on SIEM or IDS dashboards. This visual context can be crucial for rapid triage, enabling security teams to make time-sensitive decisions regarding containment, escalation, or communication with stakeholders. As noted by OSAC[3], visual artifacts often play a critical role in early situational awareness, especially when incident responders must act before full forensic acquisition is completed [4]. Nevertheless, the same accessibility and convenience that make screenshots attractive also introduce significant risks from a forensic perspective.

The principal challenge lies in the fact that screenshots are raster image files that can be easily modified using widely available image editing tools[5][6]. Common manipulations include cropping to remove contextual indicators (e.g., browser address bars), overlaying text or icons, splicing elements from different images, copy-move operations, and recompression artifacts introduced intentionally or through messaging platforms. While cryptographic hashing (e.g., SHA-256) is a standard mechanism to ensure fixity after acquisition, it only verifies that a file has not changed since it was hashed; it does not provide assurance that the image content itself was authentic at the time of capture[1][7]. As a result, a screenshot may remain hash-consistent yet still represent manipulated or misleading content.

This limitation underscores the need for an approach that goes beyond post-acquisition integrity checks. Authenticity validation must be treated as a process-level requirement that spans acquisition, documentation, preservation, and analysis, rather than being relegated solely to image forensic algorithms. Prior studies in image forensics have demonstrated that the reliability of manipulation detection is highly dependent on the availability of well-defined ground truth, consistent labeling, and transparent documentation of how manipulated samples are generated[8][9]. Without these elements, analytical results may be difficult to interpret, compare, or reproduce across studies.

In response to these challenges, this study contributes by proposing a replicable dataset and evidence preservation protocol specifically designed for cyber incident screenshots. The protocol formalizes multiple aspects of dataset construction and evidence handling, including structured folder hierarchies and evidence naming conventions, clearly defined labeling schemes, and comprehensive acquisition metadata. A key component is the use of a documented *tamper_recipe* to record how each manipulated screenshot is generated, thereby ensuring transparency and traceability of derived samples. Furthermore, the protocol recommends *case_id*-based data splitting to prevent leakage of manipulation derivatives across training, validation, and test sets an issue that has been shown to bias performance estimates in forensic machine learning studies [10].

Finally, to demonstrate practical usability without overclaiming performance, this study positions a lightweight analytical framework such as Gray-Level Co-occurrence Matrix (GLCM) features combined with classical classifiers as a reference implementation for triage modules within DFIR workflows. This analytical component is not intended to establish state-of-the-art detection accuracy, but rather to illustrate how the proposed dataset and preservation protocol can support reproducible experimentation and fair comparison of different forensic approaches. By integrating evidence preservation principles with dataset design, this work aims to bridge the gap between DFIR best practices and empirical research on screenshot manipulation detection, thereby enhancing both scientific rigor and operational relevance.

2. Literature Review

Visual-Based Cyber Incident Evidence and Its Implications

In the early stages of incident response, visual evidence in the form of screenshots is often used to accelerate triage because it contains contextual information that is not always captured in logs, such as URLs, the appearance of phishing or defacement pages, or alerts displayed on SIEM/IDS dashboards. However, this informational value is accompanied by risk: screenshots can be manipulated using simple software, meaning that "visual evidence" may potentially mislead operational decisions as well as forensic reporting. Therefore, authenticity validation must be understood as a process requirement, rather than merely an image analysis stage.

In digital forensic practice, preservation standards emphasize acquisition documentation, hash computation (e.g., SHA-256) for fixity verification, access control, and chain-of-custody records for every transfer or access to artifacts[1][3]. This framework ensures the integrity of artifacts after acquisition, but it does not automatically guarantee that the screenshot content was free from manipulation at the outset. This gap necessitates a dataset protocol that integrates visual evidence, acquisition metadata, and an audit trail so that screenshot research and validation can be replicated and held accountable.

