Scientific Research Publishing

Journal of Information Security, 2026, 17(2), 167-187
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

Robustness, Cost, and Attack-Surface Concentration in Phishing Detection

Julian Allagan*, Mohamed Elbakary, Zohreh Safari, Weizheng Gao, Gabrielle Morgan, Essence Morgan, Vladimir Deriglazov

Department of Mathematics, Computer Science, and Engineering Technology, Elizabeth City State University, Elizabeth City, NC, USA
Email: *adallagan@ecsu.edu, melbakary@ecsu.edu, zosafari@ecsu.edu, wegao@ecsu.edu

How to cite this paper: Allagan, J., Elbakary, M., Safari, Z., Gao, W., Morgan, G., Morgan, E. and Deriglazov, V. (2026) Robustness, Cost, and Attack-Surface Concentration in Phishing Detection. *Journal of Information Security*, **17**, 167-187.
https://doi.org/10.4236/jis.2026.172009

Received: March 21, 2026
Accepted: April 24, 2026
Published: April 27, 2026

Abstract

Phishing detectors built on engineered website features attain near-perfect accuracy under i.i.d. evaluation, yet deployment security depends on robustness to post-deployment feature manipulation. We study this gap through a cost-aware evasion framework that models discrete, monotone feature edits under explicit attacker budgets. Three diagnostics are introduced: minimal evasion cost (MEC), the evasion survival rate $S(B)$, and the robustness concentration index (RCI). On the UCI Phishing Websites benchmark (11,055 instances, 30 ternary features), Logistic Regression, Random Forests, Gradient Boosted Trees, and XGBoost all achieve $\mathrm{AUC} \geq 0.979$ under static evaluation. Under budgeted sanitization-style evasion, robustness converges across architectures: the median MEC equals 2 with full features, and over 80% of successful minimal-cost evasions concentrate on three low-cost surface features. Feature restriction improves robustness only when it removes all dominant low-cost transitions. Under strict cost schedules, infrastructure-leaning feature sets exhibit 17% - 19% infeasible mass for ensemble models, while the median MEC among evadable instances remains unchanged. We formalize this convergence: if a positive fraction of correctly detected phishing instances admit evasion through a single feature transition of minimal cost $c_{\min}$, no classifier can raise the corresponding MEC quantile above $c_{\min}$ without modifying the feature representation or cost model. Adversarial robustness in phishing detection is governed by feature economics rather than model complexity.

Keywords

Robustness

1. Introduction

Phishing detection is inherently adversarial. Attackers adapt observable website characteristics to evade classification, while defenders evaluate models under static train–test splits. A classifier may achieve near-perfect held-out accuracy yet remain operationally fragile when its predictions rest on surface-level attributes alterable at low cost. Recent studies report detection accuracies exceeding 95% using refined feature engineering and ensemble methods [1]-[4], but these results assume a passive threat model in which adversarial adaptation is excluded. In deployed systems, this assumption rarely holds [5]-[7].

The tension is acute because manipulation costs are asymmetric. Presentation-layer cues—URL structure, HTML artifacts, certificate presentation—are inexpensive to modify, whereas infrastructure-coupled signals such as domain age, DNS records, and traffic require sustained investment or third-party validation [8]-[11]. Many robustness analyses rely on continuous perturbation models that abstract away discrete feature semantics, or adopt worst-case threat models that ignore economically plausible attacker behavior [12]-[14].

Phishing robustness studies fall broadly into three categories. Continuous perturbation approaches apply ℓ_p-bounded adversarial examples to feature vectors, treating each coordinate as a real-valued input amenable to gradient-based attack [12] [15]. While methodologically convenient, this abstraction obscures the discrete, semantically constrained nature of website feature edits. Heuristic attack studies evaluate classifiers against hand-crafted manipulation strategies without formalizing attacker cost or optimality [1] [3] [16]. Problem-space constraint work emphasizes that adversarial perturbations must satisfy domain constraints, yet typically does not assign explicit economic costs to individual transitions or characterize the resulting attack-surface structure [6] [13] [17].

We adopt a complementary perspective. Evasion is formulated as exact shortest-path search over a cost-weighted discrete transition graph. We introduce concentration diagnostics—RCI and FirstTop1—as primary structural indicators, and establish an architecture-independent cost-floor bound. The combination of exact discrete-cost optimization, attack-surface concentration measurement, and a formal robustness ceiling distinguishes this work from prior analyses.

To bridge the gap between static evaluation and adversarial deployment, we develop a cost-aware adversarial evaluation framework that assigns explicit costs to discrete feature edits and evaluates classifiers under bounded attacker budgets. We study sanitization-style evasion under monotone edits, where the attacker removes phishing indicators and pushes feature values toward legitimate states. This threat model represents a lower bound on attacker capability: it excludes anti-feature injection, extractor-level attacks, and non-monotone manipulation, all of which can only expand the feasible action set and reduce MEC. The restriction to monotone edits is operationally motivated by empirical evidence that most phishing campaigns are short-lived (24 - 72 hours), favoring indicator removal over infrastructure construction [10] [11]. Section 7 discusses how relaxing this re-

striction would affect concentration and the cost floor.

Rather than measuring aggregate degradation, we address a structural question central to defensive design: under budget-constrained manipulation, do evasion pathways disperse across many features or collapse onto a small attack surface? The answer determines whether architectural complexity redistributes adversarial risk or leaves dominant failure modes intact.

We operationalize this analysis with three diagnostics. Minimal evasion cost (MEC) is the smallest cumulative cost required to induce misclassification for a correctly detected phishing instance. The evasion survival rate $S(B) = \Pr(\mathrm{MEC}(x) > B)$ measures resistance to the attacker's budget B. The robustness concentration index (RCI) quantifies whether successful minimal-cost edits are diffuse or concentrated on a small subset of features. Empirically, across models and full feature sets, evasion succeeds under modest budgets (median MEC = 2), and more than 80% of traces concentrate on three low-cost surface features. We formalize this convergence with a structural result: when a nontrivial fraction of instances admit evasion via a single feature transition of minimal cost, no classifier architecture can raise the corresponding robustness quantiles without modifying the feature space or cost model. We term this action-set-limited invariance.

2. Methods

We model post-deployment evasion as a shortest-path problem on a directed graph whose nodes are discrete feature vectors, whose edges represent admissible monotone manipulations, and whose edge weights encode attacker cost.

2.1. Threat Model

Let $f : \mathcal{X} \to \{-1, +1\}$ denote a deployed classifier (−1 for phishing, +1 for legitimate). Given a phishing instance x correctly classified as malicious, the attacker seeks x' with $f(x') = +1$ subject to a finite manipulation budget B. Feature vectors lie in $\{-1, 0, +1\}^d$, encoding phishing-indicative, neutral, and legitimate states. Edits are monotone: a transition $v \to v'$ is admissible only if $v' \geq v$ under the ordering $-1 < 0 < +1$. Reverse transitions incur infinite cost. This models sanitization-style evasion in which attackers remove suspicious indicators rather than inject adversarial anti-features. The attacker possesses feature-level knowledge—awareness that detection relies on discrete website features and coarse understanding of surface versus infrastructure cost asymmetries, consistent with publicly documented detection pipelines [6] [18]—but has no access to model parameters, training data, confidence scores, or gradients.

