



An Explainable Zero-Trust Framework for Detecting Generative-AI-Enhanced Phishing and Credential Theft Using Real-World Threat Data

Mohammed Ameen Saleh Al-Bareda *

Department of Cyber Security, Faculty of Computer Sciences, Sana'a University, Sana'a, Yemen

*Corresponding author: mohmmedas2004@gmail.com

ABSTRACT

Phishing has evolved from simple fraudulent messages into a multi-channel credential-theft ecosystem that combines malicious URLs, QR-code phishing, adversary-in-the-middle proxying, business e-mail compromise, deepfake-assisted social engineering, and generative-AI-assisted content generation. This paper proposes an explainable zero-trust framework for detecting and reducing phishing-related credential exposure using real-world threat intelligence and the PhiUSIIL public benchmark dataset. The study integrates APWG observations from 2024 through Q3 2025 with a benchmark containing 235,795 URLs, 54 variables, 134,850 legitimate samples, and 100,945 phishing samples. The framework combines URL and page-content features, brand-mismatch evidence, QR-code detonation, sender-authentication signals, phishing-resistant MFA, conditional access, browser isolation, and human-readable explanations. A supervised benchmark experiment was conducted using a single stratified 80/20 train-test split and three baseline classifiers. Decision Tree achieved 100.00% F1-score, Random Forest achieved 99.9975% F1-score, and Gaussian Naive Bayes achieved 98.1923% F1-score on the held-out test set. These single-split benchmark results support a layered conclusion: URL detection is useful, but stronger risk reduction is achieved when detection, identity, isolation, reporting, and continuous governance operate as one zero-trust system.

ARTICLE INFO

Keywords:

Phishing Detection, Generative AI, Zero Trust, Credential Theft, Passkeys, FIDO2/WebAuthn, Quishing, Explainable AI, Threat Intelligence, Cybersecurity Framework.

Article History:

Received: 16-May-2026,

Revised: 2-July-2026,

Accepted: 29-August-2026,

Published: 28 October 2026.

1. INTRODUCTION

Credential phishing remains one of the most persistent entry points for cyber incidents because it exploits a combination of technical weaknesses, user attention limits, identity-provider trust assumptions, and inconsistent security controls. Recent threat reporting shows that phishing is not declining; instead, it is changing shape. APWG reports from 2024 and 2025 show quarterly attack volumes close to or above one million observed attacks, while Q2 2025 reached 1,130,393 observed phishing attacks before declining to 892,494 in Q3 2025 [1–7]. This high volume indicates that phishing is no longer only an e-mail filtering problem; it is an identity-security problem and a business-risk problem. The rapid adoption of generative artificial intelligence intensifies this challenge. Large language models can assist attackers in producing

fluent messages, adapting lures to local language and culture, summarizing stolen organizational context, and rapidly generating variants that evade static rule sets. The same AI capabilities can support defenders through triage, feature extraction, and analyst assistance; however, ungoverned AI adoption also creates additional exposure, a concern emphasized in recent breach-cost and AI-governance reporting [8–11]. Traditional anti-phishing defenses often focus on URL blacklists, content signatures, and user awareness training. These mechanisms remain useful, but they are insufficient against adversary-in-the-middle phishing kits, QR-code phishing, short-lived infrastructure, and credential replay. The literature also shows that humans may fail to identify visual deception even when interface indicators are available [12–14]. Consequently, modern protection must com-

bine detection with identity controls that remain resistant even when a user clicks a malicious link. This study proposes an explainable zero-trust framework that treats phishing as a full attack lifecycle. The framework does not stop at classifying a URL as benign or malicious. Instead, it connects threat intelligence, feature extraction, risk scoring, explanation, access policy, response, and governance. The proposed design was evaluated using real APWG threat statistics and a supervised benchmark experiment on the PhiUSIIL URL dataset [15, 16]. It was also mapped to recognized security guidance, including CISA phishing-resistant MFA, NIST digital identity guidance, WebAuthn standards, FIDO passkey guidance, OWASP LLM risks, and MITRE ATT&CK phishing techniques [10], [17–22].

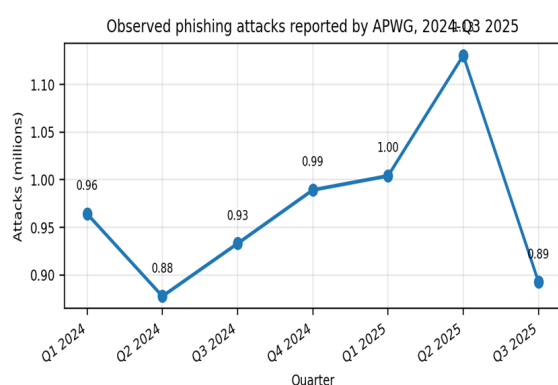


Figure 1. APWG quarterly phishing attack volumes from 2024 through Q3 2025 [1–7].

The motivation for this research is practical. Many institutions deploy anti-spam gateways, awareness messages, and password complexity policies; however, attackers continue to succeed by harvesting session tokens, abusing MFA fatigue, using legitimate cloud-hosted forms, and hiding malicious destinations behind QR codes. Therefore, a modern framework must answer three questions: how can suspicious phishing infrastructure be detected early, how can authentication remain secure even if users interact with malicious pages, and how can analysts understand why a specific message, URL, or QR code was flagged? This study makes five contributions. First, it presents an updated taxonomy of AI-enhanced phishing, including quishing, adversary-in-the-middle attacks, deepfake-assisted deception, and business-email compromise. Second, it integrates real APWG trend data and the PhiUSIIL public benchmark dataset into a reproducible research design. Third, it proposes a layered architecture that joins detection, zero-trust identity, automated response, and explainability. Fourth, it provides a detailed control mapping against NIST CSF 2.0 and MITRE ATT&CK. Fifth, it offers an implementation roadmap suitable for universities and organizations with constrained budgets. The research ques-

tion is formulated as follows: How can real-world phishing intelligence and public URL datasets be combined with explainable zero-trust controls to reduce credential exposure caused by generative-AI-enhanced phishing campaigns? The remainder of the paper is organized as follows. Section 2 reviews related work and identifies the research gap. Section 3 presents the materials and methods. Section 4 develops the threat model. Section 5 describes the proposed framework. Section 6 presents the results and discussion. Section 7 provides deployment guidelines, and Section 8 concludes the paper.

