applied sciences

MDPI

Article

An Auditable Review-Referral Framework for Public PE Malware Scores at Software Supply-Chain Gates

Changho Kweon, Yeonseung Ryu * and Sambeom Shin

Department of Defense Industrial Security, Myongji University, Yongin 17058, Republic of Korea
* Correspondence: ysryu@mju.ac.kr

Featured Application

The framework uses existing PE scores to select files for additional review at acquisition, build, release, and update gates. Results on public benchmarks do not establish equal detection performance on live supplier files, and a referral is not itself authorization to block, reject, or quarantine a file or to sanction a supplier; those actions require separate evidence and organizational approval.

Abstract

This study presents an auditable decision-governance procedure for converting existing malware-suspicion scores into additional-review referrals at software supply-chain gates under limits on false-positive rate (FPR), review capacity, and minimum recall. Instead of a new detector or selection rule, it applies constrained cost minimization on validation data only, freezes thresholds before testing, records versioned model, score, data, and approval links, and preserves every infeasible or failed result. EMBER, SOREL-20M, and BODMAS are generic benchmarks, not supplier-intake records. Of 1152 registered Policy Cβ profiles, 384 were infeasible because capacity and minimum recall conflicted arithmetically and 48 because no observed threshold met all limits. At minimum recall 0.90, all limits were retained on test in 28 of 112 validation-feasible EMBER profiles. For SOREL, retention was 0 of 128 on the prespecified official test-result file and 112 of 128 in a rescoring sensitivity analysis, which cannot validate the official result. Without verified file-level alignment, its cause is unresolved. A binding recall floor selected thresholds that then failed it on EMBER test data. Sensitivity analyses show, for the observed validation scores, when thresholds remain feasible over stated input ranges; an exploratory evidence gate adds pointwise margins, not guarantees. All results require prospective organizational validation.

Keywords: cybersecurity supply chain risk management (C-SCRM); decision governance; constrained threshold selection; selective classification; false-positive rate; review capacity; minimum recall; score provenance; EMBER; SOREL-20M

Check for updates

Academic Editor: Karim Elish

Received: 30 September 2026
Revised: 19 October 2026
Accepted: 22 October 2026
Published: 28 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

In practical terms, a detector assigns each file a suspicion score. A low threshold sends many files to analysts and may overwhelm the review team, whereas a high threshold reduces workload but may miss malicious files. This study selects a threshold that balances security performance against the number of reviews the organization can complete. The selected threshold is frozen before testing, and the procedure records a failure whenever the approved conditions cannot be satisfied.

Software supply-chain risk spans acquisition through operations, so C-SCRM combines lifecycle controls, secure development, and component transparency [1–4]. Static PE detectors provide non-executing suspicion scores, while EMBER and SOREL-20M enable reproducible evaluation of the post-score decision layer [5–7].

Conventional classification metrics alone cannot determine which artifacts should receive additional review at a supply-chain gate. That decision also requires explicit error costs, an assumed operational malicious-artifact rate, and an approved operating point [8–11].

For example, with $\pi = 1\%$, recall = 0.95, and FPR = 0.05, a stream of 10,000 artifacts produces approximately 590 additional-review cases: 95 malicious and 495 benign. A team that can complete only 200 reviews cannot keep pace. Thus, strong conventional detection performance may still yield an operationally unusable threshold.

This paper applies an additional-review threshold τR to a fixed PE malware score. Files with scores at or above τR are selected for further review; lower-scoring files remain candidates for approval subject to hashes, signatures, SBOMs, provenance, and other controls. Selection for additional review is neither proof of malware nor authorization to quarantine or reject a file, block a release, or sanction a supplier.

The contribution is a reproducible decision-governance procedure for converting existing malware scores into review referrals at software supply-chain gates; its feasibility and performance must be assessed for each target environment, and it is not offered as an already validated, generally deployable solution. The underlying selection principle—minimizing expected cost subject to constraints—is established and is not claimed as new. At the methodological level, the study (i) formulates referral as threshold selection on fixed scores under three simultaneous operational limits—an FPR ceiling, a review-capacity ceiling tied to the assumed malicious-artifact rate, and a minimum recall—and, using elementary monotonicity properties of fixed-score thresholding, characterizes when the jointly constrained set is empty, when the recall floor changes the selected operating point, and when a threshold remains feasible over a stated range of uncertain inputs, conditional on the observed validation scores (Section 3.2.5); and (ii) separates arithmetic (structural) infeasibility from score-dependent infeasibility, and both from the failure of an already frozen threshold on held-out data. At the engineering and governance level, it integrates validation-only selection with pre-test freezing, explicit preservation of infeasible and failed outcomes, versioned provenance across models, scores, data, preprocessing, code, thresholds, and approval conditions, verification of the score-generation path, an exploratory evidence gate based on pointwise validation confidence bounds, and workload evidence. Section 2.6 compares these elements with the closest existing approaches.

For each reported score, the provenance record lists the model or checkpoint, input features, preprocessing and inference settings, scoring program and version, score column and direction, and generation evidence, and states whether the generating path was verified (Section 4.3.1). The path was verified for every score generated in this study. For the official SOREL test-result file, the SHA-256 hash fixes the file's identity, but its producing checkpoint and settings remain unverified, because a file hash does not establish how a file was generated (Sections 4.3.1 and 5.4). This distinction proved decisive for SOREL: the prespecified official test-result file and a same-pipeline rescoring of the official test-feature archive with the released checkpoint lead to different frozen-test conclusions (Section 5.4).

The present study extends the ideas of two related Korean-language studies [12,13]. K.C. and S.S. coauthored [12], and K.C., S.S., and R.Y. coauthored [13]. However, no private samples, trained models, numerical results, tables, figures, or experimental outputs from those studies are reused or reanalyzed here. Study [12] linked malware-detection performance to NIST-aligned security controls, whereas [13] introduced Policy A/B/C on

2000 balanced private PE files. The present study applies malicious-artifact-rate, review-capacity, FPR, and minimum-recall constraints to fixed public EMBER and SOREL scores. It also freezes thresholds before testing, compares policies on the same test files, and supplies provenance and queue evidence for C-SCRM authorization. These benchmarks evaluate the post-score review policy, not field detection effectiveness in a supplier environment.

1.1. Plain-Language Guide to Core Terms

An additional-review rule uses an existing suspicion score to decide which files need further human or sandbox analysis; referral does not itself mean that a file is malicious. Once the threshold is fixed, each labeled test file falls into one of four cases. A malicious file sent for review is a true positive (TP), whereas a malicious file not sent for review is a false negative (FN), meaning that the procedure missed a known malicious file. A benign file sent for review is a false positive (FP), meaning that it created unnecessary review work, whereas a benign file not sent for review is a true negative (TN). The benchmark label supplies the known class, and the fixed threshold supplies the referral decision. FP and FN are therefore counted by comparing those two items; the score alone cannot reveal the true class. Recall is $TP/(TP + FN)$, and FPR is $FP/(FP + TN)$. In live operation, the true class is not known immediately, so analysts, sandbox evidence, or another independent adjudication process must confirm outcomes before operational FP and FN counts can be calculated. Review capacity is the number of referred files that analysts can complete within a stated period. Review backlog is the number left unprocessed when referrals exceed that capacity.

EMBER is a public benchmark that provides labels and static features extracted from Windows PE files without executing them. SOREL-20M is a larger public PE-malware benchmark containing labels, features, file hashes, and baseline-model outputs. The EMBER and SOREL experiments evaluate fixed public-model scores without executing PE files. The BODMAS extension fits three established reference models on its training partition and then fixes their scores for policy evaluation (Section Additional Check on the Time-Ordered BODMAS Dataset). No new detector architecture is proposed. The malicious-artifact rate π is the expected malicious share of all files entering a gate. Thus, $\pi = 0.01$ assumes about 100 malicious files among 10,000 arrivals, and this rate changes both expected error cost and review workload.

The Policy Cβ (C-beta) implementation searches candidate thresholds on validation data and selects a threshold only if it simultaneously satisfies the FPR ceiling, the maximum additional-review rate, and the minimum-recall requirement. If no threshold satisfies all three constraints, the procedure does not relax a constraint or substitute another threshold; it records the operating profile as infeasible. The month-forward split audit is a reproducibility check of the temporal evaluation, not a model-training step. It verifies that EMBER rows from November are used for threshold selection and rows from December for evaluation, confirms month and label ordering and class counts, and ensures that the two sets do not overlap.

A reject option allows a classifier to withhold a final decision on a difficult case and route it to another review process. Learning to defer extends this approach by learning when a human or another expert should take over. A surrogate estimator replaces the difficult discrete defer-or-decide objective with a proxy loss that can be optimized computationally. Mozannar and Sontag [14] proposed a consistent surrogate for learning expert deferral. This study, by contrast, does not learn expert behavior; it applies a threshold approved on validation data to fixed scores produced by existing public models.

1.2. Research Questions and Experimental Structure

To make the study's novelty and experimental structure explicit, we address three research questions.

RQ1. Under the registered operating conditions, when does a jointly constrained Policy Cβ threshold exist, which infeasible profiles are explained by the arithmetic bound $\pi\beta_{\min} \leq r_{\max}$ rather than by the score distribution, and how does feasibility respond to uncertainty in π, the loss weights, α, $r_{\max}$, and $\beta_{\min}$?

RQ2. After validation-only selection and freezing, to what extent do held-out tests retain the approved FPR ceiling α, capacity ceiling $r_{\max}$, and minimum recall $\beta_{\min}$; when does the recall floor change Policy C's decision; and does an exploratory confidence-based evidence gate change selection or retention?

RQ3. How stable are validation feasibility and frozen-test retention across EMBER, SOREL (by test-score source), and three BODMAS detectors, and what review workload follows under the representative condition?

RQ1 is addressed in Sections 3.2.5, 5.8, 5.9 and 5.17; RQ2 in Sections 5.3, 5.10–5.13 and 5.15; and RQ3 in Sections 5.4, 5.7, 5.12, 5.16 and 5.17.

2. Related Work and Study Positioning

2.1. Static Malware Detection and Evaluation Bias

Static malware detection has used raw bytes, byte sequences, executable structure, imports, metadata, and image-like representations [15–18]. Temporal and spatial bias can inflate malware-classification estimates, while adversarial manipulation can change outputs without preserving the assumptions of a conventional feature-space test [19–23]. These findings motivate strict separation of threshold selection from evaluation and caution against treating a score threshold as evidence of malicious intent.

2.2. C-SCRM, Provenance, and Complementary Evidence

Package-manager and secure-update research emphasizes delegated trust, signatures, and compromise recovery [24,25]. Mercury addresses rollback protection [26], SLSA provides artifact-integrity expectations [27], and work on reproducible builds, supply-chain attack taxonomies, incident collections, and SBOMs shows that no single artifact signal covers the full threat surface [28–31]. Accordingly, the proposed score threshold is one evidence-producing control within a larger gate, not a replacement for hashes, signatures, SBOM review, provenance, or dynamic analysis.

2.3. Probability Calibration and the Malicious-Artifact-Rate Problem

The Brier score measures squared deviation between probability-like scores and binary outcomes [32]. Work on probability estimation and calibration shows that classifiers with similar ranking can differ substantially in probability quality [33,34]. A score near 0.8 is not necessarily an 80% operational probability of malware. The primary policies in this study therefore select τR empirically on validation data. The theoretical Bayes rule is an assumption-dependent comparison, not a universally valid deployment threshold.

2.4. Selective Classification, Reject Options, and Learning to Defer

A reject option allows a classifier to withhold a decision on a difficult case and send it to another process, whereas learning to defer trains a system to decide when a human or another expert should take over. Madras et al. [35] jointly modeled the external decision maker's accuracy and bias. Mozannar and Sontag [14] proposed a consistent surrogate

estimator, a trainable proxy loss for a deferral decision that is difficult to optimize directly. Alves et al. [36] added expert-specific error costs and workload limits.

Unlike learning-to-defer methods that train a predictor or model expert behavior, this framework operates after scoring and adopts established decision principles rather than replacing them. Section 2.6 states which of its elements are methodological and which are engineering or governance contributions.

2.5. International Comparison and Differences from Related Korean-Language Studies

Table 1 shows how the present study differs from prior work. Study [12] linked malware-detection performance to NIST-aligned security controls, whereas ref. [13] evaluated Policy A/B/C on 2000 private PE files with four models, a 10:1 false-negative-to-false-positive cost ratio, and an FPR ceiling of 0.10. The closest workload-aware international study, Alves et al. [36], learns which cases to defer to multiple experts. The present study instead evaluates fixed scores from established detectors on public PE benchmarks. EMBER/SOREL use public-model outputs without retraining; BODMAS supplies a time-ordered extension with fitted reference models. It selects thresholds only on validation data, freezes them before testing, and records cases in which no threshold satisfies the approved constraints. Paired policy comparisons, provenance, execution records, and queue evidence are then integrated as support for C-SCRM policy authorization.

Table 1. Comparison with representative international methods and related Korean-language studies.

Study	Decision Setting	Operational Objective or Constraints	Evidence Protocol	Difference from the Present Study
Axelsson [8]	Intrusion-detection alarms in rare-event conditions	Malicious-artifact-rate effect and false-alarm burden	Analytical argument	Does not specify validation-only threshold selection, review capacity, or audit artifacts
Elkan [9]	Cost-sensitive classification	Class-dependent misclassification cost	Theoretical framework	Does not jointly impose FPR, capacity, and minimum-recall constraints
Geifman and El-Yaniv [37]	Post-hoc selective classification	Risk-coverage control through abstention	Trained neural network plus held-out selection	Does not address C-SCRM malicious-artifact rates, queue load, or score provenance
Madras et al. [35]	Learning to defer to a decision maker	Prediction accuracy and fairness with decision-maker behavior	Jointly learned predictor/deferral model	Requires learning expert behavior rather than governing a fixed public score
Mozannar and Sontag [14]	Classifier-rejector deferral to an expert	Consistent surrogate for system error	Learns from labels and expert decisions	Does not provide a fixed-score, three-constraint, frozen-test audit procedure
Alves et al. [36]	Deferral to multiple experts	Heterogeneous error costs and workload constraints	Learned multi-expert deferral	Closest workload-aware comparator, but not a post-score C-SCRM threshold-governance study
Tong et al. [38]	Neyman–Pearson classification	Minimize one error type subject to a high-probability bound on the other	Order-statistic threshold on held-out data	Controls one error at a time; no prevalence-dependent workload ceiling, recall floor, or infeasibility record
Angelopoulos et al. [39,40]	Risk-controlled threshold calibration	Finite-sample control of a user-specified risk	Calibration set, exchangeability, multiple testing or conformal bounds	Assumes exchangeable calibration and deployment data; does not address score-path failure or workload governance

Table 1. *Cont.*

Study	Decision Setting	Operational Objective or Constraints	Evidence Protocol	Difference from the Present Study
Kweon et al. [12]	NIST-aligned malware-detector evaluation	Detector performance and security-control linkage	Private balanced PE data; CNN training; five-fold evaluation	No public fixed-score evaluation, recall floor, frozen external test, or review backlog
Kweon et al. [13]	Policy-based use of supply-chain malware scores	Recall-first, cost-minimum, and FPR-limited thresholds	2000 balanced private PE files; four models; five-fold evaluation	Introduced Policy A/B/C; no public EMBER/SOREL evaluation, capacity or minimum-recall constraint, frozen external test, paired inference, or review backlog
Present study	Review referral at software artifact gates	Expected cost + FPR + capacity + minimum recall	Fixed scores at policy selection; validation-only thresholds; frozen benchmark tests; split, integrity, and queue evidence	Integrates feasibility, transport assumptions, provenance, and versionable C-SCRM authorization

Disclosure: K.C. and S.S. coauthored [12], and K.C., S.S., and R.Y. coauthored [13]. No private samples, code, trained models, numerical results, tables, figures, or experimental outputs from either study are reused in the present calculations or conclusions.

2.6. Contribution Relative to the Closest Approaches

The closest approaches fall into three groups, and each leaves a gap for review referral at supply-chain gates. Cost-sensitive and Bayes-threshold rules [9,10] assume scores calibrated to the operational prevalence; when they are not, the cost-optimal threshold can exceed review capacity, and these rules provide no mechanism for declaring that no acceptable threshold exists. Selective classification and learning to defer [14,35–37] learn when to abstain or defer; they require retraining or expert-behavior data, optimize coverage or system accuracy, and—with the exception of Alves et al. [36]—do not impose workload limits; none couples a capacity ceiling to an assumed malicious-artifact rate or preserves frozen-test failures. Risk-controlled thresholding [38–40] provides finite-sample guarantees for one chosen risk under exchangeability between calibration and deployment data. In security operations, however, alert volume and false positives are a recognized barrier to analyst effectiveness [41], malware score distributions change over time [42], and benchmark composition keeps evolving [43]; the SOREL score-source result (Section 5.4) shows, in addition, that exchangeability can fail because of how scores are produced, which no within-sample bound detects.

Table 2 therefore separates the elements of the present study by type. The threshold-selection rule itself—cost minimization over an empirical ROC curve subject to constraints—is a standard principle and is not claimed as new. The methodological contribution lies in the joint three-limit formulation for prevalence-dependent workload, the analytical feasibility characterization (structural versus score-dependent infeasibility, the condition under which the recall floor binds, and robust feasibility over an input-uncertainty range), and an evaluation design that keeps validation feasibility separate from frozen-test retention. These characterizations follow from elementary monotonicity properties; robust feasibility describes the observed validation scores under stated input ranges and is not a guarantee of future operational performance. The engineering and governance contribution is the integration of these elements with provenance, score-path verification, exploratory evidence gating, and approval records into one versioned procedure.

Table 2. Elements of the present study, the closest prior approaches, and contribution type.