This threat model constitutes a lower bound on attacker capability. Non-monotone edits (injecting benign artifacts), extractor-level manipulation (exploiting parser ambiguities to alter computed features without semantic change), and anti-feature attacks all enlarge the feasible action set. Any such enlargement can only decrease MEC and potentially increase concentration. The cost floor established

under monotone edits therefore provides a conservative bound: if robustness is fragile under sanitization-only attackers, it is at least as fragile under more capable adversaries.

We compute MEC values via uniform-cost search (**Algorithm 1**), yielding exact shortest paths within the prescribed budget. Exact MEC represents an upper bound on evasion efficiency under the defined action set: query-limited attackers may fail to discover optimal evasions, raising empirical survival rates, but the structural cost floor persists whenever low-cost transitions remain available.

Algorithm 1. Minimal evasion cost via uniform-cost search.

Require: Classifier f, instance x, cost function c, budget $B_{\max}$
Ensure: MEC(x) or ∞ if no evasion within budget

if $f(x) = +1$ **then**
  return 0
end if
$Q \leftarrow \{(0, x)\}$, $V \leftarrow \emptyset$
while $Q \neq \emptyset$ **do**
  $(c_{\text{curr}}, x_{\text{curr}}) \leftarrow Q$.pop_min()
  if $x_{\text{curr}} \in V$ **then**
    continue
  end if
  $V \leftarrow V \cup \{x_{\text{curr}}\}$
  if $f(x_{\text{curr}}) = +1$ **then**
    return c_{curr}
  end if
  for each feature $j \in \{1, \ldots, d\}$ **do**
    for each target value $v' > x_{\text{curr}}[j]$ in $\{0, 1\}$ **do**
      $c_{\text{edit}} \leftarrow c(j, x_{\text{curr}}[j] \to v')$
      if $c_{\text{curr}} + c_{\text{edit}} \leq B_{\max}$ **then**
        $x' \leftarrow x_{\text{curr}}$ with $x'[j] \leftarrow v'$
        push $(c_{\text{curr}} + c_{\text{edit}}, x')$ to Q
      end if
    end for
  end for
end while
return ∞

2.2. Cost Schedules

Each admissible edit $(j, v \to v')$ incurs nonnegative cost $c(j, v \to v')$. For an instance x transformed to x', the cumulative cost is additive:

$$C(x \to x') = \sum_{(j, v \to v') \in \Delta(x, x')} c(j, v \to v').$$

The feasible action set at budget B is $\mathcal{A}_B(x) = \{x' \in \mathcal{X} : C(x \to x') \leq B\}$.

Costs represent dimensionless operational friction—the difficulty of effecting a manipulation within a phishing campaign's operational window—rather than direct monetary expenditure. We calibrate using a time-to-effect principle: one cost unit corresponds to a manipulation executable within a single day by the campaign operator; four units correspond to changes requiring multi-week external accumulation (DNS propagation, organic traffic growth, reputation accrual). This calibration reflects documented campaign lifecycles: Oest *et al.* [10] report median campaign durations under 21 hours, with 95% retired within 72 hours, establish-

ing surface-level edits as effectively "free" within the operational window and infrastructure changes as largely infeasible. Bijmans *et al.* [11] corroborate this timeline for phishing-kit deployments.

What matters for the structural conclusions is the cost ordering—surface features are strictly cheaper than infrastructure features—rather than exact magnitudes. Section 5 demonstrates that proportional cost scaling shifts the median MEC linearly while preserving feature ordering, concentration structure, and architecture invariance.

We consider two schedules. The base schedule assigns low cost to surface features (URL structure, HTML presentation; $c=1,2$) and higher cost to infrastructure features (domain age, DNS, traffic, reputation; $c=4,8$). The strict schedule coincides with the base except that infrastructure-feature upgrades to the fully legitimate state are disallowed ($c=\infty$), modeling horizons in which complete infrastructure legitimization is infeasible. **Table 1** summarizes both schedules. All experiments use $B_{\max}=18$. Appendix provides a complete mapping of all 30 UCI features to their cost group, along with a one-line time-to-effect rationale for each assignment.

Table 1. Feature manipulation cost schedules with operational time horizons. Costs are calibrated by the time-to-effect principle: 1 unit ≈ changes feasible within a day; 4 units ≈ multi-week external accumulation.

Feature Group	Examples	Base Schedule			Strict Schedule			Time Horizon
		$-1\to 0$	$-1\to 1$	$0\to 1$	$-1\to 0$	$-1\to 1$	$0\to 1$	
Surface	URL_of_Anchor, SFH, Prefix_Suffix, SSLfinal_State	1	2	1	1	2	1	hours to days
Semi-domain	Domain_Reg_Length, Google_Index	3	6	3	3	6	3	days to weeks
Infrastructure	web_traffic, DNSRecord, age_of_domain, Page_Rank	4	8	4	4	∞	∞	weeks to months

SSLfinal_State is classified as surface-level because certificate presentation can be modified through front-end configuration (e.g., deploying a free DV certificate via Let's Encrypt), without sustained infrastructure investment. Under the time-to-effect principle, this operation falls within the single-day horizon. Reclassifying SSLfinal_State as semi-domain is examined in the sensitivity analysis.

2.3. Dataset, Models and Conditioning

We use the UCI Phishing Websites benchmark [19]: 11,055 instances described by 30 ternary features in $\{-1,0,+1\}$, with 4898 phishing and 6157 legitimate websites. A stratified 75/25 train–test split (seed 1337) yields 2764 test instances including 1225 phishing samples. Four classifier families are evaluated: Logistic Regression (ℓ_2 regularization, $C=1.0$), Random Forests (100 trees, max depth 10),

Gradient Boosted Decision Trees (100 estimators, learning rate 0.1, max depth 6), and XGBoost with matched hyperparameters. Classification uses a fixed threshold of 0.5 on each model's native predict_proba output, held constant across all models and configurations; threshold sensitivity is examined in Section 5. Implementations use scikit-learn 1.0.2 and xgboost 1.5.0.

Robustness is evaluated on the conditioning set $\mathcal{P}_0 = \{x \in \mathcal{P}_{\text{test}} : f(x) = -1\}$, the phishing instances correctly classified as malicious. Conditioning isolates post-detection evasion and separates robustness from baseline classification error. The per-model sizes of $\mathcal{P}_0$ are: Logistic Regression 1103, Random Forest 1118, GBDT 1142, and XGBoost 1168 (out of 1225 phishing test instances). The cross-model intersection—instances correctly detected by all four models—contains 1047 instances. For cross-model comparisons, we intersect $\mathcal{P}_0$ across all four models and uniformly sample $n_{\text{eval}} = 300$ instances. This sample size was chosen so that 95% bootstrap confidence intervals on median MEC achieve width at most 1 and those on RCI_3 achieve width at most 0.06 under the observed distributions; it represents approximately 29% of the intersection, well above the threshold for stable quantile estimation at the precision reported. To verify stability, Table 2 reports mean and standard deviation of median MEC and RCI_3 across 10 independent random draws of 300 instances from the intersection. Table 3 verifies that this intersection does not bias conclusions by comparing metrics on each model's full $\mathcal{P}_0$ with the intersection sample.

