Research on Key Technologies of an Intelligent Judgment and Homologous Expansion System for Fraud-Related Risk URLs

Hang Du, Menglei Li*

Liaoning Police College, Dalian, China

*Email: job.123@163.com

Abstract. In response to the technical countermeasures in current telecommunications network fraud, such as template-based generation of fraudulent APKs and URLs, code obfuscation and reinforcement, rapid domain name rotation, and dynamic hiding of behaviors, traditional single-dimension detection and manual analysis methods can no longer meet practical needs. This paper presents the design and implementation of an intelligent assessment and homologous expansion system for fraud-related risk URLs. The system integrates three core technologies: dynamic and static APK detection and analysis, multi-dimensional URL feature extraction, and global homologous. By employing automated unpacking, sandbox behavior simulation, and anti-detection hooks, it overcomes APK reinforcement and feature hiding to accurately extract associated URLs. A 14-dimensional URL feature system is constructed and a unique feature gene is generated for precise characterization of fraud-related URLs. Based on feature genes and the DBSCAN clustering algorithm, the system achieves dynamic mining and expansion of globally homologous fraud-related URLs.

Keywords: Fraud-related URL; fraud-related APK; intelligent judgment; homologous expansion; feature gene; dynamic-static analysis; whole-network mining

1 Introduction

With the rise of telecom fraud, website/APP scams now account for over 76%, mostly family-style samples using packing, obfuscation, domain rotation, and dynamic hiding. “A recurring tactic is the use of algorithmically generated domains (AGDs) created by domain generation algorithms (DGAs) to orchestratebotnet command-and-control, host phishing content, and distribute malware.”[1] Traditional manual and single-dimension methods are inadequate. “In the field of intelligent system security, the implementation of active interception and blocking measures imposes extremely high requirements on the accuracy of malicious URL determination. Any misjudgment of the target to be intercepted may cause severe social impacts and even harm the legitimate rights and interests of benign websites. Therefore, constructing an efficient and accurate malicious

Z. Ge et al. (eds.), *Proceedings of the 2026 6th International Conference on Education, Information Management and Service Science (EIMSS 2026)*, Atlantis Highlights in Computer Sciences 25,
https://doi.org/10.2991/978-94-6239-766-8_2

URL determination system has become a critical issue that urgently needs to be addressed in the cybersecurity domain." [2] This paper proposes an intelligent system integrating dynamic-static APK analysis, multi-dimensional URL feature extraction, and feature gene-based global mining. Key contributions include automated unpacking, a comprehensive 14-dimensional feature system, and efficient density-based homologous mining. Benchmark tests, quantitative evaluation, and real case studies fully verify the system's effectiveness.

2 Current Research Status of Related Technologies Domestically and Internationally

2.1 International Research Status

Foreign research on malware and malicious URL detection focuses mainly on general Android threats. Static tools (e.g., AndroGuard) extract permissions, API calls, and control flow graphs. Dynamic sandboxes (e.g., CuckooDroid) with Frida capture runtime behaviors for ML classification. Threat intelligence platforms like VirusTotal, URLhaus, and Talos are industry benchmarks. However, these studies lack deep optimization for telecom fraud scenarios—such as mining fraud keywords, visual features of impersonated pages, and family-style homologous relationships—thus failing to meet China's anti-fraud need of "detect one, dig out a batch." "Existing detection systems have implemented multi-modal methods such as HTML parsing, natural language processing (NLP), image-based logo identification, and detection techniques to identify impersonation attacks. Although existing work has showy with computational overhead, delays, and extensibility in low-capacity systems such as browser-based extensions or mobile applications."[3] "As a result, a vast number of new phishing webpages are continuously generated, necessitating advanced detection mechanisms. Recent advancements in machine learning have shown promise in addressing these persistent threats."[4]

Domestic Research Status

Domestic research on fraudulent APKs/URLs focuses on three areas: static decompilation, sandbox-based traffic capture, and simple hash matching. However, these methods suffer from single-dimension features, poor unpacking of commercial packers, low global mining precision, and high false positives. Most systems lack benchmark comparisons with platforms like VirusTotal and quantitative evaluation on labeled datasets, leaving detection rate, precision, and FPR unclear—hindering engineering deployment and practical use.

3 Overall System Architecture and Core Technical Methods

3.1 Overall System Architecture

The system adopts a modular design, with the overall architecture comprising three core functional modules and one data support layer, realizing an integrated process from raw sample input to global risk network discovery. The three core modules are: APK detection and analysis module, URL multi-dimensional feature extraction module, and global dynamic mining module. The data support layer is responsible for multi-source heterogeneous data collection, preprocessing, standardized storage, and quality monitoring. The overall technical workflow is: sample collection → dynamic-static APK analysis (unpacking/decompilation/behavior capture) → fraud-related URL extraction → URL multi-dimensional feature extraction and feature gene generation → global retrieval and feature matching → homologous clustering and expansion → result review and warning disposal.