Passive Image Forensics and Manipulation Categories

Passive image forensics aims to identify indications of editing without relying on watermarks or digital signatures[11]. Survey literature commonly groups manipulations such as splicing, copy-move, retouching, and compression or resampling artifacts into several major approaches: (i) handcrafted features, (ii) noise or sensor trace analysis, (iii) compression-based methods, and (iv) deep learning techniques[8][9][10][12]. This diversity of approaches indicates that manipulation detection is not a single, unified problem, but rather one that depends on data characteristics, threat scenarios, and the quality of ground truth.

In the context of a protocol or dataset paper, the key lesson from this body of work is that the success of forensic studies is strongly influenced by dataset quality, including consistent definitions of manipulation classes, thorough documentation of manipulation generation processes, and comprehensive metadata. Thus, the most fundamental contribution to strengthening screenshot forensic research lies not only in algorithm selection, but in designing dataset protocols that ensure label traceability and auditability across researchers.

Texture Feature Extraction for Forgery Detection

Texture-based approaches are widely used because manipulations often disrupt local regularities and the relationships between pixel intensities. The Gray-Level Co-occurrence Matrix (GLCM), which estimates the frequency of co-occurring intensity pairs at specific distances and orientations, has inspired Haralick features such as contrast, correlation, energy, and homogeneity[13]. Other studies have employed GLCM-based texture features or combined them with classical classifiers to detect splicing or forgery, while Local Binary Patterns (LBP) are often positioned as micro-texture features that complement texture information[14][15][16].

However, for a protocol or dataset paper, the primary role of discussing texture features is as a reference implementation: providing a lightweight, interpretable, and easily replicable starting point for dataset users. In other words, the emphasis of this section is not on identifying the best model, but on ensuring that the dataset contains sufficient information auditable labels, manipulation records, and proper data splits so that various approaches (texture-based or deep learning) can be fairly evaluated on the same data source.

Screenshot-Specific Challenges in Cybersecurity Cases

Unlike photographic images, screenshots are dominated by UI elements such as repeated text, icons, interface boundaries, and homogeneous color blocks. These characteristics have implications for dataset design: certain manipulations (e.g., cropping) may function more as context alteration (removing address bars or indicators) rather than producing strong texture artifacts, making ground truth definition and manipulation documentation critical. In addition, screenshots are often shared through messaging applications that trigger recompression or resolution reduction; these processes can obscure manipulation artifacts and introduce variations that are non-manipulative but still affect image statistics.

Consequently, screenshot datasets for cybersecurity should include: (i) source and device metadata (e.g., platform, resolution, DPI), (ii) post-processing scenarios that represent real-world distributions (recompression or resizing), and (iii) data split designs that prevent derivative leakage from a single case into different sets. Without an explicit dataset protocol, studies may produce biased performance estimates and be difficult to replicate.

Digital Evidence Preservation and Chain of Custody

Digital evidence preservation requires integrity, traceability, and auditability from acquisition through reporting. Forensic guidelines and standards recommend recording acquisition metadata, hashing for fixity checks, access control, and chain-of-custody documentation for every transfer, duplication, or analysis activity[1][3]. In the context of screenshots, these requirements are even more critical because image files can be easily copied and distributed without control, leading to the loss of provenance and handling history. From a research perspective, these preservation principles should be translated into dataset artifacts: standardized acquisition logs, hash records, and chain-of-custody logs. In this way, a dataset becomes not merely a collection of labeled images, but an auditable evidence package. This rationale underpins why the present study positions dataset protocols and chain-of-custody templates as core components of its methodological contribution.

Research Gap and Study Positioning

The literature shows that image manipulation detection has been extensively studied in photographic images and generic forgery datasets using feature-based, statistical, and learning-based approaches[8][9][10]. However, screenshots used as cyber incident evidence exhibit distinct characteristics: dominance of UI and text, homogeneous regions, and distortions due to recompression when distributed via messaging applications. Moreover, although DFIR emphasizes evidence preservation and traceability through hashing and chain-of-custody practices[1][3][7], the integration of these principles into dataset design such as data structures, metadata, manipulation recipes, and proper splits remains rarely discussed explicitly.