Table 2. Stability of robustness metrics across 10 independent random 300-instance draws from the cross-model intersection (Full/base). Standard deviations confirm that conclusions are not sensitive to the specific subsample chosen.

Model	Mean Med. MEC	Std	Mean RCI_3	Std
Logistic Regression	2	0.00	0.961	0.011
Random Forest	2	0.00	0.843	0.018
GBDT	2	0.00	0.891	0.015
XGBoost	2	0.00	0.819	0.017

Table 3. Robustness metrics on full $\mathcal{P}_0$ versus the 300-instance cross-model intersection (Full/base).

Model	Median MEC		RCI_3	
	Full $\mathcal{P}_0$	Intersection	Full $\mathcal{P}_0$	Intersection
Logistic Regression	2	2	0.96	0.96
Random Forest	2	2	0.85	0.84
GBDT	2	2	0.89	0.89
XGBoost	2	2	0.82	0.82

We evaluate six feature configurations. The full set contains all 30 features. AAS (Accuracy-Anchored Subset)-12a ($d=12$) and AAS-11b ($d=11$) retain features with the highest mutual-information scores with the target label, selected by a greedy forward-selection procedure on the training set. RA (Robustness-Anchored)-8 ($d=8$) emphasizes infrastructure-leaning signals but retains SSLfinal_State to represent the dominant surface bottleneck. VA (Vulnerability-Anchored)-8a ($d=8$) and VA-7b ($d=7$) contain only presentation-layer features. Table 4 lists the exact features included in each subset. In all cases $c_{\min} \in \{1,2\}$, so by Proposition 3.1, median MEC cannot exceed these values unless all transitions at that cost are removed.

Table 4. Feature subset definitions. AAS = Accuracy-Anchored Subset; RA = Robustness-Anchored; VA = Vulnerability-Anchored.

Subset	d	Features included
AAS-12a	12	SSLfinal_State, URL_of_Anchor, having_Sub_Domain, age_of_domain, web_traffic, DNSRecord, Page_Rank, Domain_registeration_length, Google_Index, Request_URL, Links_in_tags, Prefix_Suffix
AAS-11b	11	Same as AAS-12a, dropping Links_in_tags
RA-8	8	age_of_domain, DNSRecord, web_traffic, Page_Rank, Google_Index, Domain_registeration_length, having_Sub_Domain, SSLfinal_State
VA-8a	8	URL_of_Anchor, SFH, Prefix_Suffix, having_At_Sign, HTTPS_token, Request_URL, Iframe, on_mouseover
VA-7b	7	URL_of_Anchor, SFH, Prefix_Suffix, having_At_Sign, HTTPS_token, Request_URL, Iframe

2.4. Robustness Metrics

For each $x \in \mathcal{P}_0$, the minimal evasion cost is

$$\mathrm{MEC}(x) = \inf\{C(x \to x') : f(x') = +1\},$$

computed exactly via uniform-cost search (Algorithm 1). The search is complete up to $B_{\max}$ and returns $\mathrm{MEC}(x) = \infty$ when no evasion exists within budget. Median runtime is 0.3 s per instance for full feature sets.

Resistance at attacker budget B is summarized by the evasion survival rate $S(B) = \Pr(\mathrm{MEC}(x) > B \mid x \in \mathcal{P}_0)$. Aggregate robustness is captured by the feature robustness index

$$\mathrm{FRI} = \frac{1}{B_{\max}} \int_0^{B_{\max}} S(B)\,\mathrm{d}B,$$

approximated by a left Riemann sum over integer budgets. FRI measures the expected fraction of the budget range over which a randomly selected instance from

$\mathcal{P}_0$ resists evasion; equivalently, it is the normalized area under the survival curve up to $B_{\max}$. FRI incorporates infeasible mass (instances with $\mathrm{MEC}=\infty$ contribute $S(B)=1$ for all B), while median and quartile MEC are computed over finite values only. This separation distinguishes overall resistance from the cost distribution among evadable instances.

To examine attack-surface structure, let N_j denote the total number of edits applied to feature j across all successful minimal-cost traces. The robustness concentration index is

$$\mathrm{RCI}_k = \frac{\sum_{j\in\text{Top-}k} N_j}{\sum_{j=1}^{d} N_j},$$

measuring the fraction of adversarial effort concentrated on the k most frequently edited features. When multiple optimal paths share identical cost, deterministic priority-queue tie-breaking selects a canonical trace. This affects the representative path used for concentration metrics but not MEC itself; recomputing RCI_3 under 10 randomized tie-break orders yields standard deviation below 0.02 in all configurations.

To isolate first-step bottlenecks, let $j_1(x)$ denote the first-edited feature in the canonical minimal-cost trace. Define $j^\star = \operatorname{argmax}_j \left|\{x : j_1(x)=j\}\right|$. The FirstTop1 index is

$$\mathrm{FirstTop1} = \frac{\left|\{x : j_1(x)=j^\star\}\right|}{\left|\{x : \mathrm{MEC}(x)<\infty\}\right|},$$

capturing single-feature bottlenecks at the initial decision step.

Figure 1 summarizes the evaluation pipeline and robustness diagnostics.

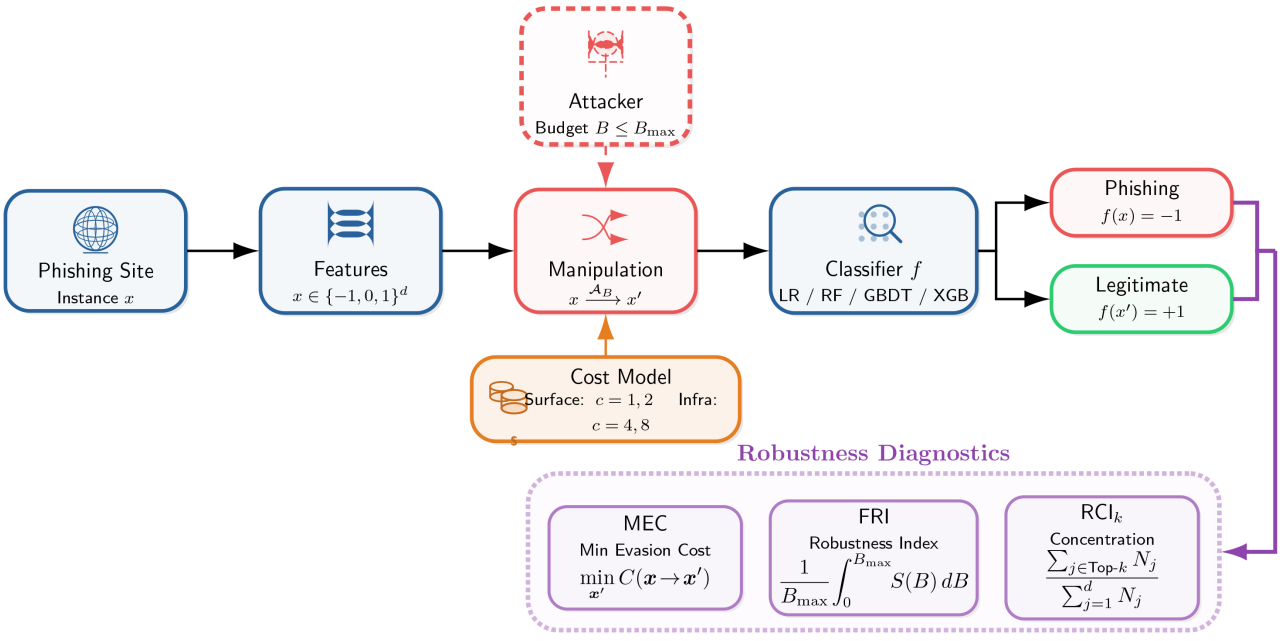

Figure 1. Cost-aware adversarial robustness framework with MEC, survival curves, and attack-surface concentration.