Element	Closest Prior Approach	Limitation for Gate Referral	What This Study Adds	Type
Selection rule: minimum expected cost subject to constraints	Cost-sensitive learning; ROC operating points [9–11]	None for the rule itself	Applied to fixed public scores under three simultaneous limits	Established principle (not claimed as new)
Joint FPR, prevalence-dependent capacity, and recall floor	Neyman–Pearson classification [38]; risk control [39,40]; workload-constrained deferral [36]	One controlled risk at a time, or workload without a recall floor; capacity not tied to an assumed π	Eligibility set with $R_\pi(t) = \pi\mathrm{TPR}(t) + (1-\pi)\mathrm{FPR}(t) \leq r_{\max}$	Methodological
Feasibility characterization	—	Infeasibility usually unreported or resolved by a fallback	Structural bound; interval form of the eligible set; binding condition for the recall floor; exact frontier and robust feasibility (Section 3.2.5)	Methodological
Validation-only selection, pre-test freezing, failure preservation	Held-out selection in selective classification [37]	Infeasible or failed settings rarely retained	Infeasible and failed outcomes recorded as results; no test-time retuning	Evaluation design and governance
Evidence gate on validation bounds	Risk-controlled calibration [39,40]	Guarantees assume exchangeability	Pointwise one-sided bounds for TPR, FPR, and referral rate as an exploratory screen, not a post-selection guarantee (Section 5.13)	Engineering (adapted)
Score provenance and score-path verification	Artifact provenance (SLSA) [27]	Applied to software artifacts, not to model scores	Versioned model–score–data–approval links; same-pipeline rescoring check (Section 5.4)	Engineering and governance
Workload and backlog evidence	Alert-fatigue studies [41]; workload-constrained deferral [36]	Workload not tied to approval records	Referral and backlog scenarios stored with each approval	Governance

3. Decision Model

3.1. Score and Review-Referral Decision

Let supplier or source u submit artifact x to security gate g. Detector m assigns suspicion score $s_m(x)$, with a larger value indicating greater suspicion. The additional-review decision at threshold τ_R is

$$D_R(x;\tau_R) = \mathbb{I}[s_m(x) \geq \tau_R].$$

If $D_R = 1$, the score is at or above the threshold and the artifact is designated for additional review, such as analyst inspection, sandbox execution, multi-engine scanning, or signature, SBOM, and provenance checks. If $D_R = 0$, the artifact is not designated for this additional review, but it still undergoes the gate's other integrity, provenance, and policy checks. Thus, $D_R = 1$ is neither a malware verdict nor an automatic blocking decision. Lowering τ_R reviews more files and may reduce missed malware but increases benign review workload; raising τ_R reduces workload but may increase missed malware. This study selects τ_R on validation data and freezes it before testing. Table 3 defines the operational terms and the authority responsible for each decision.

Table 3. Operational terms, decision authority, and scope.

Term	Definition	Decided by	What It Does Not Mean
Operational malicious-artifact rate (π)	Assumed fraction of screened artifacts that are malicious at a defined gate and period	Risk owner, from adjudicated outcomes (Section 4.4.4)	The class ratio of a benchmark
Review-referral threshold (τ_R)	Score cutoff selected only from validation data and frozen before testing	Approval authority, on the validation package	A malware verdict or an enforcement rule
Maximum referral rate (r_{max})	Fraction K/N_{op} of arrivals that the review team can complete	Security-operations manager, from completed-review records	A universal staffing constant
Referral ($D_R = 1$)	Score at or above τ_R; artifact routed to additional review	Automated policy, using the approved τ_R	Proof of maliciousness, a block, or a quarantine
Additional review	Sandbox, multi-engine, signature, hash, SBOM, provenance, or analyst review	Analyst, under the review procedure	A finding that the artifact is malicious
Approval candidate	Artifact below τ_R with no independent blocking evidence	Gate controls (hash, signature, SBOM, provenance)	Unconditional approval
Quarantine	Temporary isolation of an artifact pending investigation	Security operations, under an approved procedure	An automatic consequence of referral
Block or reject	Refusal to admit an artifact into acquisition, build, or release	Release authority, on independent evidence	An automatic consequence of a score
Supplier sanction	Contractual or procurement action against a supplier	Procurement or supplier-risk authority, with documented evidence	A consequence of scores or referrals alone
Probability calibration	Agreement between predicted probabilities and observed class frequencies	—	Ranking quality measured by ROC-AUC

Throughout the manuscript, "referral" denotes only $D_R = 1$, and "approval candidate" denotes an artifact below τ_R with no independent blocking evidence. "Quarantine", "block", "reject", and "sanction" denote downstream actions that require the authority listed in Table 3; none follows automatically from a score or a referral.

3.2. Operational Cost, Review Workload, and Threshold Selection

This section presents the full operating sequence in one place: calculate expected cost and review workload, generate candidate thresholds from validation data, apply the policy-specific selection rules, and interpret the Bayes thresholds as comparison baselines. The quantities and a worked count example are given before the selection rules.

3.2.1. Basic Calculations and Worked Example

At candidate threshold t, TPR is the fraction of malicious artifacts referred, FPR is the fraction of benign artifacts incorrectly referred, and FNR = 1 − TPR is the missed-malware rate. The assumed operational malicious-artifact rate is π. The nonnegative weights c_{FN} and c_{FP} represent the relative losses assigned to missed malware and unnecessary benign review. They are dimensionless decision weights, not measured monetary costs. Normalized expected cost per screened artifact is

$$C_\pi(t) = c_{FN}\pi\,\mathrm{FNR}(t) + c_{FP}(1-\pi)\,\mathrm{FPR}(t).$$

The first term is the weighted loss from missed malware; the second is the weighted loss from unnecessary benign review. The expected fraction of all artifacts sent for additional review is

$$R_\pi(t) = \pi\,\mathrm{TPR}(t) + (1-\pi)\,\mathrm{FPR}(t).$$

For N_{op} screened artifacts, expected referrals are $N_{op} \cdot R_{\pi}(t)$, expected benign referrals are $N_{op} \cdot (1 - \pi) \cdot \mathrm{FPR}(t)$, and expected missed malicious artifacts are $N_{op} \cdot \pi \cdot \mathrm{FNR}(t)$. If the team can process at most K cases, its capacity limit is $r_{max} = K/N_{op}$. The threshold fits the workload only when $R_{\pi}(t) \leq r_{max}$.

For example, with 10,000 artifacts per day, $\pi = 1\%$, TPR = 0.90, and FPR = 0.01, 90 malicious and 99 benign artifacts are referred, while 10 malicious artifacts are missed. The total is 189 referrals, or 1.89%. A capacity of 200 reviews per day gives $r_{max} = 2\%$, so this threshold fits the capacity. With c_{FN}:c_{FP} = 10:1, the weighted loss is $10 \times 10 + 1 \times 99 = 199$ per 10,000 artifacts, or $C_{\pi}(t) = 0.0199$ per artifact. The values 1%, 10:1, and 2% are worked examples, not measurements from an organization.

FPR and total referral rate have different denominators. As a contrasting worked example, keep 10,000 arrivals, $\pi = 1\%$, and TPR = 0.90 but set FPR = 1.5%. Expected referrals are 90 malicious plus 148.5 benign files, or 238.5 in total (2.385%). This satisfies an FPR ceiling of 5% but exceeds a 2% capacity ceiling of 200 reviews. Fractional counts are expectations, not observed file counts. Thus, satisfying the FPR ceiling alone does not guarantee manageable workload.

These are scenario calculations. They assume that the class-conditional score behavior measured on the benchmark remains approximately stable while the operational malicious-artifact rate changes. If supplier, time, file type, or collection practice changes the score distributions, $C_{\pi}(t)$ and $R_{\pi}(t)$ are not validated deployment forecasts.

3.2.2. Candidate Thresholds and Validation-Only Selection

The operating candidate set T_{op} contains every distinct positive score observed in the validation data and one data-derived boundary above the maximum score that refers no artifacts. A threshold $t \leq 0$ may be used as an ROC endpoint, but it refers every artifact and does not discriminate, so it is excluded from the operating candidates. Consequently, if a malicious artifact has score zero, even the maximum-recall policy may have recall below 1. At every candidate, TP, FP, TN, FN, TPR, FPR, expected cost, and referral rate are calculated. The threshold is selected only on validation data and then frozen for one application to the test data. Test data do not generate candidates, select or break ties, or retune the threshold.

A frozen-test result is obtained by applying the validation-selected threshold to a separate held-out test set without post hoc retuning, constraint relaxation, or policy replacement. Freezing applies to the selected threshold and the registered evaluation conditions, not to the observed performance. A validation-infeasible profile has no selected threshold and therefore no test result; a validation-feasible profile can still fail one or more test constraints.

3.2.3. Threshold-Selection Policies

The rules are stated below in their actual selection order rather than as dense optimization notation. All comparisons use unrounded validation values.

Maximum-recall policy (code A): choose the candidate with the highest TPR. Break ties by lower expected cost, lower FPR, and then higher threshold. The maximum is compared exactly with recall = recall.max(), so 0.999995 is not treated as equal to 1.0. The representative SOREL validation threshold is 0.00020531 with recall 1.000000. This policy is an extreme comparison rule, not a deployment recommendation.

99%-recall rule (code A99 in the supplement): among candidates with TPR $\geq$ 0.99, choose the one with the lowest expected cost; break ties by lower FPR and then higher threshold. If no candidate reaches 0.99, apply the maximum-recall ordering and record that the 99% target was not met.

Unconstrained minimum-cost policy (code B): choose the candidate with the lowest $C_{\pi}(t)$. "Unconstrained" means that no hard FPR ceiling or review-capacity ceiling is imposed;

it does not mean that errors have no cost. Both missed malware and unnecessary benign review remain in the objective, weighted by their relative losses and the assumed malicious-artifact rate. A cost-minimizing threshold may therefore exceed an organization's review capacity.

FPR-only cost policy (code C_FPR): retain candidates satisfying $\mathrm{FPR}(t) \leq \alpha$ and choose the one with the lowest $C_\pi(t)$. "Only" describes the constraint, not the objective: this policy still minimizes expected error cost. It limits referrals among benign files, but does not impose a ceiling on referrals among all files.

FPR-and-capacity cost policy (code C): retain candidates satisfying both $\mathrm{FPR}(t) \leq \alpha$ and $R_\pi(t) \leq r_{\max}$, then choose the lowest-cost candidate. Policies B, C_FPR, and C use the same validation scores, candidate thresholds, and cost objective; they differ in the constraints on eligible candidates. Adding capacity changes the selection rule, not the FPR ceiling, the capacity value, or the model scores. None of these three policies imposes a minimum recall; Policy Cβ adds that requirement.

Because T_{op} contains a no-referral boundary, Policy C has at least one feasible candidate when $\alpha \geq 0$ and $r_{\max} \geq 0$, although that candidate may have zero recall. The procedure never substitutes a constraint-violating fallback. An input or candidate-generation failure stops the run and is recorded as invalid. Each result retains the policy, inputs, constraint status, and data split.

Table 4 summarizes what each policy prioritizes and what it does not directly control.

Table 4. Concise comparison of threshold-selection policies.

Policy	Simple Goal	Practical Caution
Maximum recall (A)	Find as many malicious files as possible.	Review workload can become very large.
99% recall	Require at least 99% recall, then choose the lowest-cost threshold.	The target may be unavailable in the validation scores.
Unconstrained cost (B)	Minimize weighted missed-malware and benign-review cost.	No hard FPR ceiling, capacity ceiling, or recall floor.
FPR-only cost (C_FPR)	Minimize the same cost subject to the FPR ceiling.	Total review workload and minimum recall are not constrained.
FPR and capacity (C)	Minimize the same cost subject to both ceilings.	Capacity may be satisfied at very low recall; Cβ adds a recall floor.
Recall floor (Cβ)	Apply Policy C and also require a minimum recall.	If no threshold satisfies all conditions, record infeasible without a fallback.

3.2.4. Policy Cβ and Feasibility

The recall-floor policy (code Cβ) adds $\mathrm{TPR}(t) \geq \beta_{\min}$ to Policy C. It first keeps only candidates satisfying $\mathrm{FPR}(t) \leq \alpha$, $R_\pi(t) \leq r_{\max}$, and $\mathrm{TPR}(t) \geq \beta_{\min}$, and then chooses the candidate with the lowest $C_\pi(t)$.

The name Cβ means Policy C augmented by the minimum-recall requirement $\beta_{\min}$; β does not denote the Fβ score or the probability of a Type II error. The evaluated recall floors are 0.80, 0.90, and 0.95, with 0.90 used for the primary interpretation.

Cost ties are resolved by lower FPR, lower referral rate, and then higher threshold. All comparisons use unrounded values. If no candidate satisfies all three conditions, the run records selection_status = infeasible, validation_feasible = False, τ_R = NA, and all test metrics = NA. It does not use the no-referral boundary or another fallback.

Even with ideal FPR = 0, the minimum possible referral fraction is $\pi \cdot \beta_{\min}$. Thus, $\pi \cdot \beta_{\min} > r_{\max}$ is an arithmetic conflict that makes the condition structurally infeasible regardless of the score distribution. If this inequality is not violated but no observed

threshold satisfies recall, FPR, and capacity together, infeasibility is empirical and depends on the validation scores. The selected Cβ threshold is also frozen before testing.

3.2.5. Analytical Properties of the Jointly Constrained Selection

Fixed-score thresholding admits an exact empirical characterization. On T_{op}, $\mathrm{TPR}(t)$, $\mathrm{FPR}(t)$, and $R_\pi(t)$ are non-increasing in t. Propositions 1–3 characterize the feasible set and input robustness using this monotonicity; Proposition 4 states separate sampling and transport assumptions. These are elementary properties and explicit conditions supporting the governance procedure.

Proposition 1 (structural infeasibility)**.** *If $\pi\beta_{min} > r_{max}$, no threshold satisfies $TPR(t) \geq \beta_{min}$ and $R_\pi(t) \leq r_{max}$ for any score distribution, because $R_\pi(t) \geq \pi TPR(t) \geq \pi\beta_{min}$. This justifies the arithmetic precheck before any score is inspected.*

Proposition 2 (interval form and binding recall floor)**.** *Define $B = \{t \in T_{op}: TPR(t) \geq \beta_{min}\}$. If B is empty, Cβ is infeasible; otherwise let $t_\beta = \max B$, $t_\alpha = \min\{t \in T_{op}: FPR(t) \leq \alpha\}$, and $t_r = \min\{t \in T_{op}: R_\pi(t) \leq r_{max}\}$. For nonnegative ceilings, the no-referral boundary makes the last two sets nonempty. The eligible set is $\{t \in T_{op}: \max(t_\alpha, t_r) \leq t \leq t_\beta\}$; hence feasibility is equivalent to $FPR(t_\beta) \leq \alpha$ and $R_\pi(t_\beta) \leq r_{max}$. With the same objective and deterministic tie order, Cβ selects Policy C's threshold if and only if C's validation recall meets β_{min}. Otherwise, when Cβ is feasible, it selects a lower threshold: recall increases, while FPR and referral load cannot decrease. Monotonicity alone does not imply that the selected threshold is t_β. That conclusion additionally requires cost to be non-increasing over the eligible interval, with the stated tie order favoring its largest threshold. A discrete score distribution may overshoot the recall floor substantially; proximity to the floor in Section 5.10 is an empirical finding.*

Proposition 3 (feasibility frontier and robust feasibility)**.** *When t_β exists, the minimum referral rate compatible with the recall floor is $r_{min}(\pi) = FPR(t_\beta) + \pi[TPR(t_\beta) - FPR(t_\beta)]$. A profile is feasible exactly when $FPR(t_\beta) \leq \alpha$ and $r_{max} \geq r_{min}(\pi)$. For uncertainty $\pi \in [\pi_L, \pi_U]$, $\alpha \geq \alpha_L$, $r_{max} \geq r_L$, and $\beta_{min} \leq \beta_U$, define t_β using β_U. A single robust threshold exists exactly when this t_β exists, $FPR(t_\beta) \leq \alpha_L$, and $\max\{R_{\pi L}(t_\beta), R_{\pi U}(t_\beta)\} \leq r_L$. This follows because referral rate is affine in π and every lower threshold has at least as many referrals. If $TPR(t_\beta) \geq FPR(t_\beta)$, the maximum is at π_U and the upper-prevalence corner suffices, as in the reported benchmarks. Without that condition, both endpoints must be checked. Loss weights affect selection among feasible candidates, not feasibility. Robust feasibility is therefore a property of the observed validation-score curve over the stated input ranges, not a guarantee that the limits will hold on future data; carrying it to a deployment population requires the assumptions of Proposition 4.*

Proposition 4 (conditions for transfer)**.** *Invariance of the class-conditional referral probabilities at a fixed τ_R is a sufficient transport assumption for carrying TPR and FPR to another population and reweighting R_π and C_π for its prevalence. Full distributional invariance is stronger than necessary, and neither form of invariance is necessary merely for the inequalities to remain satisfied: changes may remain within available margins. Under independent sampling and a threshold fixed independently of the assessment sample, counts have binomial sampling variation. Validation estimates used to select τ_R do not automatically have fixed-threshold coverage. Large discrepancies warrant investigation of population, labels, sample identities, and scoring settings; within-sample intervals do not identify the cause. A verified scoring path addresses documented implementation differences but cannot alone rule out all provenance or population differences.*

Proposition 1 underlies the precheck in Algorithm 1, Proposition 2 explains when Cβ changes Policy C's decision (Section 5.10), Proposition 3 provides the sensitivity analysis

and robust-feasibility rule (Section 5.9), and Proposition 4 states the assumption behind every prevalence-adjusted count in this paper (Section 7.4).

Algorithm 1. Validation-only selection, freezing, and audit of a review-referral threshold

Inputs: versioned model and scoring program; validation and test (or production) score files with identities; gate scope; approved π (or $[\pi_L, \pi_U]$), c_{FN}:c_{FP}, α, r_{max} (or r_L), and β_{min}.
0. Score path. Verify that validation and test or production scores were produced by the same model checkpoint, feature extraction, preprocessing, and scoring code; otherwise record provenance_unverified and stop.
1. Arithmetic precheck. If $\pi_U \beta_{min} > r_L$, record structural_infeasible and stop (Proposition 1).
2. Candidates. $T_{op} \leftarrow$ distinct positive validation scores $\cup$ {no-referral boundary}; compute TP, FP, TPR, FPR, R_π, and C_π at every candidate without rounding.
3. Eligibility. $E \leftarrow \{t: \mathrm{FPR}(t) \le \alpha, R_\pi(t) \le r_{max}, \mathrm{TPR}(t) \ge \beta_{min}\}$. The exploratory evidence gate substitutes U95(FPR), U95(R_π), and L95(TPR); operational approval additionally requires the selection-aware or independent-sample uncertainty procedure in Section 4.6.
4. Selection. If E is empty, record score_dependent_infeasible (or evidence_insufficient) with τ_R = NA and stop; otherwise select the minimum-C_π candidate, breaking ties by lower FPR, lower R_π, and higher t.
5. Freeze. Version-lock τ_R with the model, score, data, code, and input hashes and the approval record.
6. Frozen test. Apply τ_R once to held-out scores; record TP, FP, TN, FN, recall, FPR, R_π, C_π, and margins m_α, m_r, and m_β; a negative margin is a recorded violation.
7. Decision. Approve for versioned use only if every limit is retained and field validation (Section 8) is complete; otherwise withhold and follow the governance responses in Section 6 without retuning on test data.
Outputs: selection status; τ_R or NA; validation and test metrics; margins; provenance record; approval or withholding record.