3.2 APK Detection and Analysis Module

This module aims to break through the packing and obfuscation countermeasures of fraudulent APKs and accurately extract associated fraud-related URLs and other key information from within the APK.

Static Analysis and Packing Identification. The module first uses Apktool to unpack the APK, parses the AndroidManifest.xml file to obtain basic information such as package name, version, signature, and sensitive permissions. It uses JADX to decompile classes.dex into Java code and builds a regular expression engine optimized for fraud scenarios to batch extract domain names/URLs, phone numbers, QQ numbers, and fraud-related high-frequency keywords such as "recharge", "withdraw", and "mentor". Meanwhile, a packing identification rule base covering over 20 common packer types (e.g., 360, Tencent, Baidu, VMP) is constructed, automatically determining the packing type and unpacking difficulty by matching characteristics like specific class names and abnormal code structures.

Automated Unpacking. For packed APKs, an automated unpacking technique based on Frida dynamic instrumentation is employed. The core process includes: injecting a custom Frida script into the target APK process, hooking the loadClass method of the ClassLoader to intercept dex file loading; when the dex is decrypted and loaded in memory, the script reads the dex data from memory and exports it; finally, a dex repair tool is used to fix the header information, checksums, etc., of the dumped file, generating a complete and usable dex file for subsequent decompilation.

Dynamic and Static Fusion Analysis. To capture hidden behaviors not obtainable through static analysis, the module builds an Android x86 sandbox environment based on Docker and configures a network proxy for full traffic capture. It uses Frida to hook four categories of core APIs: network APIs (OkHttp, HttpURLConnection), component lifecycle APIs, file read/write APIs, and anti-detection APIs (e.g., forcing isEmulator to return false). At the same time, automated behavior simulation scripts are designed to automatically execute key business processes like registration, login, recharging, and withdrawal, triggering and capturing all dynamic network requests. Finally, domains extracted statically and those captured dynamically are merged; the union expands coverage, while the intersection serves as high-confidence primary domains.

3.3 URL Multi-Dimensional Feature Extraction Module

This module is responsible for comprehensively and quantitatively describing URLs obtained from APK parsing or other sources, providing a data foundation for subsequent homologous mining.

Design of 14-Dimensional Feature System. To fully characterize the attributes of fraud-related URLs, this paper constructs a feature system with 14 dimensions. Specifically: 1. Basic package association features; 2. Basic URL features; 3. Hash features; 4. Registration features; 5. Network features; 6. DNS features; 7. WEB features; 8. Permission association features; 9. Network request features; 10. Fraud-related keyword features; 11. Similarity features; 12. Associated domain features; 13. API path features; 14. Homologous association features. Features in each dimension are obtained through WHOIS queries, DNS resolution, web crawling and rendering, TF-IDF keyword extraction, and other technical means.

Feature Vectorization and Feature Gene Generation. Multiple types of collected features are preprocessed: discrete features are one-hot encoded; text features are converted to text fingerprints using TF-IDF combined with SimHash; numerical features are normalized by min-max scaling. Then, all preprocessed features are concatenated into a high-dimensional vector, and Principal Component Analysis (PCA) is used for dimensionality reduction, finally generating a 256-dimensional compact feature vector. To uniquely identify a fraud-related URL or its family template, the SHA256 hash algorithm is applied to the feature vector along with key information such as the primary domain and template identifier, generating an irreversible unique "feature gene", providing a stable and efficient matching basis for global homologous mining.

3.4 Global Dynamic Mining Module

This module is the core engine for homologous expansion of fraud-related URLs based on feature genes.

Global Retrieval and Feature Matching. Using high-confidence seed domain titles, filing numbers, API paths, file hashes, etc., as keywords, global retrieval is performed using search engines and a distributed crawler cluster (Scrapy + Playwright). The crawler cluster is configured with IP pools and User-Agent pools to avoid blocking. For retrieval results, multi-dimensional matching rules are applied: URLs satisfying any three of the following criteria are identified as homologous candidate samples: title similarity ≥90%, page structure hash similarity ≥85%, core API paths identical, WHOIS/DNS/IP correlation, feature gene similarity ≥90%, or fraud-related keyword overlap ratio ≥70%.

Homologous Clustering Algorithm. The DBSCAN algorithm, which does not require pre-specifying the number of clusters, is used for automatic clustering of candidate samples. Optimizations for fraud samples: cosine similarity is used as the distance metric for high-dimensional feature vectors; the neighborhood radius is experimentally set to 0.15, and the minimum number of core points to 5; a KD-tree is introduced to accelerate distance calculation. The algorithm automatically identifies core points, border points, and noise points, grouping density-connected samples into the same fraud family, achieving the mining goal of "detect one, dig out a batch".