2.5. Query-Limited Greedy Attacker

To assess whether exact MEC meaningfully bounds realistic attacker efficiency, we also evaluate a query-limited greedy adversary. At each step, the attacker enumerates all admissible single-feature monotone edits from the current configuration x_{curr} and selects the edit $\left(j^{\star}, v \to v'\right)$ that produces the largest increase in the classifier's output score toward the legitimate class, $\hat{p}\left(x'\right) - \hat{p}\left(x_{\text{curr}}\right)$, breaking ties in favor of the lowest-cost edit. Each call to the classifier's score function counts as one query; evaluating all $O(d)$ candidate edits at a given step therefore consumes $O(d)$ queries. The attacker halts when $f\left(x'\right) = +1$ (evasion succeeds), when the cumulative manipulation cost exceeds $B_{\max}$ (budget exhausted), or when the total query count reaches $Q_{\max} \in \{50, 100, 500\}$, whichever occurs first. Because the greedy attacker follows a myopic per-step criterion and never revisits previously explored configurations, it may terminate without discovering a feasible evasion even when one exists, yielding empirically higher survival rates than exact MEC. Table 9 reports these comparisons.

3. Feature Economics and Robustness Limits

We now establish a structural limit imposed by feature-level manipulation costs. The result identifies a cost floor that bounds achievable robustness independently of model architecture.

Proposition 3.1 (Cost floor). *Let* $c_{\min} = \min_{j,v,v'} c\left(j, v \to v'\right)$ *be the minimum cost among all admissible single-feature transitions. Fix a classifier* f *and let* $\mathcal{P}_0$ *denote the set of phishing instances correctly detected by* f*. If a fraction* $\alpha > 0$ *of instances in* $\mathcal{P}_0$ *admit evasion via a single transition of cost* c_{min}*, then*

$$\Pr\left(\text{MEC} \le c_{\min} \mid x \in \mathcal{P}_0\right) \ge \alpha.$$

In particular, if $\alpha \ge \frac{1}{2}$*, then* $\text{median}\left(\text{MEC}\right) \le c_{\min}$*. Hence the* α *-quantile of the MEC distribution cannot exceed* $c_{\min}$ *without modifying the feature space or cost schedule.*

Proof. For each $x \in \mathcal{P}_0$ admitting a single-feature evasion $\left(j, v \to v'\right)$ of cost $c_{\min}$, one has $\text{MEC}\left(x\right) \le c_{\min}$ by definition of the infimum. Since these instances constitute at least an α fraction of $\mathcal{P}_0$, the distributional bound follows directly. The median statement is an immediate consequence of the definition: when $\alpha \ge \frac{1}{2}$, at least half the probability mass lies at or below $c_{\min}$. □

The force of Proposition 3.1 is not in the proof technique—which is elementary—but in the structural invariance it implies. Regardless of how a classifier partitions feature space, any instance that lies within a single cheap transition of a legitimate-classified region is evadable at cost $c_{\min}$. Whether that fraction α is large depends on the interaction between the cost landscape and the classifier's decision boundary, and the empirical contribution of this work is to show that α is indeed large across all tested architectures.

Corollary 3.1 (Action-set-limited invariance). *Fix* $\mathcal{X} = \{-1, 0, +1\}^d$ *and a*

monotone cost function c *with minimum transition cost* $c_{\min}$. *Let* $\{f_1, \cdots, f_K\}$ *be classifiers evaluated on a common conditioning set* $\mathcal{P}_0$, *and suppose that for each* f_k *at least an* $\alpha > 0$ *fraction of* $\mathcal{P}_0$ *admits single-transition evasion at cost* $c_{\min}$. *Then for every* k:

$$\Pr\left(\mathrm{MEC}_{f_k} \le c_{\min} \mid x \in \mathcal{P}_0\right) \ge \alpha.$$

Architectural variation alone cannot exceed this bound. Invariance breaks when the feature representation changes (*removing or hardening features*), *when the cost schedule is modified* (*raising* $c_{\min}$), *or when the feature extractor is made robust to manipulation* (*reducing the attacker's effective action set*).

Proof. Under a common action set and shared conditioning set, Proposition 3.1 applies identically to each f_k. □

Table 5 reports the empirical fraction $\hat{\alpha}_{c_{\min}}$ per classifier and feature configuration, confirming that the cost floor binds in practice and that the invariance argument is supported individually for each architecture rather than only in aggregate.

Table 5. Empirical mass at the cost floor $\hat{\alpha}_{c_{\min}} = \Pr\left(\mathrm{MEC} \le c_{\min} \mid x \in \mathcal{P}_0\right)$, reported per classifier and feature configuration. The architecture-invariance argument of Corollary 3.1 is supported individually across all models.

Feature set	Schedule	$c_{\min}$	$\hat{\alpha}$ (LR)	$\hat{\alpha}$ (RF)	$\hat{\alpha}$ (GBDT)	$\hat{\alpha}$ (XGB)	Med. MEC
Full	base	1	0.31	0.29	0.30	0.28	2
Full	strict	1	0.33	0.31	0.32	0.30	2
RA-8	base	1	0.28	0.27	0.28	0.27	2
RA-8	strict	1	0.15	0.06	0.07	0.05	2
VA-7b	base	1	0.62	0.61	0.60	0.60	1

4. Results

Table 6 reports held-out classification performance on the full feature set. All models achieve strong discrimination (AUC between 0.979 and 0.995), suggesting reliable deployment under static evaluation. The adversarial analysis below demonstrates that this conclusion does not survive once feature manipulation is permitted.

Table 6. Held-out classification performance (Full feature set, threshold = 0.5).

Model	Accuracy	AUC	Phishing TPR
Logistic Regression	0.927	0.979	0.900
Random Forest	0.950	0.993	0.913
GBDT	0.953	0.990	0.932
XGBoost	0.965	0.995	0.953

Table 7 presents the central robustness results. Two regularities dominate across all configurations.

Table 7. Robustness across feature sets and schedules. NoEvasion reports infeasible mass within $B_{\max} = 18$.