3.2.6. Cost-Based Bayes Comparison Thresholds

The Bayes threshold is a comparison baseline, not a new detector. Let q(x) be a calibrated maliciousness probability for the operational environment. Passing the artifact has expected loss q(x)·c_{FN}; referring it has expected loss [1 − q(x)]·c_{FP}. Referral is preferred when the first loss is at least the second, which simplifies to

$$\tau_{\mathrm{Bayes},op} = \frac{c_{FP}}{c_{FN} + c_{FP}}.$$

If $q_{bal}(x)$ is a posterior-like score obtained under a balanced class prior, the operational malicious-artifact rate π must also be included. Under prior-probability shift, the comparison threshold becomes

$$\tau_{\mathrm{Bayes},bal} = \frac{c_{FP}(1-\pi)}{c_{FN}\pi + c_{FP}(1-\pi)}.$$

With the worked values $\pi = 0.01$ and c_{FN}:c_{FP} = 10:1, $\tau_{Bayes,op} = 0.09091$ and $\tau_{Bayes,bal} = 0.90826$. The first assumes that the score is calibrated to the operational maliciousness probability; the second adjusts a balanced-prior score for an operational rate of 1% and is used as the Bayes-prior comparison baseline in this study. A score of 0.8 does not necessarily mean an 80% maliciousness probability. Brier score and ECE are supporting diagnostics, not sufficient evidence that these scores are calibrated operational probabilities. Because operational calibration and prior-shift assumptions have not been established here, the Bayes threshold remains comparison-only; the operational τ_R is selected empirically from validation data.

3.3. Compact Procedure

Algorithm 1 summarizes the prospective approval procedure, including provenance and evidence checks. The historical official-file SOREL analysis is retained as diagnostic benchmark evidence despite unverified source comparability; it does not represent an approved execution past step 0. Steps 1–4 do not use held-out test outcomes; step 6 evaluates the frozen decision.

4. Materials and Methods

4.1. Evidence Roles and Split Protocol

Table 5 distinguishes the two public-score experiments. They evaluate a post-score decision procedure on generic PE malware benchmarks; they do not constitute a supplier-stream, build-pipeline, or supply-chain incident dataset. Private PE samples and private detector-training programs are outside this article and its supplement. The separate BODMAS reference-model fitting procedure is described in Section Additional Check on the Time-Ordered BODMAS Dataset.

Table 5. Public-score evidence, temporal roles, and claim scope.

Evidence	Model/Data	Threshold-Selection Split	Evaluation Split	Claim Scope
EMBER 2018 v2	Fixed scores from the official EMBER 2018 v2 LightGBM 4.6.0 model; no retraining	Appeared = 2018-11 (100,000 rows); random and SHA-256 alternatives in sensitivity analysis	Appeared = 2018-12 (100,000 rows), frozen month-forward evaluation	Within-EMBER temporal holdout of the post-score policy; not independent organizational validation
Unified SOREL official experiment	Official seed0 checkpoint and validation features plus public official test-result CSV	Official validation, 2,495,822 rows	Analyzed test CSV, 4,195,063 rows	τ_R selected on validation and applied once to test

Harmonized Post-Score Evaluation Protocol

EMBER and SOREL use different official malware-detection models, so their raw score scales are not treated as interchangeable. The post-score operational policy is nevertheless evaluated by the same procedure in both datasets. Policy C balances the harm of missed malware against the workload caused by benign referrals and selects the lowest-cost validation threshold that satisfies the FPR and review-capacity limits. Policy Cβ adds a minimum-recall requirement: a threshold is eligible only if recall is at least βmin as well as satisfying the FPR and capacity constraints.

Both datasets use a worked reference profile with a malicious-artifact rate of 1% ($\pi = 0.01$), a false-negative-to-false-positive loss-weight ratio of 10:1 (cFN:cFP = 10:1), an FPR ceiling of 5% ($\alpha = 0.05$), a referral-rate ceiling of 2% ($r_{max} = 0.02$), and a minimum recall of 90% ($\beta_{min} = 0.90$). The 10:1 ratio is a dimensionless decision weight relative to a unit false-positive referral loss, not a measured monetary ratio. The malicious-artifact rate and capacity values are also illustrative and are not estimates from organizational records. A threshold is selected from validation data only and frozen before one application to the separate test data; test labels never participate in threshold selection or retuning.

Frozen-test evaluation reports recall, FPR, referral rate, and expected error cost. Repeated resampling of the same labeled test files (paired bootstrap) estimates a range for the Policy Cβ-minus-Policy C difference. McNemar's test separately counts files on which

only one policy agrees with the benchmark label. After testing, a 30-day first-in, first-out calculation assumes 10,000 daily arrivals and at most 200 completed reviews per day to show whether unfinished reviews accumulate. The datasets and score generators differ, but the same policy definitions, threshold controls, metrics, same-file comparisons, and backlog assumptions are used. For SOREL, the frozen thresholds are evaluated on the prespecified official test-result file and, as a score-source verification, on a same-pipeline rescoring of the official test features (Section 4.3).

4.2. EMBER Fixed-Score Experiment

The experiment uses 200,000 official EMBER 2018 v2 test-feature rows and fixed scores from the official LightGBM model [5,44]. Program S9 reads the official appeared field and stops with an error unless the first 100,000 rows are exactly 2018-11, the final 100,000 are exactly 2018-12, each month contains 50,000 malicious and 50,000 benign labels, JSONL labels match the score-array order, and the two index sets are disjoint. Thresholds are selected only on November and frozen for December. This one-month forward hold-out tests temporal extrapolation within EMBER but is exactly the previously reported original-row-order partition; it is not a new independent dataset or organization-log validation. Label-stratified random (seed = 42) and real-file-SHA-256 partitions remain sensitivity analyses of the same records. The assignment audit and all policy rows are Files S31 and S30.

4.3. SOREL Official Validation/Test Experiment

The SOREL experiment pairs 2,495,822 validation scores regenerated from the official validation-features.npz and seed0 checkpoint with 4,195,063 rows in the released seed0 test-result CSV [6,7]. The NPZ is obtained from the official SOREL URL, is 23,790,175,898 bytes, and has SHA-256 0e3775f6d581a2e0248171b9b9b755cbd9a0846e1e47e265f238538cfd306e67. The regenerated validation CSV is expected to contain 2,495,822 data rows, occupy 106,605,252 bytes, and have SHA-256 f3ab2cc253f86250ac4873ce11b182f425876516393587e5816ca60e7b93c9c9.

The original SOREL article reports 4,195,042 test examples. The public test-result CSV contains 1,360,622 malicious and 2,834,441 benign rows (4,195,063 in total), whereas the official test-feature archive contains 4,195,042 rows (1,360,614 malicious and 2,834,428 benign). The SOREL article lists 8 malicious and 13 benign validation files fewer than the validation archive used here, matching the class-count difference between the two test sources. This is consistent with possible overlap involving 21 validation records, but class totals alone cannot establish membership or bound the overlap. The test-feature NPZ provides no matching file identifiers for a verified join; the public SOREL evaluation code writes SHA identifiers as a CSV index [7], but the identifier field of the analyzed result file was not audited here. The validation NPZ is scored in chunks, and the threshold is fixed before any test score is evaluated. The published checkpoint is used without retraining.

Score-source verification. After the primary analysis, Program S19 rescored all 4,195,042 rows of the official test-feature archive (39,987,140,938 bytes; SHA-256 ddb960b…3b2e89) with the same seed0 checkpoint and the same steps used for the validation scores (float32 conversion and Booster.predict). No common file-level mapping or row alignment between the result file and the feature archive has been verified, so the two test-score sources are compared through class-conditional score distributions and frozen-threshold decisions rather than matched files (Files S54–S56). Because the official result file was the prespecified test source, it remains the primary SOREL analysis; replacing it after the fact with the source that gives more favorable results would itself be a post hoc substitution. Both sources are therefore reported side by side (Section 5.4), and every SOREL test value names its source.

4.3.1. Models, Checkpoints, and Score Provenance

Table 6 links each result to the program or model associated with its score and states whether the generating path was verified. EMBER uses only the released fixed LightGBM model, and the numerical SOREL experiment uses only the seed0 LightGBM model. The official SOREL release also includes FFNN seed0 checkpoints from epoch_1.pt through epoch_10.pt, and its README uses epoch_9.pt in an evaluation example. This study did not download, run, or evaluate any FFNN checkpoint or FFNN score file. The FFNN row records availability only and is not connected to a threshold, table, figure, or numerical result.

Table 6. Provenance from each model or checkpoint to validation and test scores, with the verification status of each generating path.

Data/Panel	Exact Model/Checkpoint	Validation-Score Generation	Frozen Test Score	Column, Direction, Consistency
EMBER/LightGBM	ember_model_2018.txt SHA-256 509de4a8…fff5e2	test_features.jsonl; PEFeatureExtractor v2; 2381 features; November p_val	December p_test in same fixed array; no retraining	predict output ↑ = malicious; full regeneration bitwise identical, max $\Delta = 0$
SOREL/LightGBM seed0	baselines/checkpoints/lightGBM/seed0/lightgbm.model SHA-256 6880f2b4…52c508 (validation scores: path verified)	validation-features.npz arr_0/arr_1 → lightgbm_seed0_validation_scores.csv	baselines/results/lightgbm/seed0/results.csv (official file; identity verified; generating checkpoint and settings unverified)	label_malware/pred_malware; higher = malicious; rows, bytes, hash locked (file identity only)
SOREL/LightGBM seed0, same-pipeline test rescoring (sensitivity analysis)	Same checkpoint (SHA-256 6880f2b4…52c508)	Not applicable; validation scores as above	test-features.npz → lightgbm_seed0_test_scores_regenerated.csv (4,195,042 rows; SHA-256 18c3f6e8…5e5319)	Archive labels; Booster.predict; higher = malicious; generating path verified; compared distributionally (Files S54–S56)
SOREL/FFNN seed0	epoch_1.pt–epoch_10.pt; official evaluation example epoch_9.pt	Not run; no validation score	Official FFNN results.csv identified only; not used	label_malware/pred_malware; excluded from every numerical claim and decision

Program S9 uses code version cbeta-temporal-paired-bootstrap-2026-09-05; the executed generator SHA-256 is 644c74e0…fc4022 (full digest in File S49). The EMBER model SHA-256 is 509de4a8…fff5e2. The released features.py applies PEFeatureExtractor(feature_version = 2) to produce 2381-dimensional vectors, which are passed to Booster.predict in 20,000-row chunks. No detector retraining or score calibration is performed.

SOREL validation scores are generated with –force-recompute by converting validation-features.npz arr_0 to float32 and using arr_1 as labels. The output is external_data/sorel_official/lightgbm_seed0_validation_scores.csv (2,495,822 rows; SHA-256 f3ab2cc2…b93c9c9). The validation-selected threshold is frozen and applied once to external_data/sorel_baseline/lightgbm_seed0_results.csv, the official baselines/results/lightgbm/seed0/results.csv (4,195,063 rows; SHA-256 14d94d6a…2e09b7c).

A SHA-256 value in Table 6 fixes the identity of a file; it does not establish how the file was generated. The generating path was verified for every score produced in this study: EMBER scores by bitwise-identical regeneration, SOREL validation scores and the same-pipeline test rescoring by logged regeneration from the identity-locked checkpoint and feature archives, and BODMAS scores by generation with recorded settings and independent recomputation (Program S14). The official SOREL test-result file is the exception.

The release places it under the seed0 LightGBM baseline, but the analyzed artifacts do not document the checkpoint, feature version, or prediction settings that produced it; its generating path is therefore recorded as unverified in Table 6 and in File S49, which states the verification status of every score source (see also Section 5.4).

4.3.2. Score-Column Semantics and Consistency Gates

For SOREL, label_malware is the ground-truth column and pred_malware is the malicious-score column. A larger value means greater maliciousness, and pred_malware $\geq \tau R$ triggers review referral. EMBER Booster.predict outputs and the BODMAS score field use the same direction. Regenerated EMBER scores were bitwise identical to the fixed arrays (max_abs_delta = 0); the SOREL validation CSV is tied to a –force-recompute log plus row, byte, and SHA-256 identities, and the official test CSV has a separate official path and hash. Files S49 and S50 provide the consolidated model-and-score record and 19 stop-on-failure checks. Program S19 additionally verifies that the official result file reproduces the recall and FPR of every thresholded SOREL row in File S7 to within 10^{-12}.

4.4. Prespecified Operating Conditions

The prespecified condition set is $\pi \in \{0.001, 0.01, 0.05\}$; c_{FN}:$c_{FP} \in \{1{:}1, 5{:}1, 10{:}1, 20{:}1\}$; $\alpha \in \{0.005, 0.01, 0.02, 0.05\}$; and $r_{max} \in \{0.005, 0.01, 0.02, 0.05\}$. This gives 192 profiles per dataset. The recall floor is a separate experiment axis and does not redefine the legacy 192 profiles. The main Cβ analysis contains $2 \times 192 \times 3 = 1152$ rows, and the three-way EMBER split analysis contains $3 \times 192 \times 3 = 1728$ Cβ rows. The primary profile is $\pi = 0.01$, c_{FN}:c_{FP} = 10:1, $\alpha = 0.05$, $r_{max} = 0.02$, and N_{op} = 10,000.

The representative condition is a worked example, not an empirically calibrated organizational setting. $\pi = 0.01$ is a rare-event scenario chosen to expose the operational burden of false referrals [8]; cFN:cFP = 10:1 is the middle dimensionless asymmetric-loss setting in the prespecified design [9,10], not a monetary estimate; $\alpha = 0.05$ is the least restrictive FPR ceiling examined; and rmax = 0.02 represents capacity to complete 200 reviews per 10,000 arrivals. The full analysis uses all 192 prespecified operating conditions per dataset to report feasibility, threshold changes, and whether the approved limits remain satisfied on test data. By contrast, the detailed tables, same-file resampling, McNemar test, and 30-day review-backlog calculation use only the representative condition. These representative results explain one worked example and are not additional evidence for the entire set of conditions. No cited source, expert elicitation, or organizational record establishes the representative values for deployment. Before deployment, File S34 requires π to be estimated from confirmed outcomes for a defined gate and period; loss weights to be based on incident-loss records, review cost, and documented expert judgment; α to be approved from a representative benign-referral budget; and rmax to be estimated from completed reviews, arrivals, service-time variation, and a backlog allowance. Each value requires a source, time window, uncertainty statement, and named approval.

The four operating variables must share one intake gate, target population, adjudication rule, measurement window, and file unit. The malicious-artifact rate π is $M_{confirmed}/N_{screened}$ for that scope and is not the benchmark class ratio. This quotient is valid only when every screened file receives independent final adjudication; under partial verification, use the probability-sampling estimator in Section 4.4.4. The capacity limit rmax is K_{eff}/N_{op}, where K_{eff} is sustainable completed-review capacity after an urgent-work reserve and queue-safety margin. α is the tolerated FPR ceiling on representative adjudicated benign traffic, and βmin is the required minimum TPR (recall) on adjudicated malicious cases. Table 7 locks their estimators and approval evidence.

Table 7. Operational definitions, evidence, and prespecified study values for π, α, rmax, and βmin.

Variable	Operational Estimator and Scope	Required Evidence and Uncertainty Rule	Study Grid	Primary
π	$M_{\text{confirmed}}/N_{\text{screened}}$; same gate, population, adjudication, and window. Valid only with complete independent adjudication; otherwise use the inverse-probability-weighted two-stratum estimate (Section 4.4.4).	Record counts, exclusions, owner, and interval; use a conservative upper stress estimate	0.001, 0.01, 0.05	0.01
α	FP/N_{benign} on representative adjudicated benign traffic. With partial verification, estimate from inverse-probability-weighted outcomes in both strata.	Approve benign-referral budget/SLA; one-sided 95% FPR upper bound must be ≤α	0.005, 0.01, 0.02, 0.05	0.05
rmax	$K_{\text{eff}}/N_{\text{op}}$; completions less urgent-work reserve and queue margin	Arrival/completion times, service time, shifts, SLA; use a conservative lower capacity bound	0.005, 0.01, 0.02, 0.05	0.02
βmin	TP/(TP + FN) on adjudicated malicious cases, stratified by risk tier. With partial verification, estimate from inverse-probability-weighted outcomes in both strata.	Approve miss consequences/risk appetite; one-sided 95% TPR lower bound must be ≥βmin	0.80, 0.90, 0.95	0.90

Before inspecting scores, the necessary arithmetic condition is $\pi\beta\text{min} \leq \text{rmax}$. If it fails, even zero FPR exceeds capacity at the recall floor, so the profile is recorded as structural_infeasible. Otherwise the capacity-implied benign-FPR ceiling at βmin is $\alpha\text{cap} = (\text{rmax} - \pi\beta\text{min})/(1 - \pi)$, giving the preliminary ceiling $\alpha\text{eff} = \min(\alpha, \alpha\text{cap})$. Because a candidate TPR may exceed βmin, final eligibility still checks its exact $R\pi = \pi\text{TPR} + (1 - \pi)\text{FPR} \leq \text{rmax}$.

The registered conditions and representative values are scenario inputs, not deployment estimates. The following interval rule is prospective and does not alter the point-estimate selections; Section 5.13 provides an exploratory pointwise-bound comparison, not a post-selection coverage guarantee. Approval first checks $\pi U\beta\text{min} \leq r_L$ and the robust endpoint condition of Proposition 3, then requires upper FPR and referral-rate bounds and a lower TPR bound to meet the approved limits. The uncertainty procedure must have prespecified joint validity for the selection design, or use independent approval data after freezing (Section 4.6). Sparse or boundary-crossing evidence remains evidence_insufficient or infeasible. No constraint is silently relaxed.