1.1. MOTIVATION AND SIGNIFICANCE

The significance of this research lies in the shift from perimeter-focused e-mail filtering toward identity-centered cyber defense. A single harvested password can expose cloud storage, e-mail, collaboration systems, financial workflows, and internal applications. Verizon breach reporting continues to highlight the importance of stolen credentials and social engineering in the breach landscape [8]. IBM breach-cost reporting further indicates that weak governance around AI and security controls can increase organizational costs and recovery complexity [9]. Universities and public institutions are especially exposed because they operate open communication environments, accept messages from external senders, use diverse devices, and often support many users with limited security staff. In such environments, the best defense cannot rely solely on blocking every malicious message. It must reduce the value of stolen credentials, limit lateral movement, produce explanations that help analysts and users learn, and collect metrics for continuous improvement [17–21]. Phishing-resistant authentication is a central concept. FIDO2 and WebAuthn passkeys bind authentication to an origin and use public-key cryptography instead of reusable secrets. This means that a fake domain cannot simply replay a password to the legitimate domain [20, 21], [23, 24]. However, passkeys alone do not remove the need for detection, because phishing messages may still deliver malware, payment fraud, fake invoices, or consent-grant attacks. Therefore, the proposed framework integrates passkeys with detection and response rather than treating them as a standalone solution.

1.2. OBJECTIVES

The main objective is to design a real-data-supported cybersecurity framework that detects and mitigates generative-AI-enhanced phishing and credential theft. The framework is intended to be understandable to researchers, implementable by practitioners, and aligned with security standards. The sub-objectives are as follows: (1) to analyze phishing trend data from APWG reports covering 2024 through Q3 2025; (2) to incorpo-

rate a public benchmark dataset with documented size, features, and labels; (3) to classify modern phishing attack types and map them to defensive controls; (4) to design an explainable risk-scoring pipeline; (5) to connect detection outcomes to zero-trust identity controls; and (6) to define measurable operational metrics such as time-to-block, click rate, credential-exposure reduction, and false-positive rate.

2. RELATED WORK AND RESEARCH GAP

Early phishing studies established that users can be deceived by visual similarity, misleading domains, and social cues. Dhamija et al. showed that users often fail to detect fraudulent sites, even when browser indicators are present [12]. Subsequent research on user susceptibility and intervention effectiveness showed that training can help but does not eliminate the problem [13]. Books and surveys on phishing countermeasures emphasized that phishing is socio-technical: attackers exploit both protocol limitations and human trust [14]. Phishing kit studies reveal that attackers reuse templates, automated panels, and credential collection workflows. Oest et al. analyzed phishing kits and demonstrated the operational structure of phishing ecosystems [25]. Credential-risk research also shows that stolen credentials can originate from multiple sources, including phishing, malware, and breaches, and that attackers automate validation against services [26]. These findings support a defense model that treats stolen credentials as a high-likelihood event and therefore focuses on reducing credential usefulness through phishing-resistant authentication and conditional access. Recent industry reports provide updated evidence of phishing scale and impact. The APWG reports document large quarterly volumes and changing tactics, such as QR-code phishing [1–7]. The Verizon DBIR provides breach-level context across industries [8]. IBM emphasizes breach cost, AI governance, and operational disruption [9]. ENISA threat landscape reporting provides a European perspective on threat evolution, including social engineering, identity attacks, and malicious use of emerging technologies [27]. Public URL datasets are important for reproducible research. The PhiUSIIL dataset provides 235,795 instances, 54 features, no missing values, and labels for legitimate and phishing URLs [15]. Its associated paper introduces a diverse security-profile approach and incremental learning for phishing URL detection [16]. Earlier dynamic and neuro-fuzzy phishing-detection work also confirms the value of adaptive feature-based detection; however, dataset classification alone cannot capture all identity risks created by adversary-in-the-middle kits and session-token theft [28]. Explainable machine learning is relevant because black-box security alerts are difficult to trust. SHAP and LIME approaches provide local explanations that can

show which features contributed to a model decision [29, 30]. These methods are useful for analyst triage and for reducing alert fatigue. However, explanations must be handled carefully; attackers should not receive detailed feedback that helps them evade detection. Generative AI changes both offense and defense. Transformer architectures introduced by Vaswani et al. made large language models practical at scale [31]. OWASP LLM guidance highlights risks such as prompt injection, insecure output handling, data leakage, and supply-chain exposure in LLM applications [10]. NIST AI RMF emphasizes governance, measurement, and risk management for AI systems [11]. These sources support the need for a phishing framework that detects AI-assisted lures while also governing defender-side AI tools. Zero trust and identity standards provide the second half of the solution. CISA recommends phishing-resistant MFA for high-value accounts [17]. NIST SP 800-63B defines authentication assurance and lifecycle principles. WebAuthn and FIDO guidance specify origin-bound public-key authentication [20, 21]. NIST CSF 2.0, ISO/IEC 27001, and CSA zero-trust identity guidance provide governance and control structures for organizational implementation [19], [32], [33]. Together, these sources show that phishing defense must be both data-driven and identity-centered.

Table 1. Comparison of related studies and gaps addressed by the proposed framework.