Features	Sched.	Model	Acc	FRI	MEC	[Q1, Q3]	RCI_3	FT1	NoEv
Full	base	Logit	0.927	0.076	2	[1, 2]	0.961	0.850	0%
Full	base	RF	0.950	0.092	2	[2, 3]	0.843	0.580	0%
Full	base	GBDT	0.953	0.076	2	[2, 2]	0.892	0.370	0%
Full	base	XGB	0.965	0.092	2	[2, 3]	0.815	0.440	0%
Full	strict	Logit	0.927	0.077	2	[1, 2]	0.975	0.847	0%
Full	strict	RF	0.950	0.093	2	[2, 3]	0.843	0.540	0%
Full	strict	GBDT	0.953	0.075	2	[2, 2]	0.854	0.397	0%
Full	strict	XGB	0.965	0.091	2	[2, 3]	0.784	0.413	0%
RA-8	base	Logit	0.869	0.081	2	[1, 2]	1.00	0.993	0%
RA-8	base	RF	0.900	0.104	2	[2, 2]	0.972	0.973	0%
RA-8	base	GBDT	0.899	0.091	2	[2, 2]	1.00	0.993	0%
RA-8	base	XGB	0.904	0.096	2	[2, 2]	0.986	0.990	0%
RA-8	strict	Logit	0.869	0.086	2	[1, 2]	0.980	0.997	0%
RA-8	strict	RF	0.900	0.247	2	[2, 2]	1.00	1.00	18%
RA-8	strict	GBDT	0.899	0.231	2	[1.75, 2]	1.00	1.00	17%
RA-8	strict	XGB	0.904	0.251	2	[2, 2]	1.00	1.00	19%
VA-7b	base	Logit	0.862	0.049	1	[1, 2]	0.983	0.897	0%
VA-7b	base	RF	0.871	0.042	1	[1, 2]	0.987	0.827	0%
VA-7b	base	GBDT	0.869	0.046	1	[1, 2]	0.997	0.827	0%
VA-7b	base	XGB	0.869	0.044	1	[1, 2]	0.880	0.827	0%

First, robustness is bounded by a low effective cost floor. On the full feature set, all architectures exhibit median MEC = 2 with narrow interquartile ranges and small FRI values. Although single-feature transitions of cost 1 exist, the empirical mass at cost 1 falls below one half, so the median binds at the next effective threshold. The convergence of linear, bagging, and boosting models to the same median MEC confirms the action-set-limited invariance of Corollary 3.1.

Second, successful evasion concentrates sharply on a small feature subset. For the full feature set under the base schedule, RCI_3 exceeds 0.78 across models and reaches 0.96 for logistic regression. Evasion traces collapse onto low-cost, high-influence features rather than dispersing across the representation. The 95% bootstrap confidence intervals (200 resamples) confirm that these patterns are statistically stable: median MEC = [2, 2] for all models, RCI_3 within ±0.03, and FRI within ±0.01 (Table 8).

Table 8. 95% bootstrap confidence intervals (200 resamples, Full/base, 300-instance intersection).

Model	Median MEC [95% CI]	FRI [95% CI]	RCI_3 [95% CI]
Logistic Regression	2 [2, 2]	0.076 [0.068, 0.084]	0.96 [0.94, 0.97]
Random Forest	2 [2, 2]	0.092 [0.082, 0.101]	0.84 [0.80, 0.87]
GBDT	2 [2, 2]	0.076 [0.069, 0.083]	0.89 [0.86, 0.92]
XGBoost	2 [2, 3]	0.092 [0.083, 0.101]	0.82 [0.78, 0.85]

The RA-8 configuration makes the cost-floor mechanism explicit. Despite emphasizing infrastructure features, RA-8 retains SSLfinal_State, a low-cost surface coordinate. Median MEC remains 2, while concentration becomes nearly degenerate ($RCI_3 \approx 1$, $\text{FirstTop1} \approx 1$). The surface-only VA-7b set exhibits the lowest robustness (median MEC = 1, FRI < 0.05).

Cost schedules matter only when they eliminate dominant cheap paths. This occurs in RA-8 under the strict schedule: ensemble models exhibit 17% - 19% infeasible mass, raising FRI to 0.23 - 0.25, while median MEC among evadable instances remains 2. The gain arises from blocked feasibility rather than uniformly higher evasion costs. Logistic regression remains fully evadable in RA-8/strict, indicating alternative low-cost paths in the linear boundary.

Figure 2 displays evasion survival curves. VA-7b collapses immediately ($S(B) < 0.05$ by $B = 2$). Full and RA-8/base decay to near zero by $B = 4$. RA-8/strict exhibits a persistent plateau near 0.18, matching the infeasible mass in Table 7. The strict schedule generates a structural tail rather than shifting the central cost distribution.

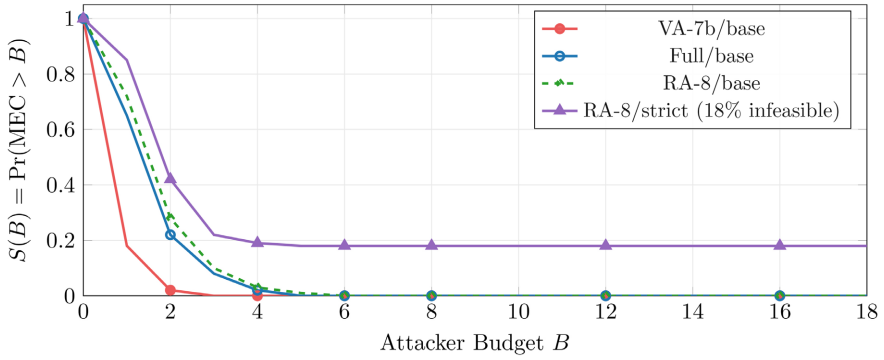

Figure 2. Evasion survival curves. RA-8/strict exhibits a persistent plateau corresponding to instances whose dominant low-cost path is blocked. Shaded bands (omitted for clarity) are narrow: 95% bootstrap intervals for $S(2)$ span ± 0.04 across configurations.

Figure 3 displays first-edit concentration across feature sets. RA-8 concentrates nearly all optimal traces on a single initial edit (SSLfinal_State), while Full distributes first edits across a small but nontrivial subset. Even in the latter case, concentration remains substantial.

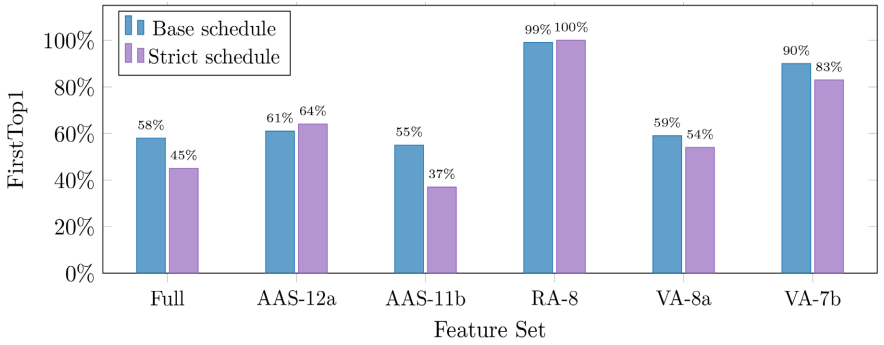

Figure 3. First-edit concentration by feature set and schedule. RA-8 exhibits near-total concentration on SSLfinal_State.

Stratification by the bottleneck feature confirms the blocked-path mechanism in RA-8/strict. When SSLfinal_State begins at −1 or 0, low-cost upgrades remain available and evasion succeeds with median MEC between 1 and 2. When SSLfinal_State is already +1, the dominant path is blocked and a persistent infeasible tail appears (**Figure 4**).