Based on this gap, the present article is positioned as a protocol or dataset paper that provides: (i) folder structures and evidence naming conventions, (ii) labeling schemes, (iii) acquisition metadata, (iv) documented manipulation generation procedures through a *tamper_recipe*, (v) recommendations for *case_id*-based data splitting to prevent cross-split derivative leakage, and (vi) a chain-of-custody template as an auditability artifact. The analytical pipeline (e.g., GLCM-based methods) is retained solely as a reference implementation to validate dataset usability, rather than as a primary performance claim.

3. Method

This section describes the dataset protocol and evidence preservation framework for cyber incident screenshots. The discussion covers the definition of the evidence unit, folder structures and naming

conventions, labeling schemes and acquisition metadata, the application of SHA-256 hashing and chain-of-custody records, documentation of image manipulations through a tamper_recipe, and a case_id-based data splitting strategy. As a complement, a lightweight analytical reference implementation is included to demonstrate the practical usability of the dataset, without making any claims regarding performance.

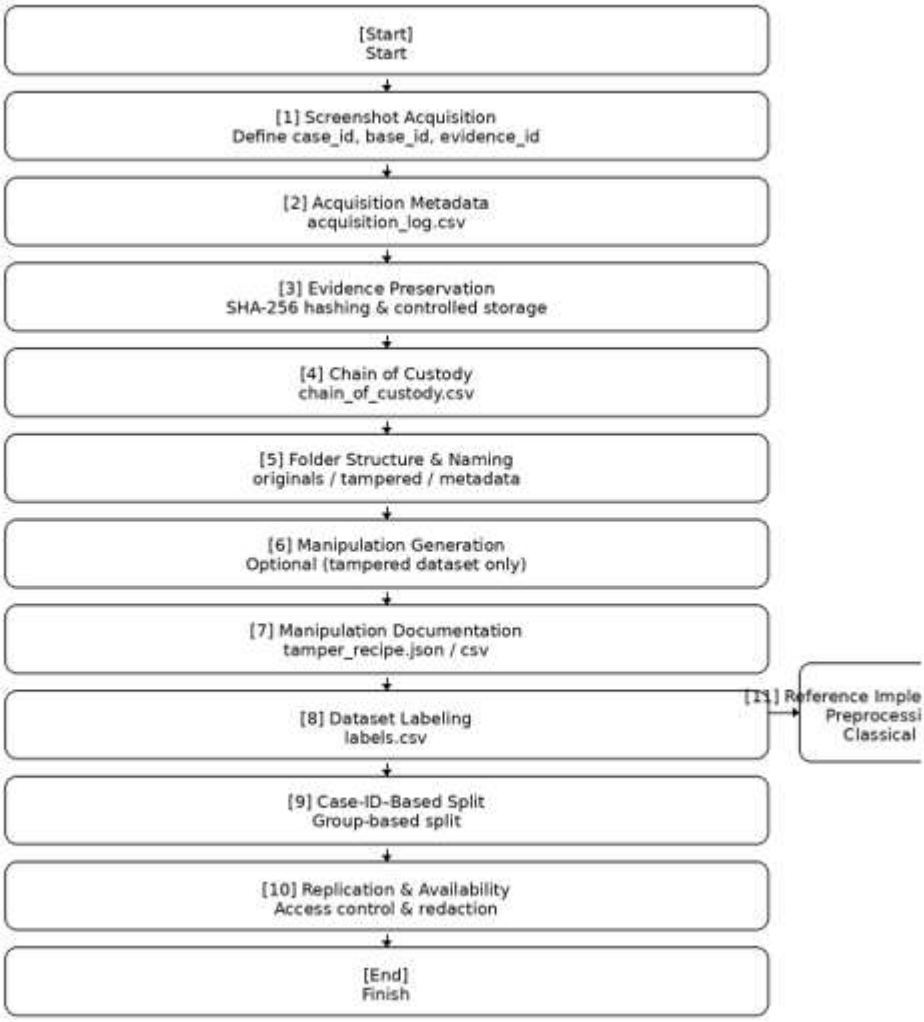

Figure 1. Flowchart of The Dataset Protocol and Evidence Preservation

This protocol is designed to construct a screenshot dataset that is relevant to cybersecurity incidents, such as phishing, website defacement, and security alerts displayed on monitoring dashboards. The dataset may be developed from two complementary sources: publicly available image manipulation datasets used as baseline references, and a dedicated dataset of cyber incident screenshots constructed by the researchers. Each screenshot file is treated as a unit of digital evidence and assigned a unique identifier (*evidence_id*). Evidence files may represent original (pristine) screenshots or manipulated derivatives generated through documented manipulation procedures. To reduce device- or platform-specific bias, the protocol encourages variation in capture devices, operating systems, screen resolutions, and display settings during evidence acquisition or construction.

Within this protocol, a hierarchical identification scheme is applied to support traceability and prevent data leakage. A *case_id* represents a single case unit, such as one incident or one evidence scenario, which may include one or more related screenshots. The *base_id* denotes the original pristine evidence that serves as the parent for all derived manipulations, while the *evidence_id* uniquely identifies each individual file, whether original or manipulated. This hierarchical definition ensures that dataset splitting based on *case_id*