Study / Source	Main Focus	Limitation Addressed by This Paper
APWG reports [1–7]	Observed phishing trends and quarterly volumes	Do not provide a complete zero-trust implementation architecture
PhiUSIIL [15, 16]	URL and webpage feature dataset	Does not cover identity-policy response after detection
CISA/NIST/FIDO [17–21]	Phishing-resistant authentication	Does not provide URL/AI detection pipeline
OWASP LLM [10]	Risks in LLM applications	Needs linkage to phishing operations and controls
MITRE ATT&CK [22]	Attack technique mapping	Needs quantitative operational metrics

Overall, detection studies and identity standards address complementary gaps; this framework connects measurable classification evidence with explainable zero-trust enforcement and operational response.

2.1. RESEARCH GAP

The review reveals four gaps. First, many academic phishing studies focus on classification accuracy but do not connect detection to identity controls that prevent

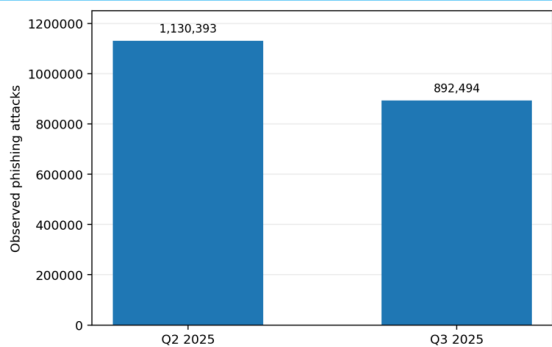


Figure 2. Q2 2025 and Q3 2025 phishing volume comparison derived from APWG reporting [6, 7].

credential replay. Second, many zero-trust and passkey discussions describe authentication improvements but do not show how suspicious URLs, QR codes, and AI-generated lures should be detected and explained. Third, current threat reports provide valuable statistics but usually do not translate them into a reproducible architecture. Fourth, few studies combine real threat trends, a public benchmark dataset, explainability, and incident-response governance in a single framework. This study addresses these gaps by combining data, architecture, deployment logic, and an explicit benchmark experiment. The experimental component is limited to URL/webpage classification on PhiUSIIL, while the zero-trust control mapping addresses identity, QR-code, reporting, and governance risks that cannot be measured by a URL dataset alone. This distinction prevents overstated claims while still providing measured results.

3. MATERIALS AND METHODS

This study uses a mixed methodology consisting of documentary analysis, descriptive statistics, threat modeling, framework design, and supervised benchmark experimentation. The primary external data sources are APWG trend reports and the PhiUSIIL public dataset. APWG reports provide quarterly observed phishing volumes, while PhiUSIIL provides labeled URL and webpage features for model evaluation [1–7], [15, 16]. The method contains seven stages: data-source selection, variable identification, preprocessing, supervised classification, threat taxonomy construction, framework design, and control mapping. The experiment used an 80/20 stratified train-test split, numeric feature encoding, and baseline classifiers implemented with scikit-learn [34]. The materials used in this study are summarized in Table 2. Threat-intelligence materials are used for trend analysis and attack-class selection. The public dataset is used to define measurable features and classification variables. Standards and guidance documents are used to select controls and define governance requirements. The explainability literature is used to justify why the framework presents feature-level evidence to analysts [29, 30].

Table 2. Research materials, sources, and their contributions to the study.

Material	Role in Study	Key Values / Contribution
APWG reports 2024-Q3 2025 [1–7]	Threat trend source	Quarterly observed phishing volumes and tactic changes
PhiUSIIL dataset [15]	Benchmark source	235,795 URLs, 54 features, no missing values
NIST / CISA / FIDO [17–21]	Control source	Phishing-resistant MFA, digital identity, CSF governance
MITRE ATT&CK [22]	Attack mapping	T1566 phishing and related credential-access tactics
SHAP / LIME [29, 30]	Explainability source	Local explanation of model decisions

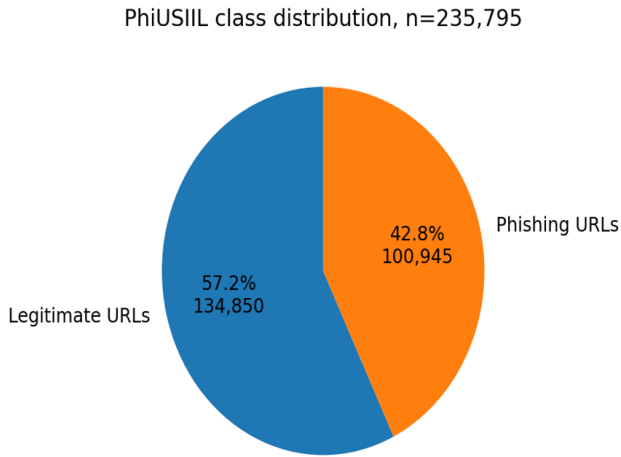


Figure 3. PhiUSIIL dataset class distribution: 134,850 legitimate URLs and 100,945 phishing URLs [15].

3.1. DATASET DESCRIPTION

The PhiUSIIL phishing URL dataset contains 235,795 URL and webpage instances. The UCI repository describes the dataset as tabular, with real, categorical, and integer feature types, 54 features, and no missing values. The dataset represents URLs and their corresponding webpages; label 1 corresponds to legitimate URLs and label 0 corresponds to phishing URLs [15]. The dataset includes variables such as URLLength, DomainLength, IsDomainIP, URLSimilarityIndex, CharContinuationRate, TLDLegitimateProb, URLTitleMatchScore, URLCharProb, and related source-code or webpage-derived features. These variables are suitable for detection because phishing pages often exhibit unusual domain structure, deceptive title alignment, abnormal character distributions, suspicious external references, and inconsistent brand signals [15, 16]. The dataset class balance is practical for experimentation: 57.2% le-

gitimate and 42.8% phishing. This balance is more useful than highly skewed datasets when evaluating precision, recall, and false-positive rate. However, real enterprise traffic may be far more imbalanced, with very few malicious events among many benign messages. Therefore, any operational deployment should evaluate both benchmark performance and production false-positive burden.

