AnonLFI 2.0: Extensible Architecture for PII Pseudonymization in CSIRTs with OCR and Technical Recognizers

Cristhian Kapelinski, Douglas Lautert, Beatriz Machado, Diego Kreutz[1]

[1]AI Horizon Labs – PPGES – Federal University of Pampa (UNIPAMPA)

{cristhianavilla,douglaslautert,beatrizmachado}.aluno@unipampa.edu.br
diegokreutz@unipampa.edu.br

Abstract. *This work presents AnonLFI 2.0, a modular pseudonymization framework for CSIRTs that employs HMAC–SHA256 to generate strong and reversible pseudonyms, preserves the structural integrity of XML and JSON documents, and integrates OCR as well as specialized technical recognizers for PII and security–related artifacts. In two case studies involving OCR applied to PDF files and the pseudonymization of OpenVAS XML reports, the system achieved 100% precision and F1 scores of 76.5% and 92.13%, demonstrating its effectiveness for the secure preparation of complex cybersecurity datasets.*

1. Introduction

Processing incident data by Computer Security Incident Response Teams (CSIRTs) is essential for automated threat analysis, detection engineering, and collaborative defense [Sarhan et al. 2022, Preuveneers and Joosen 2021]. However, such data frequently contain Personally Identifiable Information (PII), creating a structural tension between analytical utility and privacy regulations such as GDPR [Amoo et al. 2024]. Classical anonymization approaches, such as k-anonymity [Sweeney 2002], are often insufficient for this domain; research indicates that increasing the strength of k-anonymity (k-value) degrades the performance of machine learning classifiers used in threat detection [Slijepčević et al. 2021], removing the semantic granularity necessary for correlating Indicators of Compromise (IoCs).

Pseudonymization represents the most suitable strategy for this domain, but existing solutions present critical limitations in three dimensions: (1) cryptographic security, often reduced due to the use of public *hashes* vulnerable to *rainbow table* attacks; (2) structural preservation, with tools that flatten hierarchical formats like XML/JSON or ignore PII embedded in images; and (3) semantic coverage, as many do not recognize fundamental technical entities, such as malware *hashes*, certificate serial numbers, and vulnerability identifiers. The previous version of AnonLFI [Bandel et al. 2025] validated the hybrid approach based on NER/RegEx, but had a monolithic architecture and suffered from these same gaps.

This paper presents AnonLFI 2.0 [Kapelinski et al. 2025], a complete reengineering into a modular *framework* designed for cybersecurity requirements at scale. Main contributions include: (1) cryptographically robust pseudonymization via HMAC-SHA256 with secret key, enabling federated correlation and mitigating inversion attacks; (2) a processing *pipeline* composed of native processors capable of preserving the structure of XML and JSON files, plus an OCR module for extracting PII in images and PDF documents[1]; (3) specialized technical recognizers, such as HASH and CERT_SERIAL; and (4) a CLI for controlled re-identification with audit trail support. The empirical evaluation comprises two representative case studies: a PDF processed via OCR (Precision = 100%, F1 = 76.5%) and an OpenVAS XML report (Precision = 100%, F1 = 92.13%). Results demonstrate that AnonLFI 2.0 offers secure pseudonymization, preserves analytical utility, and is suitable for building complex *datasets* in cybersecurity.

2. State of the Art

Current solutions for PII de-identification range from general-purpose cloud services and extensible frameworks to domain-specific academic proposals. Table 1 summarizes the landscape, contrasting the scope and techniques of prominent solutions with AnonLFI.

Table 1. Comparison of de-identification approaches and the position of AnonLFI.

Solution	Data Scope	Main Technique
Google DLP	Structured and Text	Hashing (HMAC) or Redaction.
Amazon Comprehend	Text (NLP)	Detection and Masking..
Microsoft Presidio	Text, Images (OCR)	Extensible framework (NER/RegEx).
Vakili et al. (2024) [Vakili et al. 2024]	Clinical Text (Swedish)	Replacement with realistic *surrogates*.
Masketeer (2024) [Baumgartner et al. 2024]	Medical Text (German)	Ensemble-based pseudonymization.
Yermilov et al. (2023) [Yermilov et al. 2023]	Text (NLP)	Comparative analysis (NER vs. LLMs).
AnonLFI 1.0 [Bandel et al. 2025]	TXT, CSV, DOCX, XML/XLSX (conversion to CSV).	Public SHA256 hash (vulnerable).
AnonLFI 2.0	**PDF, Images (OCR), XML/JSON (native), TXT, CSV, DOCX.**	**HMAC-SHA256 (secure) and Reversible.**