can effectively prevent derived manipulations from the same source evidence from appearing across different data splits.

To facilitate replication and systematic dataset management, the protocol specifies a standardized folder structure that separates original evidence from manipulated derivatives and centralizes metadata artifacts. Original screenshots are organized by incident category (e.g., phishing, defacement, SIEM alerts), while manipulated evidence is grouped by manipulation type (e.g., overlay text, cropping, splicing, copy-move, and recompression). All metadata files are stored within a dedicated *metadata* directory, including label definitions, acquisition logs, chain-of-custody records, and manipulation documentation. A consistent file-naming convention is recommended to encode temporal information, case identifiers, source context, incident category, sequence number, and versioning, thereby enhancing traceability and reducing ambiguity in large-scale datasets.

Labeling is implemented through a centralized label file to support both binary classification (pristine versus tampered) and multi-class classification based on manipulation type. In addition to labels, acquisition metadata are recorded to capture contextual information surrounding screenshot capture, such as device type, platform, resolution, and acquisition time. These metadata serve as provenance records and are essential for auditability, particularly when evidence originates from heterogeneous devices, platforms, or incident contexts.

Digital evidence preservation is enforced through mandatory hashing and chain-of-custody documentation. For every evidence file entering the dataset, acquisition metadata must be recorded, a SHA-256 hash must be computed, and original evidence must be stored in a controlled environment, preferably using read-only or append-only storage mechanisms. All transfers, copies, and access events are logged in a chain-of-custody record to ensure that the dataset constitutes not merely a collection of images, but an auditable package of digital evidence that aligns with forensic best practices.

Manipulated evidence is generated to reflect realistic modifications commonly applied to screenshots in cyber incident contexts, including overlays, cropping, splicing, copy-move operations, and recompression. To maintain label transparency and auditability, each manipulation is documented through a *tamper_recipe* that records the manipulation type, relevant parameters (e.g., crop dimensions or overlay position), software tools used, and timestamps. These recipes may be stored as individual JSON files linked to each *evidence_id* or as a consolidated CSV manifest, enabling independent verification and replication of the manipulation process.