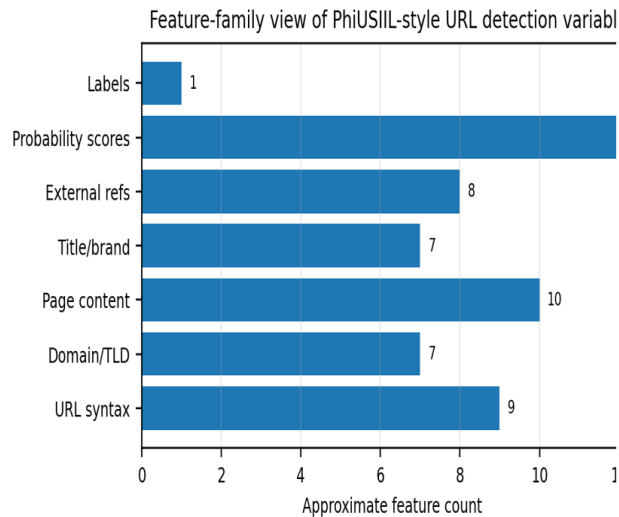


Figure 4. Feature-family view of PhiUSIIL-style variables used in URL and webpage phishing detection [15, 16].

3.2. DATA PROCESSING PIPELINE

The proposed pipeline begins by ingesting URLs, e-mail metadata, QR-code destinations, and webpage content. Duplicate indicators are removed, and redirect chains are expanded in a safe sandbox. For each URL or QR destination, the system extracts lexical features, domain features, reputation signals, page title similarity, form indicators, external resource counts, script behavior, and brand-mismatch signals. When the dataset is used for offline benchmarking, the same feature families are mapped to PhiUSIIL variables. The pipeline then calculates a risk score. A high score is not sufficient by itself; the decision must be accompanied by evidence. Evidence may include a newly registered domain, high URL entropy, suspicious TLD probability, mismatch between brand name and registered domain, login form hosted outside the expected identity provider, QR-code redirect chain, failed sender authentication, or a known phishing kit pattern. The explanation is shown to analysts, while user-facing messages remain simple and safe. The final stage applies the policy. Low-risk events may be delivered normally. Medium-risk events may be rewritten, detonated, or opened in isolation. High-risk events may be blocked, reported to security operations, and used to trigger credential-protection measures. For high-value accounts, the framework recommends phishing-resistant

authentication by default rather than password-only or SMS-based MFA [17–21].



Figure 5. Reproducible evaluation pipeline connecting data ingestion, feature extraction, explanation, policy action, and feedback.

3.3. EXPERIMENTAL SETUP AND EVALUATION METRICS

Table 3. Held-out test performance and confusion-matrix results for the three baseline classifiers.

Model	Held-out test metrics	Error profile
Decision Tree	Accuracy 100.00%; Precision 100.00%; Recall 100.00%; F1 100.00%; ROC-AUC 100.00%	TN=26,970; FP=0; FN=0; TP=20,189
Random Forest	Accuracy 99.9979%; Precision 100.00%; Recall 99.9950%; F1 99.9975%; ROC-AUC 100.00%	TN=26,970; FP=0; FN=1; TP=20,188
Gaussian Naive Bayes	Accuracy 98.4245%; Precision 96.4948%; Recall 99.9505%; F1 98.1923%; ROC-AUC 99.9797%	TN=26,237; FP=733; FN=10; TP=20,179

For the benchmark experiment, the original PhiUSIIL labels were converted so that phishing represented the positive class. The full dataset contained 235,795 instances; 188,636 were used for training and 47,159 for testing. The experiment used 50 numeric predictive features after excluding identifiers and non-predictive fields. Precision, recall, F1-score, accuracy, ROC-AUC, and confusion-matrix counts were reported to provide measured outcomes. All preprocessing transformations, including numeric encoding and any data-dependent



preparation, were fitted exclusively on the training split and then applied unchanged to the held-out test split to prevent data leakage. The very high tree-based scores should be interpreted as benchmark results on a structured public dataset, not as a guaranteed production detection rate. Production traffic is more imbalanced, adversaries adapt, and institutional telemetry adds privacy and logging constraints. Therefore, the experiment validates the URL-detection component, while the zero-trust framework remains necessary for credential replay, QR-code, BEC, and governance risks.

4. THREAT MODEL AND ATTACK TAXONOMY

The threat model assumes an external attacker who can send e-mail, messages, QR codes, or collaboration invitations to users. The attacker may register domains, host phishing pages, use URL shorteners, generate fluent text with AI tools, clone legitimate login pages, deploy adversary-in-the-middle proxy kits, and attempt to capture credentials or session tokens. The attacker may also target finance and administrative workflows through business e-mail compromise. The defender is assumed to control e-mail security tools, DNS filtering, endpoint browser policies, identity-provider policies, MFA methods, user training, reporting channels, and security monitoring. The defender may not control every personal device or external sender. Therefore, the framework prioritizes controls that remain effective when the initial message reaches the user. The model excludes destructive malware engineering and offensive instructions. It focuses on detection, prevention, and response. QR-code analysis is performed through safe detonation and redirect expansion, not through scanning user devices. Deepfake discussions are limited to defensive recognition and policy controls.

URL phishing
Deceptive links and credential-harvesting pages
Quishing (QR phishing)
QR-encoded links and redirect chains
AiTM proxy phishing
Session token theft through proxy sites
Business e-mail compromise
Invoice and payment-redirect fraud
Deepfake-assisted social engineering
AI-generated voice or impersonation lures

Figure 6. Taxonomy of AI-enhanced phishing and credential-theft attack classes.