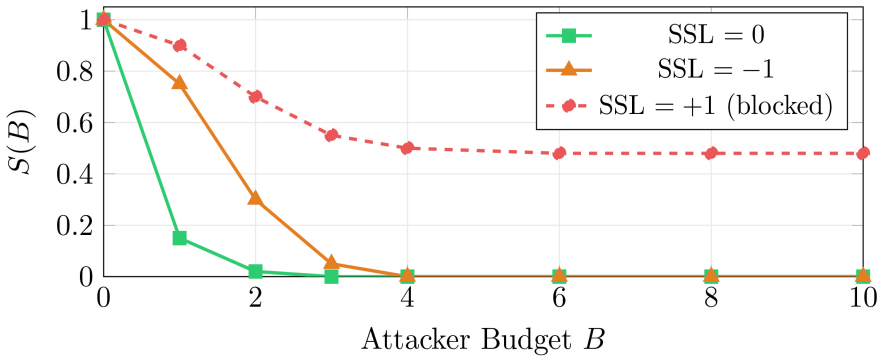

Figure 4. RA-8/strict survival stratified by SSLfinal_State initial value. A persistent infeasible tail appears when the bottleneck feature is already at +1.

Figure 5 compares i.i.d. accuracy with median MEC. All architectures align along a horizontal band at MEC = 2, confirming that higher accuracy does not yield higher median robustness when low-cost transitions remain available.

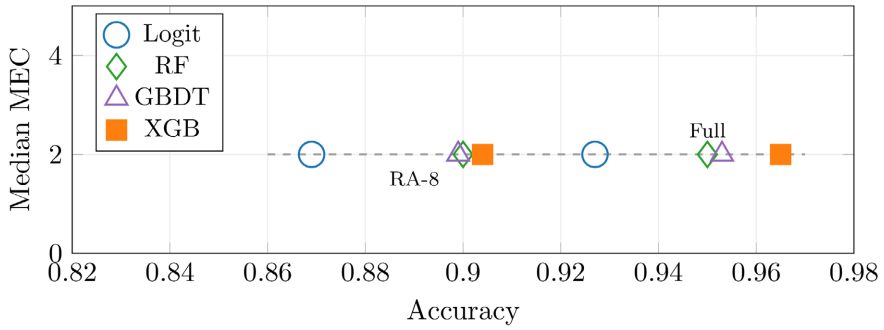

Figure 5. Accuracy versus median MEC. All architectures converge to the effective cost floor, consistent with Corollary 3.1.

Table 9 compares exact MEC with greedy approximations under query budgets of 50, 100, and 500. Query limitations modestly increase survival, particularly at 50 queries, but the deviation from exact MEC is small and narrows rapidly. In Full/base, the maximum gap at $B=2$ is 0.08. Exact MEC thus provides a meaningful upper bound on attacker capability: query-limited adversaries are less efficient but face the same structural cost-floor constraints.

Table 9. Evasion survival $S(B)$ under query-limited greedy search versus exact MEC. The greedy attacker is described in Section 2.5.

Config	Model	$S(B=2)$				$S(B=4)$			
		Q50	Q100	Q500	Exact	Q50	Q100	Q500	Exact
Full/base	Logit	0.28	0.24	0.23	0.22	0.05	0.03	0.02	0.02
Full/base	GBDT	0.30	0.26	0.23	0.22	0.08	0.05	0.03	0.02
Full/base	XGB	0.34	0.30	0.29	0.28	0.07	0.05	0.03	0.03
RA-8/str	GBDT	0.48	0.44	0.43	0.42	0.24	0.21	0.19	0.19
RA-8/str	XGB	0.50	0.46	0.44	0.42	0.26	0.22	0.20	0.19

5. Cost Sensitivity Analysis

We evaluate robustness under three classes of cost perturbation to assess whether conclusions depend on the specific magnitudes chosen. First, surface costs are scaled by $\lambda_{\text{surf}} \in \{1,2,3,4\}$. Second, semi-domain and infrastructure costs are scaled independently by $\lambda \in \{0.5,1,2\}$. Third, SSLfinal_State is reclassified from surface to semi-domain, and a rank-preserving perturbation multiplies each cost by an independent factor $U \sim \text{Uniform}[0.8,1.2]$ over 50 draws.

Table 10 reports results under surface scaling. The median MEC shifts proportionally, confirming linear cost-floor behavior: doubling surface costs increases the median from 2 to 4, while preserving the identity and ordering of the three most-edited features. Even at $\lambda_{\text{surf}}=4$, concentration remains high ($\text{RCI}_3 \geq 0.80$).

Table 10. Median MEC and concentration under surface cost scaling (Full/base, GBDT).

λ_{surf}	Median MEC	RCI_3	Top-3 features
1 (base)	2	0.89	URL_of_Anchor, SSLfinal_State, SFH
2	4	0.87	URL_of_Anchor, SSLfinal_State, SFH
3	6	0.84	URL_of_Anchor, SSLfinal_State, SFH
4	8	0.82	URL_of_Anchor, SSLfinal_State, SFH

Table 11 reports extended perturbations. Scaling semi-domain or infrastructure costs does not alter median MEC because surface transitions remain dominant. Reclassifying SSLfinal_State increases median MEC to 3 in RA-8 (where it is the bottleneck) but leaves Full unchanged due to alternative surface paths. Un-

der rank-preserving noise, mean $RCI_3 = 0.88 \pm 0.02$, indicating stability to moderate cost uncertainty.

Table 11. Extended cost sensitivity (Full/base, GBDT unless noted).

Perturbation	Median MEC	RCI_3	Notes
$\lambda_{\text{semi}} = 0.5$	2	0.89	Surface paths remain cheapest
$\lambda_{\text{semi}} = 2$	2	0.89	Semi-domain rarely on optimal path
$\lambda_{\text{infra}} = 0.5$	2	0.88	Surface transitions dominate
$\lambda_{\text{infra}} = 2$	2	0.90	Infrastructure edits avoided
SSL $\rightarrow$ semi-domain (Full)	2	0.84	Alternative surface paths used
SSL $\rightarrow$ semi-domain (RA-8)	3	0.95	Bottleneck cost increases
Random $U[0.8,1.2]$ (×50)	2 ± 0	0.88 ± 0.02	Stable under moderate noise

The main results use a fixed decision threshold of $\tau = 0.5$ for all models. To verify that robustness conclusions are not conflated with cross-model calibration differences, we re-evaluate median MEC and RCI_3 under thresholds $\tau \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$, redefining $\mathcal{P}_0(\tau) = \{x \in \mathcal{P}_{\text{test}} : \hat{p}(x) < \tau, y(x) = -1\}$ at each threshold. We additionally compare models at a matched operating point where thresholds are adjusted per model to achieve phishing TPR ≈ 0.95.

Table 12 reports results for Full/base across all four models. Median MEC is 2 for $\tau \in \{0.4, 0.5, 0.6, 0.7\}$ across all models. At $\tau = 0.3$, conditioning on high-confidence phishing detections shifts the median to 3 for all models, consistent with the cost floor: these instances tend to be farther from the decision boundary, and a single cheap transition is insufficient for a larger fraction of them. At the matched TPR ≈ 0.95 operating point, median MEC remains 2 for all models. RCI_3 is stable across thresholds (range 0.80 - 0.97). These results confirm that the robustness convergence is not an artifact of threshold choice.