4.4.1. Factorial Construction and Profile Identification

The 192 profiles are the complete Cartesian product of 3 malicious-artifact rates, 4 false-negative-to-false-positive loss-weight ratios, 4 FPR ceilings, and 4 review-capacity ceilings ($3 \times 4 \times 4 \times 4 = 192$); no sampled or hand-picked combination is omitted. Within each dataset there are 64 profiles per π value and 48 profiles per loss-weight ratio, α value, or rmax value. With Nop = 10,000, rmax = 0.005, 0.01, 0.02, and 0.05 correspond to maximum review loads of 50, 100, 200, and 500 files per 10,000 arrivals, respectively.

Program S9 enumerates π in the outer loop, followed by cFN, α, and rmax, and assigns the one-based profile_index in that deterministic order. Its operating_config_id is pi = <π>|cost = <cFN>:1|alpha = <α>|r_max = <rmax>. Thus profile 1 is pi = 0.001|cost = 1:1|alpha = 0.005|r_max = 0.005, the illustrative primary setting is profile 111 in the same

registry, and profile 192 is pi = 0.05|cost = 20:1|alpha = 0.05|r_max = 0.05. File S36 records all 192 rows exactly.

4.4.2. Per-Profile Threshold Experiment

Each operating profile follows steps 2–6 of Algorithm 1 on its own registered parameters; Policy Cβ repeats the selection for $\beta_{\min} = 0.80$, 0.90, and 0.95, an empty eligible set is recorded as infeasible with τ_R and test metrics set to NA, and test labels are used only to evaluate frozen decisions, never to form, select, break ties among, or retune candidates.

For every frozen test decision, the program compares the referral flag with the benchmark label and counts TP, FP, TN, and FN. It calculates recall, FPR, referral rate, expected cost, and whether each approved constraint is satisfied from those counts. It then combines FPR and FNR with the condition's π and Nop to report scenario-based expected referrals, benign referrals, and missed malicious files. Policies are compared only when they use the same dataset and operating_config_id. File S37 maps each analysis step to its output file and separates results for all prespecified conditions from the paired-bootstrap and review-backlog analyses performed only for the representative condition.

4.4.3. Numbers of Conditions and Output Records

The integrated threshold, validation, and test logs Files S4, S6 and S7 each contain 4224 rows: 2 datasets × 192 operating profiles × 11 policy or baseline families. The 11 families are eight non-Cβ policies/baselines (Policy A, 99%-recall rule, B, C, C_FPR, fixed 0.5, Bayes-prior, and the FPR-constrained baseline) plus the three Cβ recall floors. The Cβ-only Files S18 and S19 tables therefore contain 1152 rows ($2 \times 192 \times 3$), while the three-way EMBER split results in Supplementary output File S24 contain 1728 rows ($3 \times 192 \times 3$). These counts are assertions checked by the release audit, not rounded descriptions.

The representative condition ($\pi = 0.01$, cost ratio 10:1, $\alpha = 0.05$, rmax = 0.02) is used only for detailed tables, same-file uncertainty estimates, and the review-backlog illustration. Feasibility, threshold-change, and regression counts use every registered condition. Thus, denominators of 384 represent 2 datasets × 192 conditions, and each dataset-by-β feasibility denominator is 192. File S35 summarizes the common post-score procedure, and Files S36 and S37 provide the exact condition list and experiment map. The 192 profiles of one dataset share the same validation and test scores and differ only in their parameters, and several share parameter values. The 384 or 1152 rows are therefore an exhaustive enumeration of a registered design, not independent experiments, organizations, or trials. Counts such as 176/384 describe how much of the registered grid exhibits a property; they are not effect probabilities, improvement rates, or field-failure rates, and no percentage or hypothesis test is attached to them.

4.4.4. Outcome Adjudication and Estimation Under Partial Verification

File S52 distinguishes the simple quotient under complete independent adjudication from the probability-sampled estimate under partial verification. If only referred files are adjudicated, this ratio counts only the malicious files that the threshold already caught: it underestimates π by the miss rate and leaves recall unidentifiable, because missed malicious files are never observed (verification bias). The protocol therefore adjudicates two strata: every referred file, or a known-probability sample of them, and an audit sample of non-referred files drawn with a fixed, recorded inclusion probability p_N.

Let N_R and N_N denote referred and non-referred file totals, and m_R and m_N the adjudicated malicious counts under known, strictly positive inclusion probabilities p_R and p_N. Horvitz–Thompson totals are $\hat{M}_R = m_R/p_R$ and $\hat{M}_N = m_N/p_N$; prevalence is estimated by $(\hat{M}_R + \hat{M}_N)/(N_R + N_N)$, while recall and FPR use ratios of the corresponding weighted

totals. These ratio estimators are not generally unbiased in finite samples. For unequal sampling, sum each adjudicated outcome divided by its own inclusion probability. Intervals must respect the sampling design. A conservative recall lower bound uses simultaneous bounds $L(M_R)$ and $U(M_N)$ as $L(M_R)/[L(M_R) + U(M_N)]$ when defined; an upper bound for M_N alone is insufficient if M_R is also sampled. Sparse or zero observed misses require nonzero conservative upper bounds, rather than an ordinary bootstrap that cannot generate unobserved events. If the malicious denominator is not identifiable or evidence is too sparse, record evidence_insufficient.

Sample size is a binding issue. With 10,000 daily arrivals, $\pi = 1\%$, and recall 0.92, about 8 malicious files per day are missed. If FPR is illustratively 0.0028, expected referrals are 92 malicious plus 27.72 benign files, leaving about 9880 non-referred files. A 5% audit sample (about 494 files daily) then contains an expected 0.4 missed malicious files per day, or about 12 over 30 days. Estimates therefore require an approval window, design-based uncertainty, and an explicit evidence-sufficiency rule. Risk-tiered sampling can improve precision in priority strata when inclusion probabilities and weights are recorded; it does not automatically reduce variance for every aggregate estimate.

Adjudication uses the same written evidence rule in both strata, for example sandbox and multi-engine evidence followed by analyst confirmation. Fix a maturity window W after arrival (for example, 30 days), retain later reclassifications as versioned updates, and report pending outcomes separately. Pending-case bounds must be specific to the estimand: prevalence is bounded by assigning all pending files benign versus malicious, but the lowest recall assigns pending referred files benign and pending non-referred files malicious; the highest recall reverses those assignments. Apply the corresponding inverse-probability weights. The recall-minimizing assignment gives the upper FPR bound, and the recall-maximizing assignment gives the lower FPR bound. The label rule, p_R, p_N, W, pending counts, assignments, and uncertainty method belong in the approval record (File S52).

4.5. Direct Baselines and Ablation

Direct baselines are the fixed threshold 0.5, the balanced-prior Bayes threshold, and an FPR-constrained threshold. To remove ambiguity, the FPR-constrained baseline is a recall-maximizing comparator, not the cost-minimizing C_FPR ablation. On validation data, it chooses the threshold $t \in T_{op}$ that maximizes $TPR(t)$, subject to $FPR(t) \leq \alpha$. It imposes neither the review-capacity constraint $R_\pi(t) \leq r_{max}$ nor the minimum-recall constraint $TPR(t) \geq \beta_{min}$.

Among candidates attaining the same maximum validation recall, ties are resolved lexicographically by (1) lower malicious-artifact-rate-aware expected cost $C_\pi(t)$, (2) lower $FPR(t)$, and (3) higher threshold t. Because T_{op} includes the data-derived no-referral boundary with FPR = 0, this baseline is feasible for every non-negative α. Legacy ablations compare Policy A, the 99%-recall rule, and Policy B with Policy C. The C_FPR ablation instead minimizes $C_\pi(t)$ subject only to $FPR(t) \leq \alpha$, with ties resolved by lower FPR and then higher t; its paired comparison with Policy C therefore isolates the incremental effect of adding review capacity.

4.6. Metrics and Statistical Analysis

Thresholds are selected from validation data only and evaluated once on fixed test decisions. The reported measures are ROC-AUC, average precision, Brier score, ten-bin expected calibration error, recall, FPR, referral rate, and expected cost. Wilson 95% intervals describe binomial uncertainty for one policy. Paired comparisons of Policy Cβ and Policy C use the same frozen test files. The estimand is the difference $\Delta = M(C\beta) - M(C)$ in a test metric M (recall, FPR, referral rate R_π at the profile's π, or expected cost C_π) for

the benchmark's test distribution. The resampling unit is the individual test file: files are resampled with replacement separately within the malicious and benign groups, preserving the class counts (50,000 and 50,000 for EMBER; 1,360,622 and 2,834,441 for the official SOREL file), and the two frozen decisions of each sampled file are kept together. $B = 5000$ resamples are drawn (seeds in File S32), and the interval is the percentile 95% interval of the bootstrap differences [45]. The exact two-sided McNemar test counts files that one policy classifies correctly and the other incorrectly [46]. Neither procedure selects or adjusts thresholds. The six representative comparisons are descriptive, and no hypothesis test is attached to the exhaustive 176/384 or 178/384 counts across all conditions.

Brier score is the mean squared difference between each probability-like score and its binary label (malicious = 1, benign = 0). It measures overall probabilistic prediction quality, including calibration and discrimination, and depends on class prevalence. ECE uses ten equal-width bins over $[0, 1]$; within each nonempty bin it takes the absolute difference between the mean score and observed malicious fraction, then weights that difference by the bin's sample fraction. It is not a per-file error or a classification error rate, and its value depends on binning. These diagnostics use each benchmark partition's observed class mixture, not the illustrative operational rate of 1%, and neither selects a policy threshold. EMBER/SOREL bins include their left endpoints, with 1 included in the last bin; BODMAS retains its right-endpoint convention, with 0 included in the first bin. The supplementary revision note documents this implementation difference without changing reported values.

These intervals quantify sampling variability among the analyzed benchmark files only. They do not include uncertainty in π, the loss weights, staffing capacity, or future score distributions, which is addressed instead by the registered grid, the sensitivity analysis in Section 5.9, and the transfer conditions in Section 7.4. Statistical detectability is also distinct from operational importance: with 100,000 or more paired files, differences of a few files per 10,000 arrivals can produce very small p-values. Paired differences are therefore reported as effect sizes per 10,000 arrivals with their intervals, and decisions rest on constraint retention and operational magnitude rather than on p-values.

Evidence gate. For the worked comparison in Section 5.13, one-sided 95% Clopper–Pearson bounds are computed for validation TPR (lower) and FPR (upper), and a one-sided normal bound is computed for $R_\pi = \pi\text{TPR} + (1 - \pi)\text{FPR}$ as a linear combination of two independent binomial proportions, with π held at the profile value. A threshold passes the gate if $\text{L95(TPR)} \geq \beta_{\min}$, $\text{U95(FPR)} \leq \alpha$, and $\text{U95}(R_\pi) \leq r_{\max}$. For BODMAS, whose frozen scores are distributed, Policy Cβ is re-selected over all candidates under the gate; for EMBER and SOREL, whose raw scores are not redistributed, the gate is applied to every point-selected threshold. Because the gate's eligible set is a subset of the point-estimate set and the objective is unchanged, a point-selected threshold that passes the gate is also the gate's selection. These are pointwise nominal bounds used for an exploratory sensitivity comparison. Searching thresholds on the same validation sample does not preserve a post-selection 95% guarantee; the three reported marginal bounds are not a simultaneous confidence region, and the referral bound is approximate and can be unreliable at sparse or extreme counts. Organizational certification therefore requires a prespecified procedure with jointly valid bounds over the candidates, or assessment of the frozen threshold on an independent approval sample with an explicit joint error budget [39,40]. Under partial verification, use design-appropriate weighted intervals rather than binomial bounds for raw sampled counts. Files S60 and S61 remain unchanged and do not establish such certification.

4.7. Input, Cache, and Independent-Recomputation Controls

Program S9 validates at startup that labels are exactly {0, 1}, scores are finite and in [0–1], validation and test paths are distinct, and expected row counts match. Offi-

cial SOREL status requires role-specific filename, row-count, byte-count, and SHA-256 identities; swapped or arbitrary paired CSVs stop with an error. The score cache stores validation/test/model and feature-extractor identities, scoring parameters, schema version, score-generation code version, and generator SHA-256. –force-recompute bypasses cache reads and rewrites a new cache; –no-cache disables reads and writes.

Historical lock values test stability, not correctness. A separate reference implementation uses stable ascending sorting, cumulative counts at first indices, and searchsorted confusion-matrix evaluation. It independently reselects thresholds on validation and evaluates frozen thresholds on test. The two layers therefore check both invariance and implementation agreement.

Additional Check on the Time-Ordered BODMAS Dataset

Supplementary Program S13 uses BODMAS Cleaned v1.0, a cleaned derivative of the public BODMAS Windows PE malware dataset that includes collection-time information. It contains 134,428 rows and 2326 numeric static-PE features. The 3712 rows without timestamps are excluded rather than reassigned. The time-ordered split contains 80,250 training rows before May 2020, 23,969 validation rows from May–June 2020, and 26,497 frozen-test rows from July–September 2020, with no overlap. The registry of 192 prespecified conditions (SHA-256 73d0d30b...e1a9d3; full value in File S38) is fixed before features are read or models are fitted. The data are IT4LIA AI Factory's BODMAS Cleaned v1.0 derivative of the original BODMAS corpus and were accessed on 8 September 2026 [47,48]. The exact inputs are bodmas_clean.npz, bodmas_clean_X.npy, bodmas_clean_y.npy, and bodmas_clean_metadata.parquet. File S0_source_provenance.csv records each direct URL, byte count, and complete SHA-256 value so that readers can verify file identity.

LightGBM uses n_estimators = 240, learning_rate = 0.05, num_leaves = 63, max_bin = 127, and colsample_bytree = 0.4; ExtraTrees uses n_estimators = 160, max_features = sqrt, and min_samples_leaf = 2. LinearSGD runs eight epochs with log_loss and elasticnet (alpha = 10^{-5} l1_ratio = 0.1), fitting StandardScaler (with_mean = False) on training rows only. The seed is 20260909. These models are fit in memory and no serialized checkpoint is claimed. Validation scores alone select thresholds; frozen-test scores are used only for evaluation.

4.8. Review Backlog Calculation

For each primary Policy Cβ test referral rate, daily referrals and same-day FIFO service are defined by

$$A_d \sim \text{Binomial}(N_{op}, R_\pi), \quad K = r_{\max} N_{op}, \quad Q_d = \max\{0, Q_{d-1} + A_d - K\}.$$

The calculation uses $N_{\text{op}} = 10{,}000$, $K = 200$, a 30-day horizon, 1000 replications, and seed 20260903. It is performed only after threshold selection and test evaluation and does not adjust τ_{R}. The result is a scenario-based illustration of how unfinished review work accumulates under stated arrival and service assumptions, not a validated staffing forecast, because it omits variation in service time, case priority, shifts, holidays, absences, and rework.

4.9. Additional Post Hoc Analyses

Program S20 (Files S57–S64) adds four analyses computed only from distributed outputs: (i) a classification of every Cβ profile by whether the recall floor changed Policy C's decision and by its frozen-test outcome on each test-score source; (ii) one-factor-at-a-time variation of π, c_{FN}:c_{FP}, α, $r_{\max}$, and $\beta_{\min}$ around the representative profile; (iii) the exact feasibility frontier $r_{\min}(\pi)$ of Proposition 3, recovered for EMBER and SOREL from

the minimum validation referral rate at the recall floor recorded at $\pi = 0.001$ and 0.05 and checked at $\pi = 0.01$ to within 2×10^{-9}, and for BODMAS from the frozen scores; and (iv) the exploratory evidence-gate comparison. Program S20 reproduces the registered BODMAS selections exactly before applying the gate. No prespecified threshold or result is changed.

4.10. Generative AI Assistance

Generative AI tools were used to assist with English-language editing, terminology consistency, formatting, and code and document checks. Anthropic Claude (Claude Opus 5.5 (Model ID: claude-opus-5-5)) was used between 24 and 28 September 2026 to assist with Program S19 and the audit-program revisions, and during the first- and second-round peer-review revisions (October 2026) to draft revised text for the authors' review, to write the revision analysis, audit, and figure-rendering programs (Programs S20–S22), and to check consistency between the manuscript and the supplementary outputs. OpenAI ChatGPT (GPT-6.1-Sol (gpt-6.1-sol))/Codex 0.162.1 (Codex CLI) assisted with a further consistency review and proposed editorial and audit corrections on 15 October 2026. Generative AI was not used to generate or alter experimental data, labels, model outputs, or prespecified threshold selections. All analyses were executed from the distributed outputs, and the authors verified all content, numerical results, citations, and conclusions.

5. Results

5.1. Score Discrimination and Probability Reliability Diagnostics

Table 8 reports discrimination and probability-reliability diagnostics for the validation and test partitions.

SOREL Brier/ECE increased from 0.00616/0.00667 on validation data to 0.08215/0.11527 on the official test-result file. These partition-level diagnostics show that the official file's scores are less reliable as probabilities; they do not establish calibrated operational probabilities, prove distribution shift, or identify the cause of frozen-test constraint violations. Discrimination, calibration, and constraint retention are distinct properties (Section 7.3): an ROC-AUC of 0.98128 on the official file coexisted with an FPR at the frozen threshold 77 times its validation value, and Section 5.4 shows that the same test features scored by the validation pipeline behave very differently. Constraint retention is evaluated directly from test FPR, recall, and referral rate.

Table 8. Score quality by validation/test partition.

Dataset	Partition (Rows)	ROC-AUC	Average Precision	Brier	ECE-10
EMBER	Validation (100,000)	0.99741	0.99779	0.01563	0.01279
EMBER	Test (100,000)	0.99537	0.99606	0.02215	0.01802
Unified SOREL	Official validation (2,495,822)	0.99921	0.99904	0.00616	0.00667
Unified SOREL	Public test CSV (4,195,063)	0.98128	0.96611	0.08215	0.11527

5.2. Primary Validation Selection

Table 9 reports the validation-selected operating points under the representative profile.

Policy B and Policy C select the same primary thresholds because the validation cost minima already meet $\alpha = 0.05$ and $r_{\max} = 0.02$. Policy A retains only candidates whose validation recall equals the observed maximum exactly; the implementation history of this rule is documented in the Supplementary README.

Table 9. Primary-profile validation selection results.