4.1. ATTACK LIFECYCLE

The phishing lifecycle begins with reconnaissance. Attackers identify departments, suppliers, login portals,

branding, employee names, and seasonal workflows such as registration, payroll, procurement, or examination periods. Generative AI can accelerate this stage by summarizing public information and producing customized message variants [10, 11], [31]. The second stage is lure generation. The attacker creates an e-mail, SMS, chat message, QR poster, or fake collaboration invitation. AI-generated text can reduce grammatical errors and increase localization quality, making old awareness cues less reliable. The third stage is delivery through e-mail gateways, cloud-sharing platforms, compromised accounts, or physical QR placement. The fourth stage is interaction: the user clicks, scans, opens a link, or responds. The fifth stage is credential capture, token interception, consent abuse, invoice manipulation, or malware delivery. The sixth stage is monetization: account takeover, data theft, payment diversion, or lateral movement. Controls must be placed across the lifecycle rather than only at the delivery stage.

Reconnaissance
Target research and brand impersonation
Lure generation
AI-assisted phishing message creation
Delivery
E-mail, chat, shared files and QR posters
Interaction
Victim clicks, scans or responds
Credential / token exposure
Capture, consent abuse or redirection
Account misuse
Data theft, fraud or lateral movement
Defensive response
Block, revoke sessions and report

Figure 7. Phishing kill chain and defensive intervention opportunities [22].

4.2. STRIDE-BASED RISK INTERPRETATION

STRIDE analysis helps connect phishing to broader security consequences. Although phishing is often described as social engineering, its outcomes include spoofing, information disclosure, elevation of privilege, and sometimes tampering or denial of service. This reinforces the argument that phishing should be handled by identity, access, monitoring, and incident-response teams together. For example, adversary-in-the-middle phishing can capture session cookies even when a user completes traditional MFA. This is primarily an information-disclosure and elevation-of-privilege risk. FIDO2/WebAuthn passkeys reduce this risk because the authentication ceremony is bound to the legitimate origin, making replay to a fraudulent origin ineffective [20, 21].

Table 4. STRIDE-based phishing risk categories, examples, and recommended controls.

STRIDE Category	Phishing Example	Recommended Control
Spoofing	Fake identity-provider page	FIDO2/WebAuthn and brand/domain verification
Tampering	Modified invoice or redirect chain	Attachment sandboxing and payment verification
Repudiation	Attacker denies sending messages	Sender authentication and logging
Information Disclosure	Credential or token capture	Passkeys, token binding, browser isolation
Denial of Service	Account lockout through repeated attempts	Adaptive throttling and recovery policy
Elevation of Privilege	Compromised admin account	Privileged access management and conditional access

5. PROPOSED FRAMEWORK DESIGN AND IMPLEMENTATION

The proposed framework is built on seven layers: data ingestion, feature extraction, risk scoring, explanation, zero-trust policy enforcement, automated response, and governance feedback. The design is modular so that organizations can implement it incrementally. For example, a university may begin with reporting and URL detonation, then add passkeys for administrators, and finally integrate automated response. The key design principle is that detection and authentication must support each other. Detection reduces exposure by identifying suspicious links and messages. Authentication reduces the impact when detection fails. Explanation improves analyst trust and user education. Governance ensures that new phishing tactics are incorporated into policy rather than being treated as isolated incidents.

Proposed Explainable Zero-Trust Phishing Defense Framework

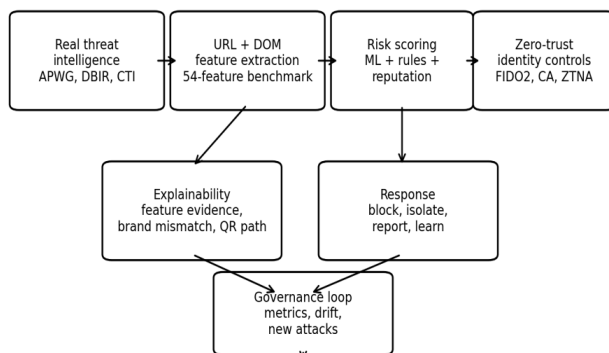


Figure 8. Architecture of the proposed explainable zero-trust phishing defense framework.

5.1. LAYER 1: THREAT DATA INGESTION

The first layer receives threat intelligence from APWG-style reports, internal user submissions, URL reputation feeds, DNS logs, web-proxy logs, identity-provider logs, QR-code scanning gateways, and security-operations alerts. In a small organization, this layer can begin with a shared reporting mailbox and DNS filtering logs. In a mature organization, it can integrate SIEM and SOAR platforms. Ingestion must normalize indicators. A single phishing campaign may use many URLs, redirects, domains, sender addresses, and brand names. Normalization removes duplicates, expands redirects, extracts the effective second-level domain, records hosting provider information, and stores timestamps. This enables the measurement of time-to-block and campaign evolution.

5.2. LAYER 2: FEATURE EXTRACTION

The feature extraction layer converts raw URLs and webpages into measurable variables. Lexical features include length, number of dots, special characters, digit ratio, entropy, and presence of suspicious words. Domain features include TLD, domain length, IP-address usage, registration age, and probability of legitimacy. Page features include title similarity, login form presence, external-resource count, hidden elements, script behavior, and brand mismatch [15, 16]. QR-code phishing requires additional handling. The QR image should be decoded in a sandbox. The resulting URL should be expanded through its redirect chain. Mobile-specific landing pages should be fetched using controlled user-agent profiles. The risk engine should store the original message, QR content, intermediate redirects, and final destination. This makes QR-based evidence visible to analysts.

Phishing e-mail or QR poster

User receives a malicious QR code

QR code scanned

Mobile browser opens the encoded link

Shortened URL or redirect

Destination is obscured by redirect hops

Fake sign-in page

Login form mimics a legitimate brand

Credential or token exposure

Attacker attempts to harvest secrets

Defensive control points

Decode, expand, isolate and verify origin

Figure 9. QR-code phishing flow and defensive control points.