Table 12. Median MEC under varying classification thresholds (Full/base). Matched TPR row adjusts each model's threshold to achieve phishing TPR ≈ 0.95.

τ	LR	RF	GBDT	XGB
0.30	3	3	3	3
0.40	2	2	2	2
0.50	2	2	2	2
0.60	2	2	2	2
0.70	2	2	2	2
Matched TPR ≈ 0.95	2	2	2	2

These experiments reveal two conditions. In the cost-floor condition, MEC quantiles scale with the cheapest admissible transition and architecture invariance holds. In the path-removal condition, prohibiting dominant transitions induces infeasible mass without shifting the cost distribution among evadable instances. The strict schedule operates in the latter condition for RA-8, producing robustness gains through blocked feasibility.

6. Discussion

Across all tested feature sets, cost schedules, and model families, robustness is governed by the cheapest admissible manipulation that remains available. The median MEC follows the effective cost floor across all configurations, rendering Proposition 3.1 empirically tight. When a low-cost transition suffices for a non-trivial fraction of correctly detected instances, architectural complexity does not move the median. This action-set-limited invariance means that linear models, bagging ensembles, and boosting methods converge to the same robustness ceiling.

The implication is a shift in defensive emphasis from model selection to representation design and attacker economics. A feature may be highly predictive under i.i.d. evaluation yet operationally brittle if it is inexpensive to edit. The RA-8 configuration illustrates this: although it prioritizes infrastructure-leaning signals, retaining a single low-cost coordinate (SSLfinal_State) creates a bottleneck through which nearly all optimal evasions pass. Cost schedules improve robustness only when they eliminate dominant cheap paths, producing infeasible mass rather than uniformly higher evasion costs. Meaningful robustness gains require removing or economically disabling low-cost transitions and anchoring detection on signals whose manipulation costs exceed realistic attacker budgets, even at the expense of i.i.d. accuracy.

The sanitization-only threat model constitutes a lower bound. Relaxing monotonicity, by allowing anti-feature injection (adding benign-looking HTML artifacts to boost legitimacy scores) or extractor-level manipulation (crafting raw pages to flip computed features without semantic change [13] [17]), enlarges the feasible action set. The cost floor can only decrease or remain unchanged, since every monotone path remains available. Concentration may increase if newly available non-monotone transitions converge on a small set of vulnerable coordinates, or shift to different features if injected anti-features provide cheaper evasion than indicator removal. The infeasible mass observed under the strict schedule would likely shrink or vanish, as non-monotone paths can bypass blocked transitions. Formalizing these effects requires specifying non-monotone cost structures and is left to future work, but the qualitative conclusion is reinforced: the monotone analysis provides a conservative bound on attacker capability.

Limitations and external validity. The UCI Phishing Websites benchmark [19] is a standard reference point but is dated: it uses a fixed, hand-engineered vocabulary that omits modern signals, including certificate-transparency logs, visual

similarity [9], JavaScript behavioral fingerprints [20], and infrastructure patterns in contemporary kit-based campaigns [10] [11]. Quantitative transfer to modern settings requires re-validation on current datasets, mapping contemporary features to a cost schedule via the time-to-effect principle, and verifying whether low-cost transitions continue to dominate MEC.

Several structural conclusions are nevertheless important to the dataset choice. The surface-versus-infrastructure cost asymmetry is an economic regularity: presentation-layer signals are cheaper to manipulate than infrastructure-coupled signals, regardless of the specific feature dictionary [8] [18]. Proposition 3.1 is a property of the action set and cost model, not the dataset; it applies whenever a nontrivial fraction of instances admit single-transition evasion at minimal cost. Concentration follows from heterogeneous costs interacting with feature influence, a generic property in discrete domains with uneven manipulation friction.

Our MEC computation assumes unconstrained black-box label access. **Table 9** shows that reasonable query budgets reduce attacker efficiency without altering feasibility patterns, but production systems with aggressive rate-limiting can increase observed survival. The cost schedule represents dimensionless operational friction calibrated by the time-to-effect principle rather than direct monetary expenditure; translating to market-level budgets remains an open empirical problem.

7. Conclusion

Near-perfect held-out accuracy does not imply deployment security when evasion is cheap. Across all model architectures, feature configurations, and cost schedules studied here, robustness is determined by the minimum manipulation cost available to the attacker, not by classifier complexity. Feature economics dominate adversarial robustness under cost-constrained post-deployment manipulation: the central obstacle to robustness is the continued availability of low-cost transitions, and architecture choice cannot compensate for this structural exposure. Effective defense requires either removing cheap-to-edit features from the detection vocabulary or raising their manipulation cost through verifiable infrastructure anchoring—even at some sacrifice in i.i.d. accuracy.

Acknowledgements

This work was supported by the U.S. Department of Education under grant number P382G240006. The authors thank the anonymous reviewers for their helpful suggestions and careful reading of the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Basit, A., Zafar, M., Liu, X., Javed, A.R., Jalil, Z. and Kifayat, K. (2020) A Comprehensive Survey of AI-Enabled Phishing Attacks Detection Techniques. *Telecommunica-*

tion Systems, **76**, 139-154. https://doi.org/10.1007/s11235-020-00733-2

[2] Mohammad, R.M., Thabtah, F. and McCluskey, L. (2014) Predicting Phishing Websites Based on Self-Structuring Neural Network. *Neural Computing and Applications*, **25**, 443-458. https://doi.org/10.1007/s00521-013-1490-z

[3] Sahingoz, O.K., Buber, E., Demir, O. and Diri, B. (2019) Machine Learning Based Phishing Detection from URLs. *Expert Systems with Applications*, **117**, 345-357. https://doi.org/10.1016/j.eswa.2018.09.029

[4] Do, N.Q., Selamat, A., Krejcar, O., Herrera-Viedma, E. and Fujita, H. (2022) Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions. *IEEE Access*, **10**, 36429-36463. https://doi.org/10.1109/access.2022.3151903

[5] Biggio, B. and Roli, F. (2018) Wild Patterns: Ten Years after the Rise of Adversarial Machine Learning. *Proceedings of the* 2018 *ACM SIGSAC Conference on Computer and Communications Security*, Toronto, 15-19 October 2018, 2154-2156. https://doi.org/10.1145/3243734.3264418

[6] Apruzzese, G., Anderson, H.S., Dambra, S., Freeman, D., Pierazzi, F. and Roundy, K. (2023) "Real Attackers Don't Compute Gradients": Bridging the Gap between Adversarial ML Research and Practice. 2023 *IEEE Conference on Secure and Trustworthy Machine Learning* (*SaTML*), Raleigh, 8-10 February 2023, 339-364. https://doi.org/10.1109/satml54575.2023.00031

[7] Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., *et al.* (2013) Evasion Attacks against Machine Learning at Test Time. In: *Lecture Notes in Computer Science*, Springer, 387-402. https://doi.org/10.1007/978-3-642-40994-3_25

[8] Khonji, M., Iraqi, Y. and Jones, A. (2013) Phishing Detection: A Literature Survey. *IEEE Communications Surveys & Tutorials*, **15**, 2091-2121. https://doi.org/10.1109/surv.2013.032213.00009

[9] Lin, Y., Liu, R., Divakaran, D.M., Ng, J.Y., *et al.* (2021) Phishpedia: A Hybrid Deep Learning Based Approach to Visually Identify Phishing Webpages. *Proceedings of the* 30*th USENIX Security Symposium*, Vancouver, 11-13 August 2021, 3793-3810.