Commercial services like **Google DLP** and **Amazon Comprehend** are limited by their closed-source nature and cloud reliance, introducing data sovereignty risks and compliance challenges for CSIRTs. Additionally, they often prioritize irreversible redaction, hindering the correlation of indicators of compromise (IoCs). While **Microsoft Presidio** offers an on-premise alternative, it functions as a generic framework lacking native logic for cybersecurity hierarchies, requiring extensive manual customization.

Recent academic proposals improve semantic utility but remain ill-suited for technical workflows. [Vakili et al. 2024] and [Baumgartner et al. 2024] focus on unstructured clinical prose, lacking mechanisms to preserve rigid schemas like OpenVAS XML or JSON logs. Furthermore, while [Yermilov et al. 2023] explores LLMs for naturalistic replacement, this creates a “privacy paradox” by requiring data transfer to third-party APIs.

[1] Partner CSIRTs reported incidents recorded in DOCX and PDF documents containing embedded screenshots.

This probabilistic method also fails to provide the cryptographic determinism needed to map IPs consistently across incidents.

AnonLFI 2.0 bridges these gaps by combining structural awareness with HMAC-SHA256 security. Unlike cloud-based tools or LLM approaches, it operates entirely on-premise, ensuring sensitive data remains within the secure perimeter while enabling controlled re-identification for audits.

2.1. AnonLFI 1.0

AnonLFI 1.0 [Bandel et al. 2025] established the first dedicated *pipeline* for pseudonymizing real security incidents processed by Brazilian CSIRTs. Its empirical evaluation, based on a hybrid approach combining spaCy NER and regular expressions, was performed on 763 real incidents and achieved Precision = 100% and Recall = 97.38%. Although these results validate the feasibility of the approach, the monolithic architecture of version 1.0 revealed seven structural limitations that restricted security, scalability, and applicability to more complex scenarios: (L1) the use of `hashlib.sha256` without a secret key made pseudonyms vulnerable to *rainbow table*-based attacks, compromising confidentiality; (L2) fixed truncation to 10 characters introduced high collision risk according to the birthday paradox, harming entity correlation; (L3) the absence of specialized recognizers for technical entities (e.g., malware *hashes*, X.509 certificate serial numbers, CPE strings) limited extensibility to new domains and formats; (L4) conversion of XML or XLSX to CSV destroyed the hierarchy and semantics of structured data, reducing analytical utility; (L5) there was no support for detecting and pseudonymizing PII present in *screenshots* or image-based documents; (L6) the reversal process depended on manual SQLite queries, making controlled and auditable re-identification impossible as required by regulatory practices; (L7) language support was restricted to Portuguese, making multilingual processing common in incident response environments unfeasible.

3. AnonLFI 2.0: Architecture and Evolution

AnonLFI 2.0[2] redesigns the previously monolithically *pipeline*, now adopting a modular *framework* composed of four decoupled components: (1) a CLI responsible for routing and handling input files; (2) a central engine based on Microsoft Presidio, in charge of orchestrating language-specific spaCy models and a multilingual Transformer model (`Davlan/xlm-roberta-base-ner-hrl`); (3) a processing module structured according to the *Factory* pattern, with dedicated processors for each format (PDF, JSON, XML, image, DOCX, XLSX); and (4) a configuration module that externalizes critical execution parameters and pseudonymization policies.

Recognition engine: The architecture maintains and extends the original hybrid engine, combining Microsoft Presidio with language-specific spaCy models, the multilingual Transformer model `Davlan/xlm-roberta-base-ner-hrl`, and specialized regular expressions. Performance metrics vary according to the characteristics of each dataset, considering entity prevalence, syntactic complexity, and application domain. The scenarios evaluated in this work introduce challenges not present in the original 763 incidents, including the need for PII extraction via OCR and recognition of specialized technical entities.