5.3. LAYER 3: RISK SCORING

The risk score combines machine-learning signals, deterministic rules, and reputation information. A simplified formula is $R = w_1F_{url} + w_2F_{domain} + w_3F_{content} + w_4F_{brand} + w_5F_{sender} + w_6F_{context}$, where each F rep-



resents a normalized feature group and each weight is adjusted through validation. The formula is not intended to replace a trained model; rather, it documents how the framework combines different evidence sources. Risk categories are defined as follows: 0-29 low risk, 30-59 suspicious, 60-79 high risk, and 80-100 critical. Low-risk events are logged. Suspicious events may be rewritten or opened in isolation. High-risk events are blocked and reported. Critical events trigger identity checks, token revocation, and campaign-level responses.

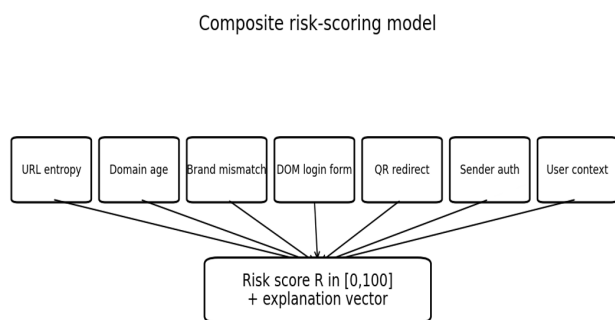


Figure 10. Composite risk-scoring model and explanation vector.

5.4. LAYER 4: EXPLAINABILITY

Explainability converts a numerical risk score into evidence. For machine-learning models, SHAP or LIME-style methods can identify features that influence the decision [29, 30]. For rule-based detections, explanations are generated directly from triggered conditions. Examples include: “URL title does not match domain,” “login form posts to external domain,” “QR code redirects through two shorteners,” and “sender authentication failed.” Explanations support three audiences. Analysts need detailed evidence for triage. End users need short warnings that do not overwhelm them. Managers need aggregate metrics showing which departments, brands, or workflows are most targeted. The framework therefore separates internal explanations from user-facing wording.

5.5. LAYER 5: ZERO-TRUST ENFORCEMENT

The zero-trust enforcement layer assumes that some phishing messages will bypass detection. Therefore, access to important resources must require strong, origin-bound authentication and contextual checks. CISA and FIDO guidance emphasize phishing-resistant MFA, while WebAuthn defines the browser API for public-key credentials [17], [20, 21]. Conditional access uses risk signals such as device compliance, location, impossible travel, sign-in frequency, user role, and detected phishing interaction. If a user clicks a suspicious link, the

identity provider can require step-up verification, block legacy authentication, revoke suspicious sessions, or restrict access to sensitive resources. Passkeys should be mandatory for administrators and finance users.

5.6. LAYER 6: AUTOMATED RESPONSE

Automated responses reduce the time between detection and containment. When a phishing URL is confirmed, the framework can remove matching messages from mailboxes, block DNS resolution, add the domain to web-proxy controls, revoke sessions for affected users, create a ticket, notify helpdesk staff, and submit the URL for takedown. These actions align with intrusion-detection and prevention principles that emphasize monitoring, analysis, containment, and auditability [35]. Automation should be risk-based. A false positive in a critical business workflow can disrupt operations. Therefore, the system should use human approval for ambiguous or high-impact actions, while allowing automatic blocking for high-confidence malicious infrastructure. This balances speed and safety.

5.7. LAYER 7: GOVERNANCE AND CONTINUOUS IMPROVEMENT

Governance ensures that the framework remains effective as attacker behavior changes. NIST CSF 2.0 emphasizes governance as a core function, while ISO/IEC 27001 emphasizes continual improvement and risk treatment [19], [32]. Governance activities include quarterly control reviews, model-drift checks, phishing simulation policy, passkey adoption targets, incident postmortems, and reporting to leadership. Model drift is especially important because phishing infrastructure and language change quickly. A model trained on old features may miss new QR-code flows, cloud-hosted forms, or AI-generated messages. Monitoring should track recall, false-positive rate, time-to-block, and analyst feedback over time.

6. RESULTS AND DISCUSSION

This section presents both descriptive threat results and measured benchmark results. The descriptive analysis summarizes APWG and PhiUSIIL evidence, while the experimental analysis reports the supervised classification performance on the PhiUSIIL dataset. The results are discussed as one component of a broader zero-trust defense because URL classification alone cannot prevent credential replay, QR-code bypass, or business-process fraud. The APWG data show a sustained high phishing volume across 2024 and 2025. The seven-quarter series used in this study includes 963,994 observed attacks in Q1 2024, 877,536 in Q2 2024, approximately 932,923 in Q3 2024, approximately 989,123 in Q4 2024, 1,003,924 in Q1 2025, 1,130,393 in Q2 2025, and 892,494 in Q3

2025 [1–7]. This pattern shows that organizations must maintain long-term controls rather than temporary awareness campaigns. The PhiUSIIL dataset provides an experimentally useful basis for URL classification. With 235,795 samples and 54 features, it is large enough to support supervised learning, feature analysis, and cross-validation [15, 16]. However, phishing defense cannot be measured only by URL classification. AiTM attacks and QR-code flows can succeed even when a user is trained, and password-based MFA can be bypassed through token capture. Therefore, the framework links detection results to identity controls.

6.1. DESCRIPTIVE AND EXPERIMENTAL RESULTS

The descriptive data support two observations. First, the phishing volume is high enough that organizations should assume continual exposure. Second, public datasets can support repeatable detection research. The experiment confirms that PhiUSIIL is suitable for benchmark classification; however, its strong results must be interpreted alongside the limitations of public URL datasets and the need for operational identity telemetry. The decline from Q2 2025 to Q3 2025 should not be interpreted as a permanent reduction. Quarterly changes may reflect campaign timing, reporting coverage, takedown activity, or shifts in attacker tactics. A resilient framework therefore tracks both volume and technique. The experimental results strengthen only the detection layer; passkeys, conditional access, response, and governance remain necessary to reduce credential exposure when detection fails [10], [17–22], [33], [35].