[10] Oest, A., Safaei, Y., Doupé, A., Ahn, G.J., *et al.* (2020) Sunrise to Sunset: Analyzing the End-to-End Life Cycle and Effectiveness of Phishing Attacks at Scale. 2020 29*th USENIX Security Symposium*, Boston, 12-14 August 2020, 361-377.

[11] Bijmans, P.H., Booij, T.M. and van Eeten, M. (2021) Catching Phishers by Their Bait: Investigating the Dutch Phishing Landscape through Phishing Kit Analysis. *Proceedings of the* 30*th USENIX Security Symposium*, Vancouver, 11-13 August 2021, 3757-3774.

[12] Ruan, W.J., Yi, X.P. and Huang, X.W. (2021) Adversarial Robustness of Deep Learning: Theory, Algorithms, and Applications. *Proceedings of the* 30*th ACM International Conference on Information & Knowledge Management*, Queensland, 1-5 November 2021, 4866-4869.

[13] Pierazzi, F., Pendlebury, F., Cortellazzi, J. and Cavallaro, L. (2020). Intriguing Properties of Adversarial ML Attacks in the Problem Space. 2020 *IEEE Symposium on Security and Privacy* (*SP*), San Francisco, 18-21 May 2020, 1332-1349. https://doi.org/10.1109/sp40000.2020.00073

[14] Carlini, N. and Wagner, D. (2017) Towards Evaluating the Robustness of Neural Networks. 2017 *IEEE Symposium on Security and Privacy* (*SP*), San Jose, 22-26 May 2017, 39-57. https://doi.org/10.1109/sp.2017.49

[15] Goodfellow, I.J., Shlens, J., and Szegedy, C. (2015) Explaining and Harnessing Adversarial Examples. 2015 3*rd International Conference on Learning Representations*, San

Diego, 7-9 May 2015, 11 p.

[16] Corona, I., Biggio, B., Contini, M., Piras, L., Corda, R., Mereu, M., *et al.* (2017) DeltaPhish: Detecting Phishing Webpages in Compromised Websites. In: *Lecture Notes in Computer Science*, Springer, 370-388. https://doi.org/10.1007/978-3-319-66402-6_22

[17] Xu, W., Qi, Y. and Evans, D. (2016) Automatically Evading Classifiers: A Case Study on PDF Malware Classifiers. *Proceedings* 2016 *Network and Distributed System Security Symposium*, San Diego, 21-24 February 2016, 1-15. https://doi.org/10.14722/ndss.2016.23115

[18] Das, A., Baki, S., El Aassal, A., Verma, R. and Dunbar, A. (2020) SoK: A Comprehensive Reexamination of Phishing Research from the Security Perspective. *IEEE Communications Surveys & Tutorials*, **22**, 671-708. https://doi.org/10.1109/comst.2019.2957750

[19] Mohammad, R.M., Thabtah, F. and McCluskey, L. (2015) Phishing Websites Data Set. UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/phishing+websites

[20] Rao, R.S., Vaishnavi, T. and Pais, A.R. (2020) CatchPhish: Detection of Phishing Websites by Inspecting URLs. *Journal of Ambient Intelligence and Humanized Computing*, **11**, 813-825. https://doi.org/10.1007/s12652-019-01311-4

Appendix: Feature-to-Cost-Group Mapping

Table A1 maps all 30 UCI Phishing Websites features to their cost group assignment under the base schedule, along with the transition costs and a one-line time-to-effect rationale. The assignment is governed by the time-to-effect principle: surface features require at most one day to modify under campaign-operational conditions; semi-domain features require days to weeks; infrastructure features require weeks to months or are effectively infeasible within a typical campaign window. This table is provided to make the cost calibration fully reproducible.

Table A1. Complete feature-to-cost-group mapping for the UCI Phishing Websites dataset (base schedule). Transition costs follow Table 1. The strict schedule sets infrastructure $-1 \to 1$ and $0 \to 1$ to ∞.

Feature	Group	Costs ($-1 \to 0$, $-1 \to 1$, $0 \to 1$)	Time-to-effect rationale
having_IP_Address	Surface	1, 2, 1	Replace IP with registered domain in minutes
URL_Length	Surface	1, 2, 1	Shorten URL string in minutes
Shortining_Service	Surface	1, 2, 1	Remove or replace URL shortener within hours
having_At_Sign	Surface	1, 2, 1	Drop @ from URL string immediately
double_slash_redirecting	Surface	1, 2, 1	Correct redirect path within hours
Prefix_Suffix	Surface	1, 2, 1	Remove hyphen from domain string within hours
having_Sub_Domain	Surface	1, 2, 1	Adjust subdomain structure within hours
SSLfinal_State	Surface	1, 2, 1	Deploy free DV certificate (e.g., Let's Encrypt) in under a day
Favicon	Surface	1, 2, 1	Replace favicon file on server within hours
port	Surface	1, 2, 1	Configure server to use standard port within hours
HTTPS_token	Surface	1, 2, 1	Remove "https" string from URL immediately
Request_URL	Surface	1, 2, 1	Adjust resource request paths within hours
URL_of_Anchor	Surface	1, 2, 1	Modify anchor href attributes in HTML within hours
Links_in_tags	Surface	1, 2, 1	Update embedded link tags in HTML within hours
SFH	Surface	1, 2, 1	Modify HTML form action to legitimate endpoint within hours
Submitting_to_email	Surface	1, 2, 1	Remove mailto: form action within hours
Abnormal_URL	Surface	1, 2, 1	Normalize URL structure relative to hostname within hours
Redirect	Surface	1, 2, 1	Adjust server-side redirect count within hours
on_mouseover	Surface	1, 2, 1	Remove or rewrite JavaScript mouseover handler within hours
RightClick	Surface	1, 2, 1	Remove JavaScript right-click disabler within hours
popUpWidnow	Surface	1, 2, 1	Remove pop-up window JavaScript within hours
Iframe	Surface	1, 2, 1	Remove <iframe> elements from HTML within hours
Domain_registeration_length	Semi-domain	3, 6, 3	Extend domain registration; requires payment and propagation over days
Google_Index	Semi-domain	3, 6, 3	Submit sitemap; indexing typically takes days to a week

Continued

Links_pointing_to_page	Semi-domain	3, 6, 3	Accumulate inbound links; requires days of coordination
Statistical_report	Semi-domain	3, 6, 3	Clear entry from phishing databases; dispute takes days
age_of_domain	Infrastructure	4, 8, 4	Domain age cannot be accelerated; accrues over months
DNSRecord	Infrastructure	4, 8, 4	DNS reputation requires weeks to establish
web_traffic	Infrastructure	4, 8, 4	Organic traffic accumulation requires weeks to months
Page_Rank	Infrastructure	4, 8, 4	PageRank accrues over months via link building