Dataset	Policy	τ_R	Recall	FPR	Expected Cost	Referral Rate	Legacy Constraints Met
EMBER	Policy A: exact recall first	9.23×10^{-9}	1.00000	0.79844	0.79046	0.80046	No
EMBER	99%-recall rule	0.02113	0.99000	0.04812	0.04864	0.05754	No
EMBER	Policy B: cost minimum	0.99244	0.94444	0.00308	0.00861	0.01249	Yes
EMBER	Policy C: cost + FPR + capacity	0.99244	0.94444	0.00308	0.00861	0.01249	Yes
SOREL	Policy A: exact recall first	0.00021	1.00000	0.92699	0.91772	0.92772	No
SOREL	99%-recall rule	0.33013	0.99000	0.00689	0.00782	0.01672	Yes
SOREL	Policy B: cost minimum	0.76511	0.97673	0.00084	0.00316	0.01060	Yes
SOREL	Policy C: cost + FPR + capacity	0.76511	0.97673	0.00084	0.00316	0.01060	Yes

5.3. Frozen-Threshold Test Results

Table 10 evaluates those frozen thresholds on the held-out test scores, using the official SOREL test-result file.

Table 10 reports point estimates. EMBER Policy C achieved recall 0.91954 (Wilson 95% CI 0.91712–0.92189), FPR 0.00312 (0.00267–0.00365), and 122.84 referrals per 10,000. On the official SOREL result file, the validation-selected $\tau_R = 0.76511$ yielded recall 0.94082 (0.94042–0.94122), FPR 0.06428 (0.06400–0.06457), and 730.50 referrals per 10,000; the violating threshold was not retuned.

Under this identical primary operating policy, EMBER kept both the FPR and referral load within their declared limits: Policy C and Policy Cβ at $\beta_{min} = 0.90$ referred 122.84 files per 10,000. On the official SOREL result file, however, the frozen threshold produced FPR = 0.06428 and 730.50 referrals per 10,000, violating both the 5% FPR ceiling and the 2% capacity ceiling while retaining recall. Section 5.4 shows that the observed violation depends on the test-score source; without file-level alignment, the cause of the difference remains unresolved. The contrast is not a head-to-head claim that one underlying detector is superior.

Table 10. Primary-profile test results (EMBER temporal holdout; SOREL official test-result file).

Dataset	Policy	τ_R	Recall	FPR	Expected Cost	Referral Rate	Referrals/10,000	False Referrals	Expected Misses
EMBER	Policy A	9.23×10^{-9}	1.00000	0.84930	0.84081	0.85081	8508.07	8408.07	0.00
EMBER	99%-recall rule	0.02113	0.97882	0.05202	0.05362	0.06129	612.88	515.00	2.12
EMBER	Policy B	0.99244	0.91954	0.00312	0.01113	0.01228	122.84	30.89	8.05
EMBER	Policy C	0.99244	0.91954	0.00312	0.01113	0.01228	122.84	30.89	8.05
SOREL	Policy A	0.00021	1.00000	0.99587	0.98591	0.99591	9959.15	9859.15	0.00

Table 10. *Cont.*

Dataset	Policy	τ_R	Recall	FPR	Expected Cost	Referral Rate	Referrals/10,000	False Referrals	Expected Misses
SOREL	99%-recall rule	0.33013	0.98637	0.23641	0.23541	0.24391	2439.13	2340.49	1.36
SOREL	Policy B	0.76511	0.94082	0.06428	0.06956	0.07305	730.50	636.42	5.92
SOREL	Policy C	0.76511	0.94082	0.06428	0.06956	0.07305	730.50	636.42	5.92

5.4. SOREL Frozen-Test Outcomes by Test-Score Source

Table 11 applies the same validation-selected SOREL thresholds to the two test-score sources: the prespecified official result file (primary analysis) and the same-pipeline rescoring (sensitivity analysis). On the official result file, the representative threshold $\tau_R = 0.76511$ violated the FPR and capacity ceilings. On the same-pipeline rescoring of the official test-feature archive, the same threshold produced FPR 0.00060 and 100.65 referrals per 10,000—close to its validation values of 0.00084 and 105.96—and retained all limits. Across the registered grid, Policy C met its FPR and capacity ceilings in 83 of 192 profiles on the official file and in all 192 on the same-pipeline scores, and Policy Cβ retained all limits in 3/0/0 versus 128/112/20 of 128 validation-feasible profiles at $\beta_{min} = 0.80/0.90/0.95$. Every same-pipeline Cβ violation was a recall-floor violation. Across all 1920 frozen SOREL thresholds of every policy, at least one limit outcome changed for 1468 (File S54).

Table 11. SOREL frozen-test outcomes by test-score source for the same validation-selected thresholds (File S56).

Quantity	Official Result File (Prespecified)	Same-Pipeline Rescoring
Test rows	4,195,063	4,195,042
Benign-score median/99th percentile	0.05555/0.95641	0.00173/0.23748
Representative $\tau_R = 0.76511$: recall/FPR	0.94082/0.06428	0.94748/0.00060
Referrals per 10,000 (validation: 105.96)	730.50	100.65
Expected cost: Policy C/fixed 0.5/Bayes prior	0.06956/0.16030/0.04493	0.00584/0.00551/0.00770
Mean day-30 backlog at $\beta_{min} = 0.90$	15,924.94	0.00
Policy C FPR and capacity met (of 192)	83	192
Policy C zero test recall (of 192)	16	0
Cβ all limits retained at $\beta_{min} = 0.80/0.90/0.95$ (of 128)	3/0/0	128/112/20
Cβ recall/FPR/capacity violations (of 384)	228/286/300	124/0/0

Table 12 separates observed differences from explanations that require matched records. The official result file has 8 more malicious and 13 more benign rows than the test-feature archive. If every other row were the same file with the same label, referring all 13 extra benign rows could change FPR by at most 4.6×10^{-6}, far below the observed 0.0637 gap; that identical-membership condition has not been verified. Similar class totals do not establish equal sample membership or label definitions. The label_malware and pred_malware columns and ROC-AUC 0.98128 support the reported score direction, but do not identify a row-level cause. The rescored archive has test FPR 0.00060 versus validation FPR 0.00084, which weakens a simple drift-only account for that archive; the official result file remains unmatched. The benign-score medians differ (0.05555 versus 0.00173), but this could reflect

sample membership, labels, model checkpoint, feature processing, prediction settings, or a combination. The producing settings of the official file are not documented in the analyzed artifacts. The public evaluation code writes SHA identifiers as a CSV index [7]; we did not audit the identifier field of the analyzed CSV, and the feature NPZ lacks corresponding identifiers for a verified join.

Table 12. Candidate explanations of the SOREL score-source discrepancy (Files S55, S56 and S62).

Candidate Explanation	Evidence	Status
Sample membership (21 extra rows)	8 malicious and 13 benign additional rows; a 4.6×10^{-6} FPR bound applies only if all remaining files and labels match, which has not been verified	Conditional bound only; membership unresolved
Label definition	Class totals differ by 21; no file-level label alignment	Unresolved
Score column or direction	label_malware/pred_malware, higher = malicious; official-file ROC-AUC 0.98128 supports direction	Direction supported; row-level semantics unverified
Change in the test population (temporal drift)	Same-pipeline archive FPR 0.00060 versus validation 0.00084; official-file membership and score provenance not matched	Drift-only attribution unsupported; population difference unresolved
Producing model, checkpoint version, feature-processing version, or prediction settings of the official file	No verified producing settings or common file-level join; benign median 0.05555 versus 0.00173; public evaluation code writes SHA as CSV index [7]	Checkpoint, processing, settings, and membership unresolved
Row-level pairing of the two sources	No verified shared file-level mapping or row alignment	No verified pairing; comparison is distributional and decision-level

The provenance checks therefore support only limited conclusions. They establish the identity of each analyzed file (rows, bytes, SHA-256), the class totals of each source, the score column and direction of the official file, that the official file reproduces the recall and FPR recorded in File S7 at every frozen threshold, and that the rescored archive was generated by the identity-locked seed0 checkpoint with the validation steps (File S55). They also show that the two sources are not aligned by row position, because their row counts and stored label sequences differ (File S55). They do not establish whether the two sources contain the same files, whether shared files carry the same labels, or which checkpoint, feature version, and settings produced the official scores. The available evidence therefore cannot determine whether the discrepancy is confined to score values or also involves sample identity, labels, or row alignment. The same-pipeline rescoring is a sensitivity analysis of the score-generation path; it does not validate, correct, or replace the prespecified official-file result, which remains the primary SOREL finding.

Two consequences follow. First, the official-file FPR increase should not be assigned to temporal deterioration or a specific scoring change without matched records and does not, by itself, justify additional staffing; the official-file backlog in Section 5.16 shows what would happen if a threshold were transferred to scores from an unverified generation path. Second, verifying that validation and deployment scores come from the same versioned model, feature extraction, and scoring code is a prerequisite for transferring any threshold (Algorithm 1, step 0; Section 6). One finding holds on both sources: the $\beta_{\min} = 0.95$ recall floor failed on test in 108 of 128 validation-feasible profiles on the same-pipeline scores and

in all 128 on the official file, and recall at the representative threshold fell from 0.97673 on validation to 0.94748 on the same-pipeline test scores, consistent with a later test window.

5.5. Direct Baselines

Table 13 compares the frozen Policy C threshold with the fixed 0.5, Bayes-prior, and FPR-constrained baselines on the test partitions.

Table 13. Test-set direct-baseline comparison (EMBER temporal holdout; SOREL official test-result file).

Dataset	Rule	τ_R	Recall	FPR	Expected Cost	Referrals/10,000
EMBER	Policy C	0.99244	0.91954	0.00312	0.01113	122.84
EMBER	Fixed 0.5	0.50000	0.96636	0.01748	0.02067	269.69
EMBER	Bayes-prior	0.90826	0.95086	0.00806	0.01289	174.88
EMBER	FPR-constrained baseline	0.01847	0.97898	0.05374	0.05530	629.92
SOREL	Policy C	0.76511	0.94082	0.06428	0.06956	730.50
SOREL	Fixed 0.5	0.50000	0.97693	0.15959	0.16030	1677.66
SOREL	Bayes-prior	0.90826	0.84042	0.02927	0.04493	373.81
SOREL	FPR-constrained baseline	0.06710	0.99592	0.47849	0.47411	4836.63

The prior-adjusted Bayes threshold reduced expected cost relative to the fixed 0.5 threshold in both datasets on these sources but was not uniformly better than direct validation selection. On the same-pipeline SOREL scores (Table 11), the ordering changed: the fixed 0.5 threshold had a slightly lower expected cost (0.00551) than Policy C (0.00584) and the Bayes-prior threshold (0.00770), and all three retained the representative limits. The difference between Policy C and the fixed threshold, 0.00033 per artifact or 3.3 weighted-loss units per 10,000 arrivals, is small. Policy C is therefore not claimed to minimize test cost; its role is to select, on validation data alone, a threshold that satisfies approved limits and to record when none exists. The Bayes threshold's probability and prior-shift assumptions are not guaranteed by the public score generators.

5.6. Separate Effects of FPR and Review Capacity

Across all registered conditions, Policy C met its FPR and capacity constraints on test in 192/192 EMBER profiles and in 83/192 SOREL profiles on the official result file (192/192 with same-pipeline scores; Section 5.4). On the official file, SOREL had exactly 16/192 Policy C profiles with zero test recall (0/192 with same-pipeline scores). These values are unchanged by adding Policy Cβ.

Table 14 compares validation-selected thresholds across 384 registered profiles (192 per dataset). Policy B and Policy C selected different thresholds in 178 of the 384 profiles, which measures the joint addition of the FPR and capacity constraints to unconstrained cost minimization. Policy C_FPR and Policy C differed in 176 profiles; their common FPR constraint isolates the incremental effect of adding capacity within this experiment. In all 176 profiles, the C_FPR-selected referral rate exceeded the registered capacity ceiling.

Table 14. Joint-constraint and isolated capacity-effect counts.

Dataset	Profiles	B vs. C Joint Limits	C_FPR vs. C Added Capacity
EMBER	192	90	88
SOREL	192	88	88
Total	384	178	176

The two additional B-versus-C differences were EMBER profiles with $\pi = 1\%$, loss ratio 20:1, $\alpha = 0.5\%$, and capacity 2% or 5%. The FPR-only policy already selected the same threshold as Policy C, so adding capacity caused no further change. Consequently, 178 is not the capacity-only count. "Threshold changed" denotes a difference between policies selected on the same validation data, not a revision made after inspecting test results.

Figure 1 shows where adding the capacity ceiling changes the FPR-constrained validation threshold across the registered profiles.

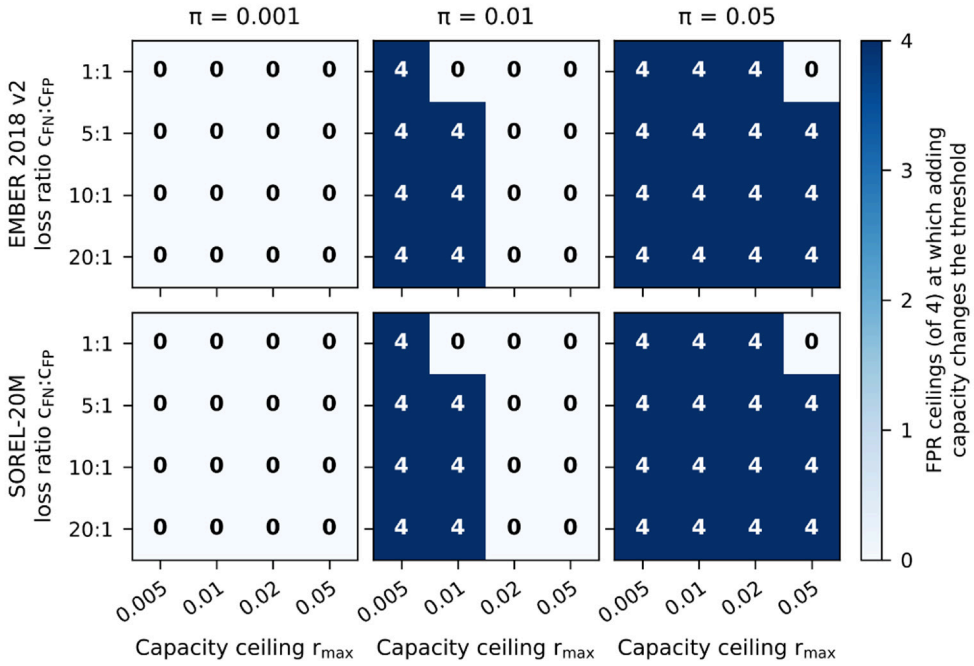


Figure 1. Number of FPR ceilings (of four) at which adding the review-capacity ceiling to FPR-constrained cost minimization changes the validation-selected threshold, by dataset, malicious-artifact rate, loss ratio, and capacity ceiling (88 EMBER and 88 SOREL profiles; 176 of 384). Source: File S13_fpr_only_vs_fpr_capacity_heatmap_table.csv.

The denominator 384 is the union of two complete 192-profile registries, not repeated tests of the primary setting. Each dataset contributes 64 profiles at each π value and 48 profiles at each cost ratio, FPR ceiling, or capacity ceiling, and File S36 allows each aggregate cell to be traced to its exact operating_config_id. These profiles share scores and contain repeated parameter combinations; they are not independent organizations or trials, and the counts describe coverage of the registered grid, not deployment-failure probabilities or real-world improvement rates.

5.7. EMBER Split Sensitivity

Under the label-stratified random and real-file SHA-256 partitions, the primary Policy C threshold was 0.99133 and 0.98504 (month-forward: 0.99244), with test recall 0.93270 and

0.94090 (0.91954), test FPR 0.00322 and 0.00430 (0.00312), and 125.15 and 136.66 referrals per 10,000 (122.84); the complete rows are in Files S15 and S30.

The capacity-change count was stable across the three EMBER partitions, but thresholds and metrics varied. The appeared-field audit confirms the original split as November selection followed by December evaluation; because it reuses the same benchmark records, it is temporal validation, not independent external validation. A separately sampled later collection or governed supplier stream is still required.

5.8. Policy Cβ Feasibility

Table 15 summarizes validation feasibility, infeasibility classes, and frozen-test retention for each dataset and recall floor.

Table 15. Policy Cβ validation feasibility, infeasibility class, and test retention (SOREL: official test-result file).

Dataset	$\beta_{\min}$	Total	Validation Feasible	Structural Infeasible	Empirical Infeasible	Test All Constraints	Threshold Differs from Policy C
EMBER	0.80	192	128	64	0	68	16
EMBER	0.90	192	112	64	16	28	56
EMBER	0.95	192	96	64	32	0	88
SOREL	0.80	192	128	64	0	3	0
SOREL	0.90	192	128	64	0	0	0
SOREL	0.95	192	128	64	0	0	32

Of 1152 registered Cβ profiles, 432 were validation-infeasible and 720 admitted a threshold. The 384 structural cases satisfy $\pi\beta_{\min} > r_{\max}$ before any score is examined: even zero benign referrals would exceed capacity at the recall floor. For each of the three recall floors, the conflicting (malicious rate, capacity) pairs are (1%, 0.5%), (5%, 0.5%), (5%, 1%), and (5%, 2%). Four such pairs times four loss ratios times four FPR ceilings give 64 cases per dataset–recall-floor cell, or 384 across two datasets and three floors. Loss ratios affect cost minimization, not the existence of a jointly feasible candidate.

The remaining 48 cases were score-dependent and occurred only on EMBER. At malicious rate 1% and capacity 1%, achieving recall 0.90 required a minimum referral rate of 1.01682%; recall 0.95 required 1.34238%. At malicious rate 5% and capacity 5%, recall 0.95 required 5.1279%. Each setting occurred at four loss ratios and four FPR ceilings, giving 16 + 16 + 16 = 48 profiles (File S21). Under the FPR and capacity limits, the corresponding maximum recalls were 89.488%, 89.488%, and 94.214%. In these cases capacity was exceeded even though the arithmetic lower bound alone allowed feasibility; the existing validation scores supplied no qualifying threshold. The 432/1152 fraction is grid coverage, not an estimated field-failure probability.

Table 16 separates frozen-test retention by the recall, FPR, and capacity constraints among validation-feasible profiles.

With same-pipeline SOREL scores, all 384 validation-feasible SOREL profiles met the FPR and capacity ceilings. The recall floor was met in 128, 112, and 20 profiles at $\beta_{\min} = 0.80$, 0.90, and 0.95 (Section 5.4).

Figure 2 displays the corresponding validation-feasibility counts across the registered FPR and review-capacity ceilings.