Finally, the protocol recommends a data splitting strategy based on *case_id* to prevent leakage of derived manipulations across dataset partitions. All evidence files associated with the same case are assigned to a single subset (e.g., training, validation, or testing), ensuring that models are evaluated on genuinely unseen cases rather than unseen files derived from known evidence. This principle is particularly critical when a single pristine screenshot generates multiple manipulated variants. For cross-validation scenarios, group-based splitting at the *case_id* level is advised, reinforcing the notion that fair evaluation units in screenshot forensics reside at the case level rather than at the individual image file level.

4. Results and Discussion

Replication Artifacts and Dataset Reusability

The primary outcome of this study is the availability of a comprehensive set of replication artifacts that operationalize the proposed dataset and evidence preservation protocol for cyber incident screenshots. These artifacts are designed to enable independent researchers to reconstruct the dataset structure,

labeling logic, and evidence handling procedures in a consistent and auditable manner. By formalizing folder hierarchies, naming conventions, metadata schemas, and chain-of-custody records, the protocol transforms the dataset from a mere collection of labeled images into a reproducible and forensically grounded research asset.

A key result of this work is the definition of a standardized dataset architecture that clearly separates pristine screenshots from manipulated derivatives while centralizing all provenance-related metadata. The provision of structured templates such as *labels.csv*, *acquisition_log.csv*, *tamper_recipe*, and *chain_of_custody.csv* ensures that relationships among *evidence_id*, *base_id*, and *case_id* remain explicit and traceable throughout the research pipeline. This design directly addresses common reproducibility challenges in image forensics studies, where incomplete metadata and undocumented manipulation processes often limit cross-study comparison and validation.

In addition to protocol artifacts, the study provides an optional reference implementation in the form of a lightweight analytical pipeline suitable for triage scenarios. The inclusion of this pipeline serves a demonstrative purpose only, illustrating that the dataset can be readily consumed by conventional forensic analysis workflows. Importantly, this component is not presented as an experimental benchmark, but rather as an example of how the dataset supports practical usage without introducing methodological bias or overstating analytical performance.

Replication Specifications and Execution Environment Considerations

From a reproducibility standpoint, the results highlight the importance of explicitly documenting execution environments and analytical configurations when using the proposed protocol. Variations in operating systems, library versions, preprocessing steps, feature extraction parameters, and random seeds can materially influence analytical outcomes, even when the underlying dataset remains unchanged. As such, the protocol emphasizes documentation of these factors as part of the replication process rather than treating them as incidental implementation details.

This emphasis reflects an understanding that forensic datasets are particularly sensitive to preprocessing and parameterization choices. For example, resizing strategies, patch-based processing, or quantization levels in texture analysis may alter image statistics in ways that confound cross-study comparisons. By recommending transparent documentation of such configurations, the protocol mitigates the risk that observed differences in results are driven by implementation variance rather than substantive methodological differences.

Data Availability Strategies and Risk-Aware Sharing

A central discussion point arising from this study concerns the tension between scientific openness and data sensitivity. Cyber incident screenshots may contain personal identifiers, internal system information, or organizational secrets, making unrestricted public release impractical or unethical in many cases. To address this challenge, the protocol articulates multiple data availability models open, restricted, and controlled-access environments allowing dataset sharing strategies to be calibrated according to risk level and institutional policy.

This flexible availability framework ensures that the scientific objective of protocol replication can be achieved without increasing the risk of data leakage. Even when raw screenshot data cannot be openly distributed, the availability of protocol artifacts, metadata templates, and manipulation recipes allows other researchers to reconstruct equivalent datasets under their own governance constraints. In this sense, the protocol decouples methodological transparency from unconditional data exposure, which is particularly important in cybersecurity and digital forensics research.

Privacy Risks in Cyber Incident Screenshots

The discussion further underscores that screenshots used as cyber incident evidence often embed sensitive information, including user identities, session tokens, internal IP addresses, configuration details, and contextual organizational data. These risks are amplified when evidence is shared across research teams or stored without strict access controls. Consequently, ethical and privacy considerations are treated as integral components of the dataset protocol rather than as post hoc safeguards.