[2] `https://github.com/AnonShield/AnonLFI2.0`

Table 2 presents the mapping between limitations of the previous version and the technical and architectural solutions introduced in AnonLFI 2.0. The proposed solutions systematically address the problems identified in v1.0. In terms of security (L1), the use of HMAC-SHA256 with `SECRET_KEY` defined as an environment variable prevents *rainbow table*-based attacks and enables federated correlation between pseudonymized datasets. To ensure integrity and avoid collisions (L2), the `--slug-length` parameter (default 64) reduces collision probability to negligible values ($< 10^{-60}$), while storing the complete 256-bit hash in the database ensures precise re-identification.

Table 2. AnonLFI 2.0: solutions to version 1.0 limitations

ID	Limitation	Solution
L1	Vulnerable SHA256	HMAC-SHA256 with SECRET_KEY
L2	Collisions (10 chars)	Configurable `--slug-length` (default 64)
L3	No technical recognizers	HASH, CERT_SERIAL, CERT_BODY, CPE
L4	CSV flattening	Native JSON/XML/XLSX processors
L5	No images	OCR pipeline (Tesseract)
L6	Manual re-identification	CLI `deanonymize.py` with audit
L7	Fixed language	`--lang` parameter (24 languages)

On the extensibility axis (L3), the inclusion of new regular expression-based recognizers (`HASH`, `CERT_SERIAL`, `CERT_BODY`, `CPE_STRING`) enables processing of technical entities common in vulnerability reports. Regarding structured format handling (L4), the `JsonFileProcessor` and `XmlFileProcessor` components recursively traverse native file structures, preserving the original hierarchy. Expanded data coverage (L5) is achieved with the `ImageFileProcessor`, which applies OCR (Tesseract) to both standalone images and images embedded in documents.

Re-identification (L6) is now conducted by a dedicated CLI (`deanonymize.py`), which requires the presence of `SECRET_KEY` and records audit events. Finally, multilingual support (L7) is made flexible through the `--lang` parameter, which dynamically loads spaCy models according to the desired language, complemented by the `--allow-list` parameter, which defines terms that should be preserved.

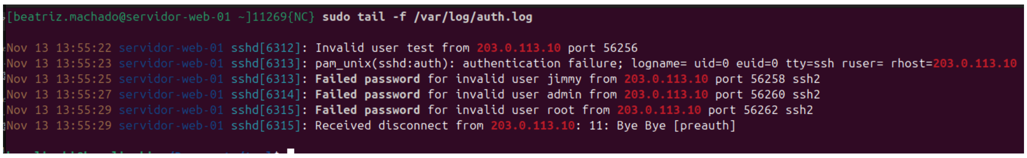
4. Case Studies

Two case studies were used to validate the main architectural capabilities introduced in AnonLFI 2.0. The evaluation employed standard NER metrics: Precision ($TP/(TP+FP)$), Recall ($TP/(TP+FN)$), and F1-Score. The *ground truth* was established through manual annotation independently performed by two security researchers, ensuring consistency and reliability of labels.

4.1. Scenario 1: PDF with Images (OCR)

This scenario validates the OCR *pipeline* (solution L5) using a security incident report (`incidente_ssh.pdf`) concerning an SSH attack. The document includes PII both in digital text and in a terminal screenshot with *syntax highlighting* (Figure 1).

Results: The ground truth contains 21 sensitive entities. We obtained TP = 13, FP = 0, and FN = 8, resulting in **Precision = 100%**, **Recall = 61.9%**, and **F1 = 76.5%**.



```
[beatriz.machado@servidor-web-01 ~]11269{NC} sudo tail -f /var/log/auth.log

Nov 13 13:55:22 servidor-web-01 sshd[6312]: Invalid user test from 203.0.113.10 port 56256
Nov 13 13:55:23 servidor-web-01 sshd[6313]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=203.0.113.10
Nov 13 13:55:25 servidor-web-01 sshd[6313]: Failed password for invalid user jimmy from 203.0.113.10 port 56258 ssh2
Nov 13 13:55:27 servidor-web-01 sshd[6314]: Failed password for invalid user admin from 203.0.113.10 port 56260 ssh2
Nov 13 13:55:29 servidor-web-01 sshd[6315]: Failed password for invalid user root from 203.0.113.10 port 56262 ssh2
Nov 13 13:55:29 servidor-web-01 sshd[6315]: Received disconnect from 203.0.113.10: 11: Bye Bye [preauth]
```