Table 16. Constraint-specific Cβ test retention among validation-feasible profiles (SOREL: official test-result file).

Dataset	β_{min}	Validation Feasible	Recall Floor Met	FPR Ceiling Met	Capacity Met	All Three Met
EMBER	0.80	128	68	128	128	68
EMBER	0.90	112	28	112	112	28
EMBER	0.95	96	0	96	96	0
SOREL	0.80	128	96	38	32	3
SOREL	0.90	128	40	38	32	0
SOREL	0.95	128	20	22	20	0

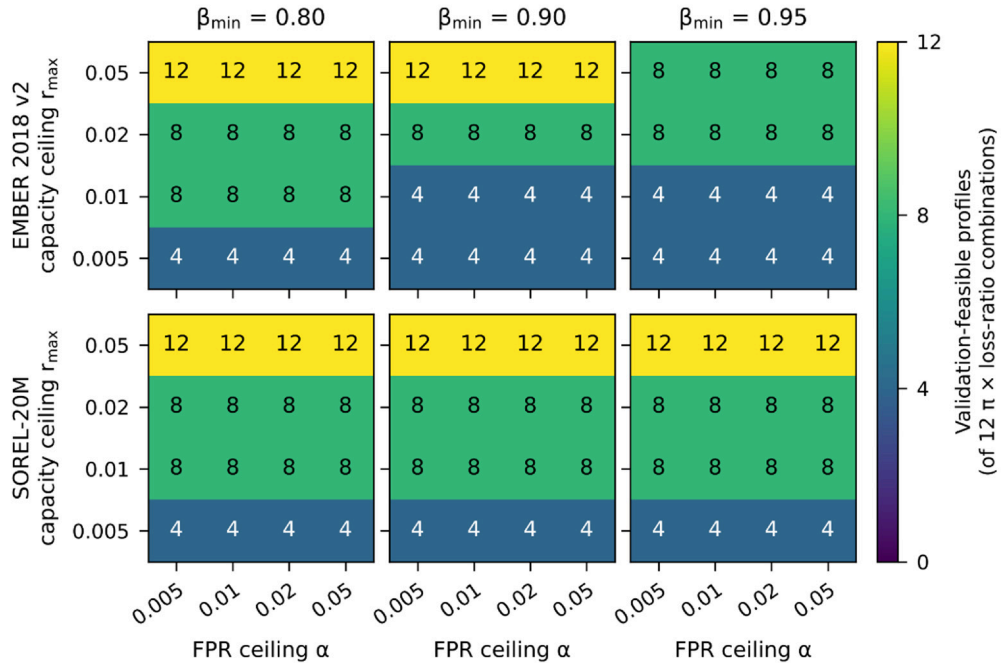


Figure 2. Number of validation-feasible Policy Cβ profiles (of the 12 combinations of malicious-artifact rate and loss ratio) by dataset, β_{min}, FPR ceiling, and review-capacity ceiling. Validation feasibility does not depend on the SOREL test-score source. Source: File S25_cbeta_heatmap_table.csv.

5.9. Sensitivity of Feasibility and Retention to Operating Inputs

Table 17 varies one input at a time around the representative profile ($\pi = 0.01$, 10:1, $\alpha = 0.05$, $r_{max} = 0.02$, $\beta_{min} = 0.90$); every row is an existing registered profile (File S58). First, varying α alone in this local analysis changed neither feasibility nor the selected threshold. Across the full grid, α did not determine Cβ feasibility because FPR(t_β) was at most 0.396% for EMBER and 0.016% for SOREL, below the smallest ceiling of 0.5%; it did affect cost-minimizing selection in some profiles. For EMBER at $\pi = 1\%$, loss ratio 20:1, and capacity 2% or 5%, all three recall floors selected $\tau_R = 0.98315$ at $\alpha = 0.5\%$ versus 0.95570 at $\alpha = 1\%$, 2%, or 5%. Second, π and r_{max} govern feasibility: capacity 1% made the representative EMBER case score-dependent infeasible, and $\pi = 5\%$ made both representative cases structurally infeasible. Third, the loss ratio changes selection and frozen-test retention without changing feasibility: EMBER ratios 1:1 and 5:1 failed recall 0.90 on test, whereas 10:1 and 20:1 retained all limits.

Table 17. One-factor-at-a-time sensitivity around the representative profile (File S58).

Varied Input	Value	EMBER: Validation	EMBER: τ_R	EMBER: Frozen Test	SOREL: Validation	SOREL: τ_R	SOREL: Official File	SOREL: Same Pipeline
Representative profile	—	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
π	0.001	Feasible (floor binds)	0.99919	Fails: recall	Feasible	0.90741	Fails: recall, capacity	Retained
π	0.05	Structural	—	—	Structural	—	—	—
c_{FN}:c_{FP}	1:1	Feasible (floor binds)	0.99919	Fails: recall	Feasible	0.90741	Fails: recall, capacity	Retained
c_{FN}:c_{FP}	5:1	Feasible	0.99864	Fails: recall	Feasible	0.81689	Fails: FPR, capacity	Retained
c_{FN}:c_{FP}	20:1	Feasible	0.95570	Retained	Feasible	0.65770	Fails: FPR, capacity	Retained
α	0.005	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
α	0.01	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
α	0.02	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
r_{max}	0.005	Structural	—	—	Structural	—	—	—
r_{max}	0.01	Score-dependent	—	—	Feasible	0.87074	Fails: recall, capacity	Retained
r_{max}	0.05	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
β_{min}	0.8	Feasible	0.99244	Retained	Feasible	0.76511	Fails: FPR, capacity	Retained
β_{min}	0.95	Feasible (floor binds)	0.98566	Fails: recall	Feasible	0.76511	Fails: recall, FPR, capacity	Fails: recall

Statuses refer to validation selection; "floor binds" means that Cβ selected a threshold different from Policy C. Frozen-test outcomes are for EMBER's December holdout and for SOREL's official result file and same-pipeline rescoring.

Figure 3 turns Proposition 3 into a continuous sensitivity analysis. Because validation feasibility depends on π and r_{max} only through $r_{min}(\pi) = \mathrm{FPR}(t_\beta) + \pi[\mathrm{TPR}(t_\beta) - \mathrm{FPR}(t_\beta)]$, the frontier is a straight line for each dataset and recall floor (File S59). At the representative $\pi = 1\%$, EMBER required at least 0.83%, 1.02%, and 1.34% review capacity for $\beta_{min} = 0.80$, 0.90, and 0.95, and SOREL required 0.80%, 0.90%, and 0.97%. Conversely, with $r_{max} = 2\%$, $\beta_{min} = 0.90$ remained feasible up to $\pi = 2.09\%$ on EMBER and 2.22% on SOREL, whereas $\beta_{min} = 0.95$ on EMBER remained feasible only up to $\pi = 1.69\%$.

This yields a decision rule when inputs are uncertain. Suppose an organization can state only that π lies between 0.5% and 2% and that sustainable capacity is at least 2%. By Proposition 3, a robust threshold exists if and only if the worst-case corner ($\pi = 2\%$, $r_{max} = 2\%$) is feasible. For $\beta_{min} = 0.90$, the corner requires 1.92% capacity on EMBER and 1.80% on SOREL, so robust thresholds exist on the observed validation scores; for $\beta_{min} = 0.95$ on EMBER it requires 2.29%, so no threshold is robust over the stated range. The procedure then records the profile as not robustly feasible and does not deploy; the organization must narrow the uncertainty with adjudicated data (Section 4.4.4), increase sustainable capacity, add an upstream filter, or formally accept a lower recall floor (Section 6). Robust validation

feasibility remains only a necessary condition: retention must still be checked on frozen test data, as the loss-ratio rows of Table 17 show.

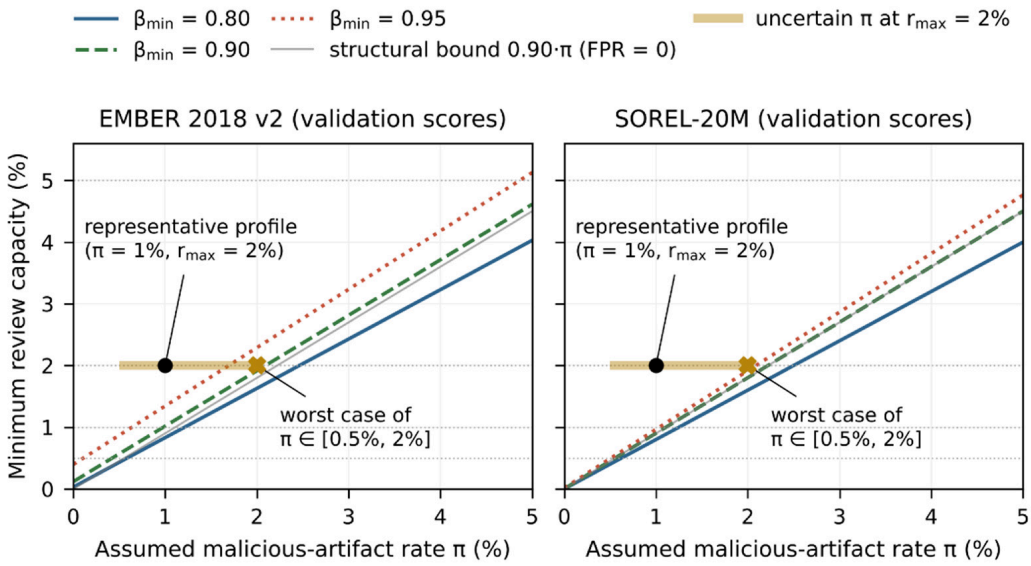


Figure 3. Exact validation feasibility frontier for Policy Cβ: the minimum review capacity $r_{\min}(\pi) = \mathrm{FPR}(t_\beta) + \pi[\mathrm{TPR}(t_\beta) - \mathrm{FPR}(t_\beta)]$ needed to meet the recall floor, as a function of the assumed malicious-artifact rate π (Proposition 3). Profiles above a line are validation-feasible (provided $\alpha \geq \mathrm{FPR}(t_\beta)$, as for feasibility in the registered grid). The thin grey line is the structural bound $0.90{\cdot}\pi$ at FPR = 0; dotted horizontal lines mark the registered capacity ceilings; the shaded bar marks π uncertain in [0.5%, 2%] at $r_{\max} = 2\%$, with its worst-case corner (×). Source: File S59.

5.10. Policy Cβ Versus Policy C: When the Recall Floor Changes the Decision

By Proposition 2, Policy Cβ selects Policy C's threshold whenever Policy C's validation recall already meets $\beta_{\min}$; this held for all 720 validation-feasible profiles (File S57). Table 18 classifies every registered profile accordingly. On SOREL, the floor never changed the decision at $\beta_{\min} = 0.80$ or 0.90 and changed it in 32 profiles at 0.95. On EMBER, it changed the decision in 16, 56, and 88 profiles at $\beta_{\min} = 0.80$, 0.90, and 0.95.

Table 18. When Policy Cβ equals Policy C, changes the decision, is infeasible, or fails on test (File S57).

Dataset	$\beta_{\min}$	Structural	Score-Dependent	Cβ = C: Profiles (Retained)	Cβ ≠ C: Profiles (Retained)	Main Test Failure
EMBER	0.80	64	0	112 (68)	16 (0)	Recall floor
EMBER	0.90	64	16	56 (28)	56 (0)	Recall floor
EMBER	0.95	64	32	8 (0)	88 (0)	Recall floor
SOREL	0.80	64	0	128 (3/128)	0	FPR and capacity (official); none (same pipeline)
SOREL	0.90	64	0	128 (0/112)	0	FPR and capacity (official); recall (same pipeline)
SOREL	0.95	64	0	96 (0/20)	32 (0/0)	Mixed (official); recall (same pipeline)

Retained = all three limits met on frozen test data. For SOREL, retained counts are given as official result file/same-pipeline rescoring.

When the floor changed the decision, the selected threshold sat at the floor: its validation recall exceeded $\beta_{\min}$ by only 0.0002–0.0120. Every such EMBER threshold then failed the recall floor on the December test data (16/16, 56/56, and 88/88), because test recall at a fixed threshold was 0.023–0.088 lower than validation recall; the 32 changed SOREL profiles failed the floor on both test-score sources. In this study, therefore, a binding recall floor bought higher validation recall at the cost of more referrals but did not secure the floor on later data; retention required a margin above the floor that the validation data did not supply. Where the floor was not binding, Cβ and C are identical by construction and share their frozen-test outcomes. Mean and median Cβ-minus-C test deltas remain available in File S23.

5.11. Primary Policy Cβ Profiles

Table 19 reports the primary-profile frozen-test results at each minimum-recall setting, with SOREL evaluated on the official test-result file.

Table 19. Primary-profile Policy Cβ test results by recall floor (SOREL: official test-result file).

Dataset	$\beta_{\min}$	τ_R	Recall	FPR	Referral Rate	Referrals/10,000	All Three Constraints
EMBER	0.80	0.99244	0.91954	0.00312	0.01228	122.84	Yes
EMBER	0.90	0.99244	0.91954	0.00312	0.01228	122.84	Yes
EMBER	0.95	0.98566	0.92792	0.00384	0.01308	130.81	No
SOREL	0.80	0.76511	0.94082	0.06428	0.07305	730.50	No
SOREL	0.90	0.76511	0.94082	0.06428	0.07305	730.50	No
SOREL	0.95	0.76511	0.94082	0.06428	0.07305	730.50	No

The primary EMBER $\beta_{\min} = 0.90$ threshold maintained all constraints. The corresponding SOREL threshold maintained the recall floor but violated both the FPR and capacity ceilings on the official result file; on the same-pipeline scores it retained all limits at $\beta_{\min} = 0.80$ and 0.90 (Table 11). The $\beta_{\min} = 0.95$ primary rows failed to maintain their test recall floor in both datasets and on both SOREL sources.

5.12. Policy Cβ Split Sensitivity

Across the 1728 Cβ split rows, original-order validation-feasible counts at $\beta_{\min} = 0.80/0.90/0.95$ were 128/112/96 and test all-constraint counts were 68/28/0. The stratified-random partition produced 128/112/60 feasible and 96/40/60 retained profiles; the real-SHA-256 partition produced 128/112/60 and 102/98/60. At $\beta_{\min} = 0.90$, the primary thresholds for original/random/SHA-256 splits were 0.99244/0.99133/0.98504, with test recall 0.91954/0.93270/0.94090, FPR 0.00312/0.00322/0.00430, and 122.84/125.15/136.66 referrals per 10,000.

5.13. Exploratory Confidence-Based Evidence Gate: Worked Comparison

Sections 4.4 and 7.5 describe prospective uncertainty checks, whereas the registered experiments select on point estimates. Table 20 is an exploratory comparison using pointwise nominal bounds. A point-selected threshold that passes this gate remains the gate's selection because eligibility only shrinks and the objective is unchanged; a failing threshold can change or be withheld. This set-inclusion result does not provide post-selection confidence coverage (Section 4.6).

Table 20. Exploratory evidence gate (pointwise bounds) versus point-estimate selection for Policy Cβ (Files S60 and S61).

Data/Model	β_{min}	Point-Selected	Unchanged by Gate	Fails Recall Bound	Fails Capacity Bound	Frozen-Test Retained: Point → Gated
EMBER	0.80	128	104	0	24	68 → —
EMBER	0.90	112	68	32	12	28 → —
EMBER	0.95	96	8	88	16	0 → —
SOREL	0.80	128	104	0	24	3/128 → —
SOREL	0.90	128	104	0	24	0/112 → —
SOREL	0.95	128	72	32	24	0/20 → —
BODMAS LightGBM	0.80/0.90/0.95	128 each	100 each	0	28 each	100 → 116
BODMAS ExtraTrees	0.80/0.90/0.95	128 each	112 each	0	16 each	116/100/28 → 128/112/40
BODMAS LinearSGD	0.80/0.90/0.95	24 each	24 each	0	0 each	24 → 24

No profile failed the FPR bound. For EMBER and SOREL, full re-selection under the gate requires the raw scores, which are not redistributed (—); SOREL retention is given as official result file/same-pipeline rescoring. For BODMAS, every gate failure led to a changed threshold and no profile became evidence-insufficient.

For EMBER and SOREL, 460 of the 720 point-selected Cβ thresholds passed the gate unchanged. The remaining 260 sat at a binding boundary: 152 failed the recall bound (EMBER at $\beta_{min} = 0.90$ and 0.95; SOREL at 0.95) and 124 the capacity bound, where Policy C's cost minimum is attained exactly at r_{max}; 16 failed both. At the representative profile, the $\beta_{min} = 0.90$ thresholds of both datasets passed (EMBER: L95(TPR) = 0.94273, U95(FPR) = 0.00352, U95(R_π) = 0.01290), whereas the EMBER $\beta_{min} = 0.95$ threshold (validation recall 0.95072, L95(TPR) = 0.94910) would have been withheld.

The pointwise gate adds a sampling-uncertainty margin but is not a simultaneous or post-selection risk guarantee and does not address distribution change. On EMBER at $\beta_{min} = 0.90$, it would have withheld 32 of the 56 floor-binding thresholds; the other 24 passed yet still failed the recall floor on test, because the gate's margin (about 0.002 with 50,000 malicious validation files) is an order of magnitude smaller than the observed validation-to-test recall decrease of 0.049–0.058 at those thresholds. Conversely, 12 non-binding EMBER thresholds at the capacity boundary that the gate would have withheld retained all limits on test, illustrating its conservatism.

For BODMAS, full re-selection under the gate (File S61) left feasibility unchanged for all three models—no profile became evidence-insufficient—and changed the threshold in 28 LightGBM and 16 ExtraTrees profiles per β_{min} and in no LinearSGD profile. The changed thresholds were capacity-boundary selections: in 120 of the 132 changed cases, the point-selected referral rate had exceeded r_{max} on test. The gate moved them to slightly higher thresholds, lowering test referral rates by 0.00026–0.00037 on average at a mean test-recall cost of 0.005 (ExtraTrees) to 0.011 (LightGBM), and 92 of the 132 retained all limits. Frozen-test retention rose from 100 to 116 of 128 profiles per β_{min} for LightGBM and from 116/100/28 to 128/112/40 for ExtraTrees (Section 5.17). The observed improvement is consistent with adding a capacity margin, but this comparison does not establish that the original failures were caused by sampling error. The pointwise gate provides no guarantee against systematic changes in recall or score generation.