1. Redaction and Data Minimization Policies
   To mitigate privacy risks, the protocol advocates a data minimization and redaction strategy that balances scientific utility with ethical responsibility. Techniques such as masking personal identifiers, selectively cropping irrelevant regions, and replacing sensitive tokens with placeholders are recommended as standard practice. Crucially, all redaction actions are recorded in metadata to preserve provenance and analytical transparency. When unredacted evidence is required for specific forensic objectives, the protocol explicitly categorizes such data as restricted, requiring enhanced access controls and audit logging.
2. Access Control, Retention, and Governance
   Access control mechanisms form another important discussion outcome. The protocol aligns dataset governance with forensic chain-of-custody principles by recommending access classification, approval workflows, usage agreements, activity logging, encryption during storage and transfer, and clearly defined retention policies. These measures ensure that the dataset remains accountable throughout its lifecycle and that responsibility for evidence handling is clearly assigned, even in collaborative research settings.

Limitations and Future Work

1. Protocol and Dataset Limitations
   The scope of this study is deliberately focused on protocol design and evidence preservation rather than on achieving or comparing state-of-the-art analytical performance. As a result, the analytical pipeline included in this work functions solely as a reference implementation. Users seeking performance-oriented insights must conduct separate empirical evaluations tailored to their specific research questions.
2. Another limitation concerns dataset representativeness. The diversity of screenshot sources across devices, operating systems, display configurations, and post-processing pipelines directly affects the statistical characteristics of the dataset. Without careful documentation and diversification, such variability may limit the generalizability of downstream analyses. The protocol therefore places responsibility on dataset builders to explicitly report acquisition conditions and transformations.
3. Threats to Validity
   Several threats to validity are identified in the discussion. Internal validity may be compromised by data leakage if manipulated derivatives from the same base screenshot are split across training and evaluation sets, by labeling errors arising from undocumented manipulations, or by tool-specific artifacts when manipulations are generated using a single editing application. The protocol mitigates these risks through group-based splitting at the *case_id* or *base_id* level, auditable manipulation recipes, and encouragement of multi-tool manipulation generation.
4. External validity is challenged by platform heterogeneity and domain specificity. Models trained on screenshots from a single operating system, application type, or incident category may not generalize to different environments. Addressing this limitation requires cross-device and cross-application data collection, as well as transparent reporting of per-source and per-compression performance when empirical studies are conducted.

5. Directions for Future Research
   Future work may extend the protocol by incorporating additional baseline analytical methods, such as alternative texture descriptors or lightweight convolutional models, to broaden comparative reference points. Further studies may explore ablation analyses to assess sensitivity to feature parameters and preprocessing choices, as well as cross-domain evaluations to test robustness under varied operational conditions. Statistical significance testing, expanded replication artifacts, and deeper integration with SOC and DFIR operational workflows represent additional avenues for strengthening both the scientific and practical impact of the proposed protocol.

5. Conclusion

This study proposes a comprehensive and replicable protocol for constructing datasets and preserving digital evidence in the context of cyber incident screenshots. Recognizing that screenshots are frequently used as preliminary or primary evidence in cybersecurity practice, yet are inherently vulnerable to manipulation, the protocol integrates dataset design with established principles of digital forensic evidence preservation. Key contributions include the formal definition of evidence units and case hierarchies, standardized folder structures and naming conventions, auditable labeling and acquisition metadata, documented manipulation procedures through *tamper_recipe*, and strict chain-of-custody records supported by SHA-256 hashing.

By adopting a *case_id*-based data splitting strategy, the protocol mitigates data leakage risks that commonly undermine the validity of empirical evaluations in image forensics. The inclusion of optional reference implementations demonstrates dataset usability for triage-oriented analysis without overstating analytical performance, thereby maintaining a clear separation between methodological contribution and experimental claims.