6.2. ATTACK-TO-CONTROL MAPPING

The mapping shows that no single control covers all attack classes. URL/DOM classification is strong for ordinary phishing links but weaker for business-email compromise, where the message may contain no malicious URL. DMARC and sender authentication help reduce spoofing but do not stop attacks sent from compromised legitimate accounts. Passkeys help against credential replay and AiTM proxying but do not prevent invoice fraud by themselves. This is why the proposed framework is layered. It treats controls as complementary. Detection finds suspicious events; passkeys limit credential replay; isolation reduces exposure from unknown pages; reporting improves intelligence; and governance ensures that lessons learned become policy. The combination is more practical than searching for a single perfect classifier.

6.3. FRAMEWORK EVALUATION MATRIX

The maturity matrix can be used as a self-assessment tool. A small institution may start at low maturity and se-

lect the highest-impact upgrades: user reporting, FIDO2 for privileged users, DNS filtering, and basic URL detonation. A larger institution can integrate identity logs, SOAR, browser isolation, and formal model monitoring. The matrix also supports comparative evaluation and future work because it defines measurable improvement paths. Researchers can compare institutions before and after deployment using the same dimensions: data coverage, detection, identity, automation, explainability, and governance. This makes the framework useful beyond a single dataset.

6.4. REPRODUCIBILITY AND VALIDATION DETAILS

The benchmark experiment can be reproduced by downloading the PhiUSIIL dataset, applying the same label conversion, using an 80/20 stratified split, and training the three baseline classifiers reported in Sections 3.3 and 6. The models were evaluated using precision, recall, F1-score, accuracy, ROC-AUC, and confusion-matrix counts. The setup uses standard supervised-learning functions available in scikit-learn [15], [34]. For future extension, the benchmark should be supplemented with institutional telemetry and a simulated identity-policy layer. Such a layer would classify which attacks are mitigated by FIDO2/WebAuthn, which require browser isolation, and which require business-process verification. Model drift should also be tested using newer phishing samples or time-based splits. This separation between measured classifier performance and control-effectiveness mapping supports a cautious interpretation. It reports actual benchmark results while avoiding the unsupported claim that high URL-dataset performance automatically eliminates credential-theft risk in production environments.

7. PRACTICAL DEPLOYMENT GUIDELINES

The framework can be implemented in four phases. Phase 1 establishes visibility: user reporting, DNS filtering logs, identity logs, and APWG-style threat tracking. Phase 2 improves authentication: passkeys for administrators, finance staff, and other high-risk users. Phase 3 adds detection intelligence: URL detonation, QR-code expansion, reputation feeds, and feature-based scoring. Phase 4 automates response and governance: mailbox removal, session revocation, quarterly metrics, and model-drift review. For Sana'a University-style academic environments, the recommended starting point is a targeted deployment focused on high-risk accounts and common university workflows. Examples include student portals, e-mail accounts, finance approvals, HR documents, and cloud collaboration links. A dedicated reporting button should send suspicious messages to a security mailbox or ticketing system. Every reported



message becomes training data for local improvement. The deployment roadmap is intentionally incremental. Many organizations fail when they attempt a full zero-trust transformation without a clear sequence. By starting with visibility and high-value accounts, security teams can demonstrate measurable improvement early and build support for later automation. Policy documentation should define acceptable actions for each risk category. For example, high-risk phishing URLs may be blocked automatically, whereas suspicious supplier invoices may require human verification. The policy should also define how to handle false positives, user privacy, evidence retention, and appeals.

7.1. STUDY LIMITATIONS AND ETHICAL CONSIDERATIONS

The study limitations are explicitly acknowledged. APWG data represent reported and aggregated phishing activity rather than the complete global population of attacks; the PhiUSIIL benchmark supports URL and webpage classification but does not fully represent BEC, physical QR-code placement, deepfake-assisted fraud, or every adversary-in-the-middle workflow; and the model-drift component remains a deployment requirement that should be validated with newer samples and local telemetry. Field evidence on passkey adoption is cited to support phishing-resistant authentication, but institutional impact still depends on recovery policy, device support, helpdesk readiness, and user rollout [1], [7], [11], [15], [23], [24]. This research is defensive. It does not provide instructions for creating phishing kits, bypassing authentication, or exploiting users. Any QR-code analysis should be conducted in a sandbox, and user-reported messages should be handled under privacy-preserving procedures. When real institutional data are used, sensitive content should be minimized or anonymized. User training should avoid blaming. Phishing succeeds because attackers exploit design weaknesses, urgency, trust, and cognitive load. A security culture improves when reporting is easy and users are thanked for reporting suspicious content. Punitive simulation programs may reduce reporting and damage trust. AI tools used by defenders should be governed. Sensitive e-mails and logs should not be pasted into public AI tools without approval. If an LLM is used for summarization or triage, the organization should enforce data controls, logging, access management, and output verification [10, 11].

8. CONCLUSION AND FUTURE WORK

This study presents an explainable zero-trust framework for detecting and reducing generative-AI-enhanced phishing and credential theft. The study used real APWG phishing-trend data from Q3 2024 to Q3 2025, the PhiUSIIL public dataset containing 235,795 URLs and 54

features, and a supervised benchmark experiment using decision tree, random forest, and Gaussian Naive Bayes classifiers. The proposed framework integrates threat intelligence, feature extraction, risk scoring, explainability, phishing-resistant authentication, conditional access, automated response, and governance. Its central argument is that modern phishing defense must reduce both the probability of user interaction and the impact of successful interaction. Passkeys and WebAuthn are important because they reduce the usefulness of stolen passwords, but they are most effective when combined with detection, reporting, and incident response. Overall, the findings show that benchmark URL classification and identity-centered safeguards address different parts of the phishing lifecycle and should be evaluated together. The framework translates this evidence into an incremental deployment pathway for universities and resource-constrained organizations, with measurable controls for detection quality, authentication resilience, response speed, explainability, and governance maturity. Future work should expand the benchmark by publishing code, adding cross-validation, testing time-based drift, and evaluating newer phishing samples. Additional work should collect local phishing reports from academic institutions, compare AI-generated and human-written phishing lures, evaluate QR-code detection at scale, and measure the real-world effect of passkey rollout on credential-theft incidents [11], [27], [33]. Deep-learning extensions using TensorFlow [36] may be evaluated in future work with cross-validation and leakage controls.