5.14. Recalculation and Integrity Evidence

The completed full recomputation re-created EMBER's 200,000 labels and scores from official features and the fixed model, with bitwise identity to the fixed arrays (maximum absolute difference 0). A later run recorded a validated cache hit. Negative tests swapped or altered SOREL inputs and failed before policy calculation. The new SOREL score-only run used –force-recompute to regenerate the complete validation CSV from the official NPZ and checkpoint in a new path; its row count, byte count, and SHA-256 matched the prespecified identity.

The independent reference implementation matched every legacy Policy B, C_FPR, and Policy C threshold and validation/test metric. It re-derived 178/384 joint changes and 176/384 incremental capacity changes without using those historical constants as selection inputs. The fixed-lock and independent-recomputation layers therefore address different questions: whether prior results changed and whether the production calculation agrees with an independent method.

5.15. Policy Differences and Uncertainty on the Same Test Files

Under the representative condition, Policies Cβ and C made identical test decisions at $\beta_{\min} = 0.80$ and 0.90 in both datasets and on both SOREL sources, so every paired difference and interval was zero and the exact McNemar p-value was 1. For EMBER at $\beta_{\min} = 0.95$, Cβ minus C changed recall by 0.00838 (95% paired-bootstrap interval 0.00760 to 0.00920), FPR by 0.00072 (0.00048 to 0.00096), referral rate by 0.00080 (0.00056 to 0.00104), and expected cost by −0.000125 (−0.000367 to 0.000125); 419 files were correct only under Cβ and 36 only under C (exact McNemar $p = 7.44 \times 10^{-84}$). Per 10,000 arrivals at $\pi = 1\%$, this is about 0.8 additional malicious referrals and 7.1 additional benign referrals (8.0 referrals in total), with an expected-cost change of −1.25 weighted-loss units (interval −3.67 to +1.25). The very small p-value reflects 100,000 paired files and a joint increase in recall and FPR. Because the cost interval includes zero, it does not show that Cβ lowers operational cost, and the $\beta_{\min} = 0.95$ threshold failed its recall floor on test (Table 19). These intervals describe sampling variation among the benchmark test files, not uncertainty in π, loss weights, capacity, or future score distributions (Section 4.6). The SOREL decisions at $\beta_{\min} = 0.95$ were also identical under the representative condition.

5.16. Review Backlog Under the Representative Condition

Table 21 compares daily referrals, capacity, and backlog for the representative EMBER and SOREL cases (File S63). At $\beta_{\min} = 0.90$, EMBER's 122.84 expected referrals per day used 61% of the 200 daily reviews and left no backlog; even with 20% more arrivals the load ratio would be 0.74, and backlog would begin only above about 16,300 arrivals per day. On the official SOREL file, 730.50 referrals per day produced a mean day-30 backlog of 15,924.94 (p95 16,159.00); on the same-pipeline scores, 100.65 referrals per day (load ratio 0.50) produced none. These calculations are scenario-based illustrations of workload accumulation under fixed arrivals and service; they are not validated staffing forecasts and do not justify threshold retuning or, for the official SOREL file, additional staffing (Section 5.4). Figure 4 shows the corresponding simulated backlog trajectories.

For the official file, an operational decomposition clarifies the source of the overload. From Table 10, 730.50 total referrals minus 636.42 false referrals implies approximately 94.08 malicious referrals per 10,000 arrivals. At capacity $K = 200$, only about 105.92 slots would remain for benign referrals; holding the observed recall fixed solely for diagnostic interpretation, capacity compatibility would require an FPR of approximately 105.92/9900 = 0.0107 (1.07%), compared with 0.06428 on the official file and 0.00060 on the

same-pipeline scores. This decomposition interprets the frozen result and is neither a new selection rule nor post-test retuning.

Table 21. Thirty-day review backlog for primary Policy Cβ profiles (1000 replications; scenario illustration).

Dataset	β_{min}	Expected/Day	Capacity/Day	Load Ratio	Mean Day-30 Backlog	p95 Backlog	Stable in Expectation
EMBER	0.80	122.84	200	0.614	0.00	0.00	Yes
EMBER	0.90	122.84	200	0.614	0.00	0.00	Yes
EMBER	0.95	130.81	200	0.654	0.00	0.00	Yes
SOREL (official file)	0.80	730.50	200	3.653	15,908.24	16,135.00	No
SOREL (official file)	0.90	730.50	200	3.653	15,924.94	16,159.00	No
SOREL (official file)	0.95	730.50	200	3.653	15,911.70	16,142.10	No
SOREL (same pipeline)	0.80	100.65	200	0.503	0.00	0.00	Yes
SOREL (same pipeline)	0.90	100.65	200	0.503	0.00	0.00	Yes
SOREL (same pipeline)	0.95	100.65	200	0.503	0.00	0.00	Yes

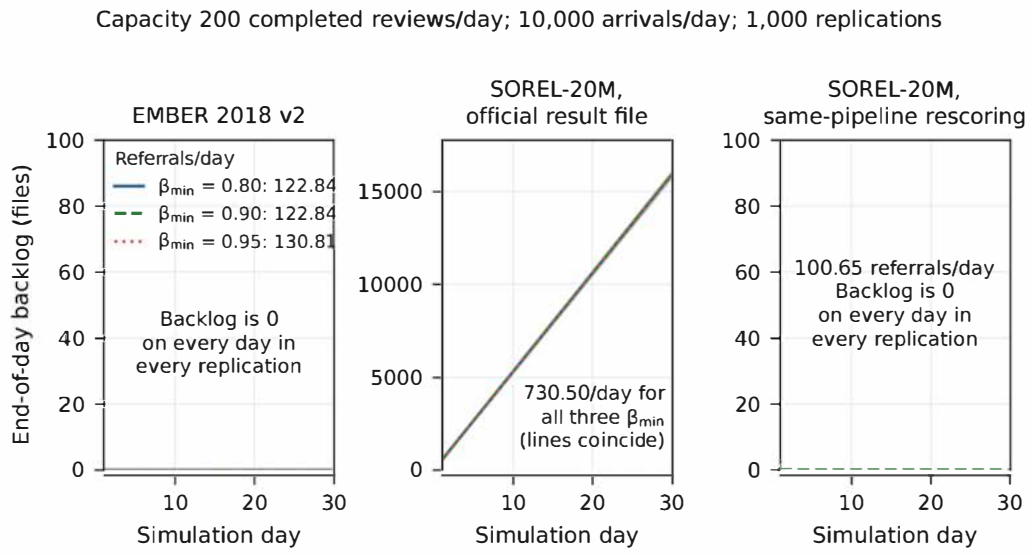


Figure 4. Mean end-of-day review backlog and simulated 5th–95th percentile range over 30 days for the primary Policy Cβ profiles: EMBER, SOREL with the official test-result file, and SOREL with the same-pipeline rescoring. Capacity is 200 completed reviews per day; the calculation is a scenario illustration, not a staffing forecast.

5.17. Results from the Time-Ordered BODMAS Dataset

Each model produces 576 status rows: 192 profiles at each of three βmin values. Structural conflict under $\pi\beta\min > r\max$ is identical at 64 profiles in every model–β cell. After that arithmetic screen, score-dependent infeasibility is zero for LightGBM and ExtraTrees and 104 per β for LinearSGD. Thus, structural infeasibility is model-independent, whereas the additional score/FPR conflict depends on score quality. Table 22 summarizes the validation and frozen-test outcomes for all three reference models.

Table 22. BODMAS validation feasibility and frozen-test retention at βmin = 0.80/0.90/0.95.

Model	Validation Feasible (0.80/0.90/0.95)	Structural	Score-Dependent	Test Pass	Test Fail	Primary Profile
LightGBM	128/128/128	64/64/64	0/0/0	100/100/100	28/28/28	Pass/Pass/Pass
ExtraTrees	128/128/128	64/64/64	0/0/0	116/100/28	12/28/100	Pass/Pass/Pass
LinearSGD	24/24/24	64/64/64	104/104/104	24/24/24	0/0/0	Infeasible/Infeasible/Infeasible

Among validation-selected profiles, frozen-test failures are 28 per β for LightGBM and 12/28/100 for ExtraTrees. LinearSGD selects only 24 profiles per β and all 24 retain the test constraints, but its illustrative primary profile is score-dependent infeasible at all three β values. The LightGBM and ExtraTrees primary profiles pass at all three values. Program S14 independently matches all 1728 statuses, 840 thresholds, and every metric or failure class. Figure 5 compares validation feasibility and test retention under point-estimate selection and the exploratory evidence gate.

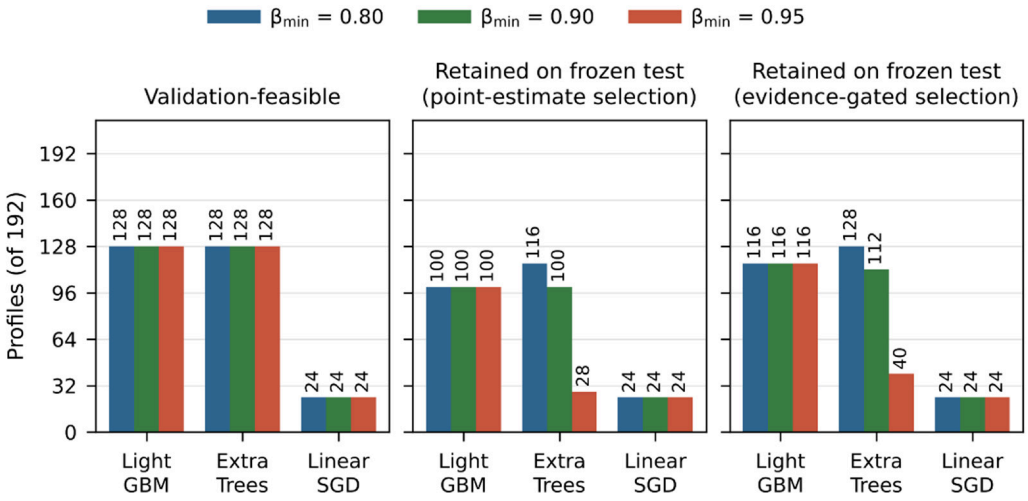


Figure 5. Policy Cβ validation feasibility and frozen-test constraint retention across three BODMAS models, under point-estimate selection and under the exploratory evidence gate (Section 5.13). Sources: File S44_bodmas_multimodel_feasibility_summary.csv and File S61.

This extension is a robustness check on a public cleaned derivative and static EMBER-style features, not a live supplier stream or organizational field validation. Its cross-model results broaden the observed scope of the policy-audit phenomena but do not establish a universal ranking of operational detector performance.

6. Operational C-SCRM Integration

The threshold belongs inside an evidence workflow. At intake, the gate records artifact identity, provenance, model/version, and score. Referral triggers additional investigation, but enforcement remains a governed decision based on independent evidence. Monitoring records confirmed outcomes, workload, drift, and recurring supplier signals. Table 23 directly links these actions to current NIST planning, due-diligence, and CI/CD integration guidance [49–51].

Table 23. Mapping from the score decision to C-SCRM operating and audit procedures.

Stage	Inputs	Automated Policy Output	Human/Governance Action	Required Audit Record	NIST Linkage
Intake and due diligence	Hash, signature, supplier, SBOM, provenance, model score	Approval candidate or review referral	Verify identity, supplier context, and routing	Artifact ID, model/version, τ_R, rule, timestamp	SP 1326: supplier/product due-diligence evidence [49]
Planning and authorization	Validation package, π, costs, α, r_{max}, β_{min}	Versioned threshold proposal or infeasible state	Authorize parameters, capacity, exceptions, and review cadence	Approver, assumptions, source hashes, selection_status	SP 800-18 Rev. 2: system C-SCRM plan and traceable responsibilities [50]
CI/CD investigation	Referred artifact and independent signals	Ranked review evidence	Sandbox, signature-chain, SBOM, provenance, and analyst disposition	Evidence sources, outcome, analyst, elapsed time	SP 800-204D: integrate supply-chain controls into CI/CD [51]
Release	Investigation result and business context	No automatic enforcement from score alone	Approve, delay, quarantine, or reject under policy	Decision authority, rationale, linked evidence	SP 800-18 Rev. 2 authorization and plan maintenance [50]
Monitoring and reassessment	Confirmed outcomes and workload	Drift, FPR, capacity, and backlog indicators	Revalidate assumptions, supplier controls, or capacity	FPR, referral rate, misses, backlog, supplier trend	SP 1326 reassessment and SP 800-18 plan updates [49,50]

This separation prevents two category errors: treating a ranking score as a calibrated probability and treating a referral as proof of malware. A threshold record must identify validation data, model hash, π, costs, α, r_{max}, β_{min}, policy, code version, score-generation path, selection status, and approval authority. Table 24 lists the permitted governance response for each recorded state. Each response is an authorized, versioned decision and is distinct from automatically relaxing a constraint or changing the threshold after testing.

Operationally, the evidence sources in Table 23 can be applied as a staged triage workflow. A referred artifact may be held for prioritized sandbox, multi-engine, signature-chain, hash, SBOM, provenance, and analyst review; release, delay, quarantine, or rejection remains a separately authorized downstream decision. No single supporting signal is treated as conclusive, and the score itself remains a routing input rather than a malware verdict.

Table 24. Governance responses to unverified score provenance, infeasibility, and frozen-test failure.

Recorded State	Evidence	Permitted Responses (Authorized, Versioned)	Not Permitted
Unverified score-source comparability	No verified file-level alignment; model, feature, inference settings, and labels not jointly checked	Stop; verify identifiers and versioned settings, compare matched scores, and repeat selection and frozen testing if the pipeline changes	Attributing an unmatched result to drift, a pipeline defect, or staffing need
Structural infeasibility ($\pi_U\beta_{min} > r_L$)	Arithmetic precheck before scores are read	Increase sustainable capacity; add an upstream filter; narrow the scope; formally accept a lower β_{min}	Searching for a threshold; using the no-referral boundary as a fallback
Score-dependent infeasibility or insufficient evidence	Empty eligible set; frontier $r_{min}(\pi)$ (Figure 3)	Evaluate another detector or recalibrate; collect more representative validation data; revisit capacity or β_{min} through risk acceptance	Silently relaxing α, r_{max}, or β_{min}; selecting the nearest violating threshold

Table 24. *Cont.*

Recorded State	Evidence	Permitted Responses (Authorized, Versioned)	Not Permitted
Not robustly feasible over input uncertainty	Endpoint robust-feasibility condition fails (Proposition 3)	Narrow the uncertainty with adjudicated outcomes (Section 4.4.4); stage deployment under a narrower scope	Approving at the nominal point while the stated range includes infeasible values
Frozen-test failure	Negative margin m_{α}, m_{r}, or m_{β} on held-out data	Preserve the failure; diagnose the failed limit; new validation sample and new prospective holdout; consider the exploratory evidence gate or a margin	Retuning τ_R on test data; replacing the policy after seeing test results
Field alarm after approval	FPR, backlog or SLA, audited miss, or drift indicators (Section 7.5)	Suspend or re-estimate inputs; repeat Algorithm 1 under a new version	Feeding production outcomes back into the current version's threshold

7. Discussion

7.1. Main Finding

Adding review capacity to FPR-constrained cost minimization changed the validation-selected threshold in 176 of 384 registered profiles. The comparison shows that controlling FPR alone did not control total review workload in those settings. The constraints may already be satisfied in other profiles, so adding them need not change a threshold. Structural infeasibility reflects contradictory capacity and recall requirements; score-dependent infeasibility reflects the absence of a qualifying threshold in the observed validation scores. Neither should be conflated with failure of an already selected, frozen threshold on test data.

Satisfying the constraints on validation data and retaining them after the threshold is frozen and applied to test data are different properties, and both should be recorded in the approval log. The evidence for this distinction does not rest on the official SOREL file: on EMBER, all limits were retained in only 28 of 112 validation-feasible profiles at $\beta_{\min} = 0.90$; on same-pipeline SOREL scores, the $\beta_{\min} = 0.95$ floor failed in 108 of 128; and on BODMAS, 28 LightGBM profiles per $\beta_{\min}$ failed on test. The framework therefore does not guarantee that an acceptable threshold exists or that a validation-feasible threshold will retain its limits on unseen data; it records observed infeasibility and constraint violations in the analyzed data. It cannot guarantee detection of unadjudicated production failures.

7.2. SOREL Frozen-Test Failure and Score Provenance

The SOREL result shows why approved conditions must be audited after freezing and why the audit must include the score-generation path. On the prespecified official result file, the frozen representative threshold kept recall (0.94082) but violated the FPR and capacity ceilings, and no validation-feasible profile retained all limits at $\beta_{\min} = 0.90$. The procedure recorded this failure without retuning. The score-source comparison then showed that the test-feature archive, rescored by the validation pipeline, retained all limits in 112 of 128 profiles and gave an FPR comparable to the validation FPR at the representative threshold (Section 5.4). This sensitivity analysis shows that the frozen-test conclusion depends on the score source; it does not validate the official-file result or show which source reflects the detector's behavior.

The response to such a failure is therefore ordered. The first step is to establish file-level identities, labels, and the versioned model, feature extraction, preprocessing, and prediction settings for both sources. If these checks cannot establish comparability, the cause remains unresolved; only matched and reproducibly generated scores can support an

interpretation about the detector or target population. Had the official-file result been read as temporal drift, it would have suggested a 3.65-fold capacity increase or a lower recall floor that the same-pipeline scores do not support. Had the favorable same-pipeline source simply replaced the prespecified one, an unfavorable result would have been removed after the fact. Reporting both, with their provenance, avoids both errors.

More generally, the statement that no threshold satisfies the security and workload constraints simultaneously can arise from four distinct causes—an arithmetic conflict, the detector's score distribution, a change between validation and deployment, or a mismatch in score generation—and each requires a different governance response (Table 24). None is resolved by relaxing a constraint or moving the threshold after testing.

7.3. Discrimination, Calibration, and Operational Feasibility

Three properties must be distinguished. Discrimination (ROC-AUC, average precision) concerns ranking. Calibration concerns agreement between scores and conditional event frequencies; ECE is a binned diagnostic, while Brier score measures overall probabilistic accuracy and also reflects discrimination and prevalence. Operational feasibility concerns whether a frozen threshold meets its approved FPR, recall, and workload limits. Strong ranking alone guarantees neither calibration nor feasibility. Conversely, a constant prevalence predictor can be calibrated without useful discrimination, and loose constraints can be feasible even for a weak score. The official SOREL file had ROC-AUC 0.98128 yet an FPR at the frozen threshold 77 times its validation value; EMBER kept ROC-AUC above 0.995 on both partitions yet lost 0.023–0.088 recall at fixed thresholds.