Beyond technical aspects, the protocol explicitly addresses ethical, privacy, and access-control considerations, offering flexible data availability models that balance scientific reproducibility with the protection of sensitive information. While the protocol does not aim to establish state-of-the-art detection accuracy, it provides a robust foundation for reproducible and auditable research on screenshot manipulation in cybersecurity contexts. Future studies can build upon this framework to conduct fair comparative evaluations, extend analytical methods, and strengthen the operational integration of screenshot forensics within DFIR workflows.

6. References

[1] B. Guttman, D. R. White, S. Williams, and T. Walraven, “Digital evidence preservation: Considerations for evidence handlers,” 2022.
[2] S. Chinthala, “Analyzing the Effectiveness of SIEM Tools in Threat Mitigation: A Qualitative Study in Cybersecurity.” University of the Cumberlands, 2024.
[3] A. Subcommittee, “OSAC 2024 - N - 0011 Standard Guide for Forensic Digital Image Management,” pp. 1–6, 2024, [Online]. Available: https://www.nist.gov/system/files/documents/2024/03/29/OSAC 2024-N-0011 Standard Guide for Forensic Digital Image Management_Version 1.0.pdf
[4] G. Johansen, *Digital Forensics and Incident Response: Incident response tools and techniques for effective cyber threat response*. Packt Publishing Ltd, 2022.
[5] J. D. Swerzenski, “Fact, fiction or Photoshop: Building awareness of visual manipulation through image editing software,” *J. Vis. Lit.*, vol. 40, no. 2, pp. 104–124, 2021.
[6] H. Fuchs, S. M. Pizer, E. R. Heinz, S. H. Bloomberg, L.-C. Tsai, and D. C. Strickland, “Design of and

image editing with a space-filling three-dimensional display based on a standard raster graphics system," in *Processing and Display of Three-Dimensional Data*, SPIE, 1983, pp. 117–129.

[7] R. Policy, "Scientific Working Group on Digital Evidence Scientific Working Group on Digital Evidence," *SWGE*, 2024, [Online]. Available: https://www.swgde.org/wp-content/uploads/2024/11/2024-11-20-Guidelines-for-Forensic-Image-Analysis-16-I-002-2.0.pdf

[8] P. Duszejko, T. Walczyna, and Z. Piotrowski, "Detection of Manipulations in Digital Images: A Review of Passive and Active Methods Utilizing Deep Learning," *Appl. Sci.*, vol. 15, no. 2, p. 881, 2025.

[9] D. Sharma, "A survey of image forensics: Exploring forgery detection in image colorization," 2025.

[10] M. Zanardelli, F. Guerrini, R. Leonardi, and N. Adami, "Image forgery detection: a survey of recent deep-learning approaches," *Multimed. Tools Appl.*, vol. 82, no. 12, pp. 17521–17566, 2023.

[11] X. Zhao, P. Bateman, and A. T. S. Ho, "Image authentication using active watermarking and passive forensics techniques," in *Multimedia Analysis, Processing and Communications*, Springer, 2011, pp. 139–183.

[12] R. G. Mani, R. Parthasarathy, S. Eswaran, and P. Honnavalli, "A survey on digital image forensics: Metadata and image forgeries," in *Workshop on Applied Computing, January*, 2022, pp. 27–28.

[13] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Trans. Syst. Man. Cybern.*, no. 6, pp. 610–621, 2007.

[14] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002.

[15] A. Xiang, J. Zhang, Q. Yang, L. Wang, and Y. Cheng, "Research on splicing image detection algorithms based on natural image statistical characteristics," *arXiv Prepr. arXiv2404.16296*, 2024.

[16] R. J. Al-Azawi, N. M. G. Al-Saidi, H. A. Jalab, R. W. Ibrahim, and D. Baleanu, "Image Splicing Detection Based on Texture Features with Fractal Entropy," *Comput. Mater. Contin.*, vol. 69, no. 3, 2021.