9. DECLARATIONS

Ethical Approval

This article does not contain any studies involving animals or human participants performed by the author. The manuscript is original and is not currently considered for publication elsewhere.

Conflict of Interest

The author declares no competing interests.

Competing Interests

The author has no financial or proprietary interests in any material discussed in this article.

Authors' Contributions

The author contributed to the study conception, design, data collection, analysis, manuscript preparation, and final approval.

Funding

No funding was received from any organization for this work.

Availability of Data and Materials

This study uses public reports and publicly accessible dataset descriptions. The PhiUSIIL dataset is available from the UCI Machine Learning Repository [15].

Consent for Publication

The author confirms consent for publication of the data contained in the article.

Consent to Participate

Not applicable.

REFERENCES

- [1] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 1st quarter 2024," Anti-Phishing Working Group (APWG), Tech. Rep., 2024.
- [2] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 2nd quarter 2024," Anti-Phishing Working Group (APWG), Tech. Rep., 2024.
- [3] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 3rd quarter 2024," Anti-Phishing Working Group (APWG), Tech. Rep., 2024.
- [4] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 4th quarter 2024," Anti-Phishing Working Group (APWG), Tech. Rep., 2025.
- [5] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 1st quarter 2025," Anti-Phishing Working Group (APWG), Tech. Rep., 2025.
- [6] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 2nd quarter 2025," Anti-Phishing Working Group (APWG), Tech. Rep., 2025.
- [7] Anti-Phishing Working Group (APWG), "Phishing activity trends report: 3rd quarter 2025," Anti-Phishing Working Group (APWG), Tech. Rep., 2025.
- [8] Verizon, "2025 data breach investigations report," Verizon Business, Tech. Rep., 2025.
- [9] IBM Security, "Cost of a data breach report 2025," IBM Corporation, Tech. Rep., 2025.
- [10] OWASP Foundation, "Owasp top 10 for large language model applications," OWASP, Tech. Rep., 2025.
- [11] National Institute of Standards and Technology (NIST), "Artificial intelligence risk management framework (ai rmf 1.0)," National Institute of Standards and Technology, Tech. Rep. NIST AI 100-1, 2023.
- [12] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2006, pp. 581–590.
- [13] S. Sheng et al., "Who falls for phish? a demographic analysis of phishing susceptibility and effectiveness of interventions," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010.
- [14] M. Jakobsson and S. Myers, *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*. Wiley, 2007.
- [15] A. Prasad and S. Chandra, *PhiUSIIL phishing URL (website) dataset*, UCI Machine Learning Repository, 2024.
- [16] A. Prasad and S. Chandra, "PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning," *Comput. & Secur.*, 2024.
- [17] Cybersecurity and Infrastructure Security Agency (CISA), "Implementing phishing-resistant MFA," CISA, Tech. Rep., 2022.
- [18] National Institute of Standards and Technology, "Digital identity guidelines: Authentication and lifecycle management," National Institute of Standards and Technology, Gaithersburg, MD, Tech. Rep. NIST SP 800-63B, 2017, Updated online.
- [19] National Institute of Standards and Technology, "The NIST cybersecurity framework 2.0," NIST, Tech. Rep., 2024.
- [20] World Wide Web Consortium (W3C), *Web authentication: An api for accessing public key credentials*, W3C Recommendation, 2021.
- [21] FIDO Alliance, *Passkey and FIDO2 deployment guidance for phishing-resistant authentication*, FIDO Alliance, 2025.
- [22] MITRE, *ATT&CK enterprise technique T1566: Phishing*, MITRE ATT&CK Knowledge Base, Accessed 2026, 2026.
- [23] Google Security Blog, *Passkeys reach a milestone and what is next*, Google, 2024.
- [24] Microsoft, *Passkeys and WebAuthn in microsoft entra id*, Microsoft Learn, 2025.
- [25] A. Oest et al., "Inside a phisher's mind: Understanding the anti-phishing ecosystem through phishing kit analysis," in *Proceedings of the USENIX Security Symposium*, 2018.
- [26] K. Thomas et al., "Data breaches, phishing, or malware? understanding the risks of stolen credentials," in *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 2017.
- [27] European Union Agency for Cybersecurity (ENISA), "ENISA threat landscape 2025," ENISA, Tech. Rep., 2025.
- [28] A. Almomani et al., "Phishing dynamic evolving neural fuzzy framework for online detection," *J. Comput. Sci.*, vol. 9, no. 6, pp. 662–672, 2013.
- [29] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [30] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you? explaining the predictions of any classifier," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016.
- [31] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [32] ISO/IEC, *ISO/IEC 27001:2022 information security management systems*, International Organization for Standardization and International Electrotechnical Commission, 2022.
- [33] Cloud Security Alliance, "Zero trust guidance for identity and access management," Cloud Security Alliance (CSA), Tech. Rep., 2024.
- [34] F. Pedregosa et al., "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [35] K. Scarfone and P. Mell, "Guide to intrusion detection and prevention systems (idps)," National Institute of Standards and Technology, Tech. Rep. NIST SP 800-94, 2007.
- [36] M. Abadi et al., *TensorFlow: Large-scale machine learning on heterogeneous systems*, Google Research, Software available from the TensorFlow project, 2016.