Figure 1. Terminal screenshot with PIIs in high and low contrast areas

Analysis: False Negatives have two main origins: (1) *OCR failures (5 cases)*: the *hostname* `servidor-web-01` was read as `seryidor web 01` due to low contrast between dark blue and the purple background of the Ubuntu terminal. This limitation is consistent with broader challenges in forensic text extraction, where image variability, resolution, and background noise significantly hinder recognition accuracy [Blanco-Medina et al. 2020]. (2) *Recognition engine failures (3 cases)*: absence of a specific recognizer for the `[user@host]` pattern in the prompt `[beatriz.machado@servidor-web-01]`; partial recognition of compound name (inappropriately preserving the surname `Machado`); and non-classification of the *hostname* when inserted in log context. Among the 13 correctly identified entities, 6 were IP addresses extracted via OCR, demonstrating the method's effectiveness when visual contrast is adequate.

4.2. Scenario 2: Vulnerability XML (OpenVAS)

This scenario validates technical recognizers (solution L3) and structural preservation (solution L4). An OpenVAS scan executed against `localhost` generates a native XML file with nested hierarchical structure, containing specialized technical entities (e.g., *hashes*, X.509 certificates, CPE strings) and a significant volume of structured data.

```
$ uv run anon.py vulnerabilidadexml.xml --lang en \
  --slug-length 10 --preserve-entities "CPE_STRING" \
  --allow-list "App,OS,Done,UTC,Default Accounts, \
   Greenbone,Greenbone AG,Greenbone Community Feed, \
   GCF,MQTT Broker Detection, (TCP),Redis"
```

Iterative process: Initial validation revealed false positives involving product names, metadata, generic technical terms, a numeric timestamp, and the expression `MQTT Broker Detection (TCP)`, identified by auditing the SQLite database. Applied corrections consisted of refining the `--allow-list` parameter and adjusting the hexadecimal *hostname* RegEx to exclude *timestamp* patterns.

Results: The reference set contained 48 sensitive entities. TP = 41, FP = 0, and FN = 7 were obtained, resulting in *Precision = 100%*, *Recall = 85.42%*, and *F1 = 92.13%*.

Analysis: The 100% precision confirms the effectiveness of new technical recognizers. The 7 False Negatives derived from a credential in a non-conventional format and geographic locations in X.509 certificates (`DE`, `Osnabrueck`). While the credential failure stems from an unrecognized pattern, the missed locations illustrate the *context gap*: general-purpose NER models, typically trained on natural language prose (e.g., news corpora), rely heavily on grammatical cues and general context, struggling with the specialized technical vocabulary and lack of annotated data in the security domain [Ma et al. 2020, Zhang et al. 2025]. The 41 TP include IPs, *hostnames*, *hashes*, and complete X.509 certificates, indicating good overall performance.

5. Discussion

Design decisions. The v2.0 architecture was designed to enable the secure use of complex data in automated analyses and LLM training. The use of HMAC-SHA256 with `SECRET_KEY` enables federated correlation between trusted CSIRTs without PII exposure. Unlike simple hashing, HMAC provides a robust mechanism for privacy-preserving record linkage (PPRL), aligning with keyed approaches used in distributed systems [Randall et al. 2014], eliminating vulnerabilities associated with public hashes. The `--slug-length` parameter (default 64) reduces collision probability to negligible values, ensuring cryptographic integrity even in massive *datasets*, while the complete hash stored in the database ensures precise re-identification. The observed variation in metrics (F1 = 76.5% in OCR scenario and F1 = 92.13% in OpenVAS technical scenario) reflects differences inherent to *dataset* characteristics, expected behavior in NER systems.

Limitations and future work. Current main limitations include: (1) support for a single language per document, harming multilingual cases; and (2) dependence on manual SQLite database inspection for fine-tuning configuration, unsuitable for high-scale environments.

Futuros trabalhos: (1) segurança federada com PKI; (2) detecção de idioma para NER dinâmico; (3) assistência automatizada via Ollama para análise, listas e relatórios; (4) *fine-tuning* de modelos NER para reconhecimento contextual em cibersegurança.