Alert Triggering and Disposal Integration. The system adopts a dynamic threshold scoring alert triggering mechanism. For each mined homologous family, three scores are comprehensively calculated: **Family size score** (based on the number of samples in the family, log-normalized), **Risk concentration score** (based on the proportion of high-risk IPs in the family, average registration freshness, etc.), and **Temporal burst score** (based on time series mutation detection of domain registration and activity times). The final risk score = 0.4 × family size score + 0.4 × risk concentration score + 0.2 × temporal burst score. When the total score exceeds a dynamically adjusted threshold (initially set to 0.75), a batch alert is triggered. Alerts are encapsulated in STIX/TAXII standard format, containing family core characteristics, all member IOCs (domains, IPs), risk score, and confidence. The system automatically links with ISPs (for domain/IP blocking) and CERTs (for intelligence sharing) through predefined APIs.

4 Experimental Design and Result Analysis

4.1 Experimental Environment and Datasets

Hardware environment includes a server cluster (16 cores, 32GB RAM × 4), Android x86 sandbox nodes (×8), and a crawler cluster (8 cores, 16GB RAM × 6). Software environment is based on Python 3.9, Java 11, Frida 16.0, and Scrapy 2.6.

For benchmark testing and quantitative evaluation, the following datasets were constructed:

Labeled test set: 5,000 confirmed fraud-related URLs or APK-associated domains randomly selected from historical case data as positive samples, and 2,000 benign

URLs from the top 100,000 sites of Alexa (shopping, news, government, etc.) as negative samples.

Open-source platform comparison dataset: 1,000 URLs marked as "malicious" or "phishing" in the past 30 days from VirusTotal and 1,000 from URLhaus, totaling 2,000.

Real-world validation dataset: 3,000 fraud-related APKs and 87,000 associated URLs accumulated in an anti-fraud practical platform in 2025.

4.2 Evaluation Metrics

General metrics: accuracy, recall, F1 score, false positive rate, processing efficiency. Specific metrics for core modules: APK unpacking success rate, domain extraction accuracy, feature extraction completeness, feature gene uniqueness, homologous mining accuracy, number of homologous expansions per sample.

4.3 Benchmark Test Comparison with Open-Source Platforms

To demonstrate the incremental value of this system over aggregated public threat intelligence sources, it was compared against VirusTotal and URLhaus. The test method: treat the three platforms/systems as independent detection engines, compare their detection coverage and update latency on the 2,000 malicious URLs from open-source platforms. It should be noted that this system does not use these two platforms as data sources, ensuring test independence. The comparison results are shown in Table 1.

Table 1. Benchmark Comparison with Open-source Threat Intelligence Platforms

Platform/System	*Detection Coverage*	*Avg. Update Latency*	*IP Expansion Accuracy*
VirusTotal	94.2% (multi-engine aggregation)	2-6 hours	Not provided
URLhaus	86.5%(community-reported)	1-12 hours	~35.2%
This System	**91.8%**	**15 minutes**	**88.6%**

The results show that although this system's detection coverage is slightly lower than VirusTotal (which aggregates hundreds of engines), it is significantly higher than URLhaus. More importantly, the update latency of this system is much lower than both, enabling rapid response to fraud domains that change on an hourly basis. Furthermore, the unique IP expansion accuracy of this system reaches 88.6%, whereas open-source platforms generally do not provide or provide only very basic IP associations, proving the unique added value of this system in proactive mining and correlation analysis.

4.4 Quantitative Evaluation of Core System Metrics

Quantitative evaluation of the overall system and core modules was performed on the self-built labeled test set (5,000 positive + 2,000 negative).

Evaluation of APK Detection and Analysis Module. Testing was conducted using 100 fraud-related APK samples (86 of which were packed). Results show that the dynamic-static fusion analysis scheme achieved an automated unpacking success rate of 96.2%, significantly higher than traditional static analysis (62.3%). Primary domain extraction accuracy reached 97.1%, and the average parsing time per APK was 18.6 seconds, meeting batch processing requirements.

Evaluation of URL Multi-Dimensional Feature Extraction Module. Tests on 3,000 fraud-related URLs showed a feature extraction completeness of 98.3%, feature gene uniqueness of 100%, and an average feature extraction time of 2.1 seconds per URL.

Evaluation of Global Dynamic Mining Module. Starting from 100 high-confidence seed domains, the system achieved a homologous mining accuracy of 96.7%, with an average expansion of 23 homologous URLs per seed sample, and an average global retrieval time of 128 seconds.