6. Final Remarks

This paper presented AnonLFI 2.0, a redesigned and extensible framework that adds HMAC–SHA256 pseudonymization, configurable slug sizes, specialized technical recognizers, native processors for hierarchical formats (JSON, XML, XLSX), an OCR pipeline for image–based PII extraction, a controlled re–identification CLI, and multilingual support for 24 languages. Case studies with a PDF containing images (Precision = 100%, F1 = 76.5%) and an OpenVAS XML report (Precision = 100%, F1 = 92.13%) confirm the effectiveness of these new capabilities, including functionalities not available in the original engine. With these advances, AnonLFI 2.0 enables CSIRTs to prepare complex datasets of incidents and vulnerabilities[3] for advanced analysis and secure sharing in compliance with structural, cryptographic, and regulatory requirements.

Acknowledgments. This research was partially supported by CT-Cibersegurança/RNP[4]; by CNPq[5], grant 409743/2025-9; and by FAPERGS[6], through grant agreements 24/2551-0001368-7 and 24/2551-0000726-1.

References

Amoo, O. O., Atadoga, A., Osasona, F., Abrahams, T. O., Ayinla, B. S., and Farayola, O. A. (2024). GDPR's impact on cybersecurity: A review focusing on USA and European practices. *International Journal of Science and Research Archive*, 11:1338–1347.

[3] We are also working on extracting vulnerabilities from scanner reports for use in AnonLFI 2.0 [Machado et al. 2025].

[4] `https://plataforma.rnp.br/ct-ciberseguranca`

[5] `https://www.gov.br/cnpq/pt-br`

[6] `https://fapergs.rs.gov.br`

Bandel, C. T., Esteves, J. P. R., Guerra, K. P., Bertholdo, L. M., Kreutz, D., and Miani, R. S. (2025). Anonimização de incidentes de segurança com reidentificação controlada. In *Anais do XXV SBSeg*. SBC.

Baumgartner, M., Kreiner, K., Wiesmüller, F., Hayn, D., Puelacher, C., and Schreier, G. (2024). Masketeer: An ensemble-based pseudonymization tool with entity recognition for german unstructured medical free text. *Future Internet*, 16(8):281.

Blanco-Medina, P., Fidalgo, E., Alegre, E., Alaiz-Rodríguez, R., Jáñez-Martino, F., and Bonnici, A. (2020). Rectification and super-resolution enhancements for forensic text recognition. *Sensors*, 20(20):5850.

Kapelinski, C., Lautert, D., Machado, B., and Kreutz, D. (2025). AnonLFI 2.0: Extensible architecture for PII pseudonymization in CSIRTs with OCR and technical recognizers. *arXiv preprint arXiv:2511.15744*.

Ma, P., Jiang, B., Lu, Z., Li, N., and Jiang, Z. (2020). Cybersecurity named entity recognition using bidirectional long short-term memory with conditional random fields. *Tsinghua Science and Technology*, 26:259–265.

Machado, B., Lautert, D., Kapelinski, C., and Kreutz, D. (2025). Structured extraction of vulnerabilities in openvas and tenable was reports using llms. `https://arxiv.org/abs/2511.15745`.

Preuveneers, D. and Joosen, W. (2021). Sharing machine learning models as indicators of compromise for cyber threat intelligence. *J. of Cybersecurity and Privacy*, 1(1).

Randall, S. M., Ferrante, A. M., Boyd, J. H., Bauer, J. K., and Semmens, J. B. (2014). Privacy-preserving record linkage on large real world datasets. *J. of Biomed Informatics*.

Sarhan, M., Layeghy, S., Moustafa, N., and Portmann, M. (2022). Cyber threat intelligence sharing scheme based on federated learning for network intrusion detection. *Journal of Network and Systems Management*, 31.

Slijepčević, D., Henzl, M., Klausner, L., Dam, T., Kieseberg, P., and Zeppelzauer, M. (2021). k-Anonymity in practice: How generalisation and suppression affect machine learning classifiers. *Computers & Security*, 111:102469.

Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570.

Vakili, T., Henriksson, A., and Dalianis, H. (2024). End-to-end pseudonymization of fine-tuned clinical BERT models: Privacy preservation with maintained data utility. *BMC Medical Informatics and Decision Making*, 24(162).

Yermilov, O., Raheja, V., and Chernodub, A. (2023). Privacy- and utility-preserving nlp with anonymized data: A case study of pseudonymization. In *TrustNLP*, pages 232–241.

Zhang, Y., Liu, J., Zhong, X., and Wu, L. (2025). SecLMNER: A framework for enhanced named entity recognition in multi-source cybersecurity data using large language models. *Expert Systems with Applications*, 271:126651.