Overall System Performance Evaluation. On the complete labeled test set, the overall system achieved a detection accuracy of 96.8%, recall of 95.4%, F1 score of 96.1%, and false positive rate of 3.2%. Detailed results are shown in Table 2.

Table 2. Overall System Performance on Labeled Test Set

Metric	*Accuracy*	*Recall*	*F1 Score*	*False Positive Rate*
Overall System	96.8%	95.4%	96.1%	3.2%
APK Analysis Module	97.1%	96.5%	96.8%	-
Feature Extraction Module	98.5%	-	-	-
Mining & Expansion Module	96.7%	94.9%	95.8%	-

4.5 Real Law Enforcement Case Analysis

To verify the practical utility of the system, three anonymized real investigation cases are presented.

Case 1: Uncovering the Backend Network of a Fake Investment App. A municipal public security bureau received a report of a fake investment APP fraud. After inputting the involved APK into the system, the APK analysis module successfully unpacked it and extracted the core domain api.invest-xxx.com. The feature extraction module generated a feature gene for this domain. Based on this gene, the global mining module discovered and expanded 47 homologous fraudulent websites using the same backend API with only different domain names, and associated them with three core C2 server IPs sharing the same payment domain. These actionable clues directly supported subsequent digital forensics and requests for assistance from the server hosting provider.

Case 2: Family-Style Discovery of Fake Loan Websites. A city anti-fraud center received multiple reports about "no-collateral loan" websites. These URLs were input into the system as seeds. By analyzing page structure hash similarity (>92%) and WHOIS registration email correlations, the mining module clustered a large fake family containing 128 sites. Further IP feature analysis revealed that these sites were mainly hosted on five IP addresses located in a Southeast Asian country, and these IPs had historically hosted other flagged fraud-related domains. The system then generated an intelligence package containing all domains and IPs, successfully preventing further victimization by this family. In this case, relying solely on passive scanning platforms like VirusTotal would not have discovered these new unreported sites.

Case 3: Alert Triggering and False Positive Analysis. During routine monitoring, the system triggered a medium-level alert due to a domain's registration characteristics (newly registered, privacy protection) and the historical behavior score of its associated IP. Manual review revealed that the domain belonged to a legitimate startup's new project website, and the cloud service provider's IP had been previously flagged due to misconduct by other tenants. This case exposes that correlation analysis based on shared infrastructure (cloud IPs) may lead to false positives. The system has optimized this by marking such IPs as "high-commensal risk" and introducing an "IP reputation dilution" factor (i.e., if the proportion of fraud-related domains on the same IP is low, the IP's risk weight is reduced). This case also emphasizes the necessity of manual review mechanisms in critical decision-making.

5 Conclusion

“Traditional detection methods, often based on isolated analysis of email content or embedded URLs, fail to comprehensively address these evolving attacks. [5]” This paper presents an intelligent system integrating APK dynamic-static analysis, multi-dimensional URL features, and global homologous mining. It overcomes single-dimension detection limits via automated unpacking, feature genes, and density clustering. Benchmark tests show advantages in update timeliness and proactive mining. Core metrics exceed 95% with acceptable false positives. Real cases confirm actionable intelligence.

Future work includes unpacking against custom protectors, enriching overseas threat intel, and using LLMs for adaptability.

Acknowledgement

Liaoning Province Science and Technology Plan Joint Program (Technology Research and Development Program Project), project name: Black Sample Detection System for Telecom Network Fraud Crime APP, 2024JH2/102600157.

Liaoning Police College's Science and Technology Driven Police Technology Research Project "Unveiling the List and Leading the Way", project name: Research on Dalian Telecom Network Fraud Malicious Software Sample Analysis System.

References

1. Wang Fangyuan, Lian Zhichao, Li Qianmu, et al. Research on Malicious URL Detection Method Based on Multi-dimensional Features and PageRank Optimization[J]. Information Network Security, 2025, 25(04): 564-577.
2. Ibrahim Altan, Moshiur Farazi, et al. Dual-Path Phishing Detection: Integrating Transformer-Based NLP with Structural URL Analysis[C]/Proceedings of the ACS/IEEE 22nd International Conference on Computer Systems and Applications (AICCSA 2025), 2025.
3. Lucas Torrealba, Pedro Casas-Hernandez, et al. Malicious Domain Names Detection with DeepDGA, a Hybrid Character and Word Embeddings Deep Learning Architecture[C]// 21st International Conference on Network and Service Management (CNSM 2025), 2025: 1-6.
4. Dan Komosny. Phishing detection on webpages in European non-English languages based on machine learning[J]. Scientific Reports volume 15, Article number: 37472 (2025).
5. Real-Time Phishing Detection for Brand Protection Using Temporal Convolutional Network-Driven URL Sequence Modeling[J]. Electronics, 2025, 14(18): 3746.