The SOREL Brier/ECE deterioration in Section 5.1 is descriptive evidence about the official file's scores as probabilities, not a causal explanation of the constraint violations; the score sources yield different outcomes, but the cause is unresolved (Section 5.4). The Bayes reference threshold illustrates why the common 0.5 threshold need not be appropriate, but its direct use requires calibrated, prior-adjusted probabilities that the public scores do not guarantee. It therefore remains a comparator, and the operational threshold is selected on representative validation data and assessed on separate test data.

7.4. Scope of the Conclusions and Conditions for Transfer

Table 25 separates conclusions that rest on the complete registered grid from those that depend on the representative scenario. Statements about structural and score-dependent infeasibility, the effect of adding capacity, the condition under which the recall floor binds, and frozen-test retention counts use every registered profile. Costs, referral counts, paired differences, and backlogs are reported only for the representative profile and are worked examples. Three kinds of evidence are kept distinct throughout: benchmark experiments on public data (Sections 5.1–5.15 and 5.17), scenario-based workload illustrations (representative-profile costs and referral counts and the backlog in Section 5.16), and prospective organizational validation, which this study does not perform (Section 8).

Every prevalence-adjusted count in this paper—expected referrals, benign referrals, missed malicious files, expected cost, and backlog—assumes that the class-conditional score distributions measured on the benchmark at the frozen threshold carry over to the target gate while only the malicious-artifact rate changes (Proposition 4). Benchmarks do not establish this. Changes in supplier populations, artifact types such as installers, drivers, or signed updates, collection procedures, labeling rules, time periods, and score-generation paths can each alter F_0 and F_1, not only π. Transfer is defensible only when (i) the target stream's artifact types and collection procedure are represented in the validation data, (ii) the same versioned scoring pipeline is used, (iii) validation data come from the same gate and a recent window, (iv) outcomes are adjudicated with partial-verification correction

(Section 4.4.4), and (v) monitored FPR, referral rate, and audited recall remain within the approved bounds. Until such data exist, the results are evidence about the behavior of the decision procedure, not forecasts of supplier-stream performance.

Table 25. Evidence base of the main conclusions.

Conclusion	Evidence Base	Depends on Representative Values?
Arithmetic conflicts recur across datasets and detectors (64 per dataset–$\beta_{\min}$ cell)	All 192 profiles × 3 floors; EMBER, SOREL, BODMAS	No; deterministic property of the grid
Adding capacity changes the FPR-only threshold in 176 of 384 profiles	Full grid; EMBER and SOREL	No; a count of grid coverage
The recall floor changes Policy C only when C's validation recall is below $\beta_{\min}$; binding floors failed on EMBER test data	720 validation-feasible profiles	No
Validation feasibility does not imply frozen-test retention	Full grid; EMBER, SOREL (both sources), BODMAS	No
The official-file SOREL failure depends on the test-score source	1920 frozen SOREL thresholds	No
Feasibility frontier and robust-feasibility rule	Exact for each dataset and $\beta_{\min}$	No; evaluated for any (π, $r_{\max}$)
Expected costs, referrals per 10,000, paired differences, and backlog	Representative profile only	Yes; worked examples
Effect of the exploratory evidence gate on retention	BODMAS full re-selection; EMBER/SOREL gate check	Partly; depends on validation sample sizes; pointwise bounds without post-selection coverage

7.5. Organizational Adoption Procedure

Before threshold selection, an organization should approve the expected malicious-artifact rate π, the dimensionless false-negative-to-false-positive loss-weight ratio, the FPR ceiling α, the maximum additional-review rate rmax, and the minimum recall βmin. The malicious-artifact rate should come from adjudicated outcomes for a defined gate and period. Review capacity should be based on completed cases and service-time variability rather than the number of cases the team can merely accept.

Structural infeasibility, score-dependent infeasibility, or a frozen-test constraint violation must not be erased by adjusting the threshold after observing the test result. The failed frozen-test result should be preserved, and a documented remediation process should follow.

First, confirm that the validation and deployment scores came from the same versioned model, feature extraction, preprocessing, and scoring code (Algorithm 1, step 0). Then construct a new validation sample that better represents the intended operating environment. When necessary, the model scores may be recalibrated, the upstream detector retrained, or an additional screening step introduced. A revised threshold is then selected and frozen on the new validation sample. Finally, the frozen threshold is evaluated once on a new prospective holdout that was not used during selection.

If the joint constraints remain infeasible, the policy should not be deployed. Review capacity may be expanded, a pre-filter may be added, or the deployment scope or security requirements may be formally changed before revalidation. Expanding capacity or relaxing βmin, α, or rmax is not a statistical correction; it is a governance and risk-acceptance decision that requires authorization.

Approval proceeds in order: (1) lock the gate, population, adjudication rule, and window; (2) estimate π and $K_{\text{eff}}/N_{\text{op}}$ within that same scope, correcting for partial verification (Section 4.4.4); (3) approve α from the benign-referral budget and $\beta_{\min}$ from miss consequences and risk tier; (4) precheck $\pi_U \beta_{\min} \leq r_L$ and robust feasibility at the worst-case corner (Proposition 3); (5) verify the score-generation path; and (6) select only on validation data and version-lock the threshold with the model, score, code, and input hashes. Values with different periods or denominators must not be combined. The approval record names the estimator, data owner, uncertainty method, exceptions, approver, and expiry date.

Every frozen test decision stores point margins $m\alpha = \alpha - \text{FPR}$, $mr = r_{\max} - R_\pi$, and $m\beta = \text{TPR} - \beta_{\min}$; a negative margin is retained as a violation. Conservative deployment margins are $m\alpha,\text{CI} = \alpha - U95(\text{FPR})$, $mr,\text{CI} = r_L - U95(R_\pi)$, and $m\beta,\text{CI} = L95(\text{TPR}) - \beta_{\min}$. Validation feasibility and frozen-test constraint retention remain separate fields, and a test violation is never post hoc replaced by another threshold. The CI margins are meaningful only under the prespecified uncertainty design in Section 4.6; nominal pointwise bounds after threshold search are not an operational 95% certificate.

Re-estimation is triggered by approval-window expiry, π leaving its approved band, an FPR or backlog/SLA alarm, a high-consequence malicious miss, a change in traffic, staffing, or service time, or any change in the model, checkpoint, features, preprocessing, or score-generation path. The organization then estimates π, $r_{\max}$, α, and $\beta_{\min}$ from new evidence, approves a new version, and repeats validation and unused-test evaluation. Production or prior test outcomes are not fed back to retune the current version. File S52 locks the stages and required audit fields.

8. Limitations

This study analyzes how fixed public PE malware scores can be converted into review referrals. It neither develops a new malware detector nor demonstrates malware detection performance in an operational supplier stream.

The EMBER month holdout and alternative split experiments reuse the same 200,000 records in different partitions and therefore do not constitute validation on an independently collected dataset. SOREL is also a public benchmark rather than a record of actual supply-chain incidents or an organization's operating logs. External field validation will require independently sampled later data or an adjudicated acquisition or build stream in which benign and malicious outcomes have been established by qualified reviewers.

Malicious-artifact-rate reweighting assumes that the class-conditional TPR and FPR measured on the public benchmarks remain similar in supplier data (Section 7.4); this assumption was not tested on live supplier files. Neither the benchmark class ratio nor the representative condition is an organizational estimate, and costs, referral counts, and unfinished-review counts under the representative condition are worked examples only (Table 25). Before deployment, the operational-parameter protocol in Supplementary output File S34 requires recalibration from adjudicated organizational outcomes, loss records, staffing, service times, and workflow.

The review-backlog calculation simplifies real review work. It omits variation in the time needed for each file, urgent-case priority, shifts, holidays, absences, and rework. Actual waiting time and analyst requirements may therefore differ from the calculated values.

Wilson intervals and paired-bootstrap intervals describe sampling variation within the analyzed data, and McNemar's test compares the files on which two policies make different correct-or-incorrect decisions. These methods do not account for uncertainty in π, loss

weights, review capacity, long-term distribution change, or analyst behavior, and they do not establish operational value after deployment.

The framework ends at referral. A referral alone must not authorize automatic quarantine, rejection, blocking, or supplier sanctions; those actions require separately approved authority, procedures, and evidence. Reproducibility also depends on the continued availability and licensing terms of the EMBER and SOREL public data that are not redistributed with the supplement.

The BODMAS extension adds an independent public corpus and three score models but remains limited to a public cleaned derivative, static features, and a historical interval. It does not validate organization-specific supplier streams, interactions with signatures, SBOMs, or dynamic analysis, or real analyst service times. The BODMAS models are fit in memory: reproducible settings, scores, and audit outputs are provided, but no serialized checkpoint is claimed.

The SOREL score-source discrepancy is observed but not causally resolved: file-level alignment and the official file's producing settings have not been verified, and overlap involving 21 validation records remains a hypothesis rather than a measured upper bound (Section 4.3). The evidence-gate re-selection could be computed in full only for BODMAS, because raw EMBER and SOREL scores are not redistributed; for those benchmarks the gate was applied to the point-selected thresholds. In addition, Files S60 and S61 use pointwise nominal bounds after validation selection, with an approximate referral-rate bound; these results do not establish simultaneous or post-selection 95% coverage. Organizational approval needs the independent-sample or selection-aware uncertainty procedure described in Section 4.6.

The next major step is prospective validation on independently collected, adjudicated supplier-intake or build-pipeline data with partial-verification correction (Section 4.4.4). Such validation is necessary to establish whether the framework can support operational decisions under realistic prevalence, workload, and distributional conditions.

9. Reproducibility and Data Availability

The supplement provides the analysis programs and outputs needed to audit and reproduce the reported results, including EMBER/SOREL/BODMAS figures, temporal-assignment checks, same-file statistical comparisons, operating-parameter records, model/score provenance, the SOREL score-source comparison (Files S54–S56), the additional post hoc analyses (Files S57–S64), recomputation logs, environment specifications, a manifest, and SHA-256 integrity values. Programs S21 and S22 check the manuscript against the supplementary outputs. The reproducibility inventory File S65 maps every program and output to its inputs, its use in the manuscript, and its reproducibility class.

To reproduce the results, researchers should verify the recorded versions and hashes, obtain the required EMBER and SOREL data from their official distributors, and run the supplied programs in the documented order. The supplementary audit scripts check agreement between the manuscript's principal numerical claims and the generated outputs.

Results therefore fall into two reproducibility classes, recorded output by output in File S65. Class A outputs can be regenerated from the supplied files alone; they include all R1 analyses computed from distributed outputs (Program S20; Tables 12, 17, 18 and 20), the SOREL side-by-side summary behind Table 11, the manuscript figures, the BODMAS independent recomputation and exploratory evidence-gate re-selection from the distributed frozen scores (File S42), and all provenance and integrity audits. Class B outputs require separately obtained benchmark files: the EMBER and SOREL policy experiment of Program S9 (for example, Tables 8–10, 13–16 and 19 and the backlog simulations) requires the

official EMBER and SOREL files; the same-pipeline rescoring of Program S19 requires the SOREL test-feature archive; and the BODMAS model fitting of Program S13 requires the BODMAS Cleaned files. Class B outputs are distributed and can be audited against the manuscript without these files, but regenerating them requires the files.

The supplementary package does not redistribute malware executables, raw EMBER or SOREL score files, feature archives, caches, internal samples, or the approximately 23.8 GB SOREL validation archive. These inputs must be obtained separately from their official sources because of licensing or file-size constraints. BODMAS frozen scores are distributed as File S42 for independent re-selection. Distributed logs remove personal workstation paths while retaining the information required to audit each run.

10. Conclusions

This study proposed a reproducible decision-governance procedure that converts fixed PE malware scores into review referrals at software supply-chain gates. It takes the malicious-artifact rate, loss weights, an FPR ceiling, review capacity, and minimum recall as organization-approved inputs; selects thresholds only on validation data; freezes them before testing; and records versioned links among the model, scores, data, and approval conditions. The threshold-selection principle itself is established; the contribution is the joint three-limit formulation, an elementary analytical feasibility characterization, and their integration into an auditable governance procedure, not a generally deployable supply-chain review solution: feasibility and performance must be established for each target environment.

Four findings hold across the registered grid. First, infeasibility has distinct causes: 384 of 1152 Cβ profiles were arithmetically infeasible because capacity could not accommodate the recall floor, and 48 were infeasible because the validation scores supplied no qualifying threshold; feasibility depends on π and $r_{\max}$ through an exact linear frontier, which also yields a robust-feasibility rule for stated input ranges, conditional on the observed validation scores and not a guarantee of future performance. Second, the recall floor changes Policy C's decision only when Policy C's validation recall falls below it, and in the analyzed profiles the selected threshold lay near the floor; on EMBER, every such threshold failed the floor on test. Third, validation feasibility does not guarantee frozen-test retention, as EMBER, BODMAS, and both SOREL test-score sources show. Fourth, SOREL frozen-test retention at $\beta_{\min} = 0.90$ differed by score source (0 of 128 on the prespecified official test-result file versus 112 of 128 in the same-pipeline rescoring sensitivity analysis); file-level alignment and the cause are unresolved, so verifying both sample and score provenance is a prerequisite for transferring any threshold.

The framework does not guarantee that an acceptable threshold exists or that an approved threshold will retain its limits on unseen data. It records infeasibility and constraint violations observed in the analyzed data, attributes a cause only where evidence permits, and preserves failed results without post hoc adjustment; this is not a guarantee of detecting unadjudicated production failures. Costs, referral counts, and backlogs are scenario-based illustrations. The next major step is prospective validation on independently collected, adjudicated supplier or build-pipeline data, which is required before any threshold receives versioned authorization for operational use.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app162110682/s1, The English-named supplementary package contains Programs S7–S22, outputs S0–S65, manuscript figures, integrity and regression evidence, recomputation logs, environment specifications, README, a file inventory (manifest), and SHA256SUMS.txt, which lists SHA-256 file-integrity values. Files S49 and S50 provide the model/score provenance, including the verification status of each score-generation path, and 22 checks;

Files S54–S56 compare the two SOREL test-score sources; Files S57–S64 contain the additional post hoc analyses (Cβ decision classes, input sensitivity, feasibility frontier, exploratory evidence gate, SOREL score-source audit, and backlog comparison). Program S10 audits the supplementary package; Programs S21 and S22 audit consistency between the manuscript and the supplement, and Program S22 writes the reproducibility inventory File S65. Raw EMBER/SOREL scores, feature archives, caches, and malware binaries are not included; BODMAS frozen scores are included as File S42. These S-numbers identify files in the supplementary package rather than separately numbered manuscript tables; File S65 provides the file inventory, input requirements, manuscript links, and reproducibility classes for all outputs.

Author Contributions: Conceptualization, C.K. and Y.R.; methodology, C.K.; software, C.K.; validation, C.K.; formal analysis, C.K.; investigation, Y.R.; data curation, C.K.; writing—original draft preparation, C.K.; writing—review and editing, C.K., Y.R. and S.S.; visualization, C.K.; supervision, Y.R. and S.S.; project administration, Y.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable. This study analyzes malware-detection scores and public cybersecurity benchmark data and does not involve humans or animals.

Informed Consent Statement: Not applicable.

Data Availability Statement: Code, aggregate policy results and review-backlog results, BODMAS frozen scores, model/score provenance Files S49 and S50, and reference-number audit File S51 are supplied as Supplementary Material. Public EMBER and SOREL features and scores must be obtained from their original distributors under the applicable licenses. The SOREL validation URL, byte count, SHA-256, generation command, expected CSV identity, and official test-score path are documented in the Supplementary README and score-provenance record. Raw executables and large or license-restricted inputs are not redistributed. BODMAS Cleaned v1.0 was accessed on 8 September 2026 at https://huggingface.co/datasets/it4lia/BODMAS_cleaned. The input identities are bodmas_clean.npz (15f46e891f27f3661ccd02fab9ebe504b8c6e1c218a47039462753c6ffc07a94), bodmas_clean_X.npy (9d5f3f91a01745c5ee55c747a40dc061f4ec7e0152b9d28ab92ebd008be216d6), bodmas_clean_y.npy (d3bb7abef44aebdaad44d83919747d7c9afe905b1e8557dfcf070b9a1eda9e30), and bodmas_clean_metadata.parquet (24539194c92bd76d02a4a026982dd6f78cab870a95a338567c89434184f00665) [48]. File S0_source_provenance.csv records the corresponding direct per-file URLs and byte counts. The joint approval and refresh protocol for the operating variables is supplied as File S52. The SOREL score-source comparison (Files S54–S56) and the additional post hoc analyses (Files S57–S64) are supplied with the programs that generate them. File S65 lists, for every program and output, its inputs, its use in the manuscript, and whether it can be regenerated from the supplied files alone or requires the separately obtained EMBER, SOREL, or BODMAS files.

Acknowledgments: During the preparation of this manuscript, the authors used Anthropic Claude and OpenAI ChatGPT/Codex for the purposes described in Section 4.10. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Boyens, J.; Smith, A.; Bartol, N.; Winkler, K.; Holbrook, A.; Fallon, M. *Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations*; NIST SP 800-161 Rev. 1 Update 1; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2024.
2. Joint Task Force. *Security and Privacy Controls for Information Systems and Organizations*; NIST SP 800-53 Rev. 5; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2020.
3. National Institute of Standards and Technology. *Secure Software Development Framework (SSDF) Version 1.1*; NIST SP 800-218; NIST: Gaithersburg, MD, USA, 2022.
4. National Telecommunications and Information Administration. *The Minimum Elements for a Software Bill of Materials (SBOM)*; NTIA: Washington, DC, USA, 2021.

5. Anderson, H.S.; Roth, P. EMBER: An open dataset for training static PE malware machine learning models. *arXiv* **2018**, arXiv:1804.04637.
6. Harang, R.; Rudd, E.M. SOREL-20M: A large scale benchmark dataset for malicious PE detection. *arXiv* **2020**, arXiv:2012.07634.
7. Sophos AI. SOREL-20M: Sophos-ReversingLabs 20 Million Dataset. GitHub Repository. 2020. Available online: https://github.com/sophos/SOREL-20M (accessed on 13 October 2026).
8. Axelsson, S. The base-rate fallacy and the difficulty of intrusion detection. *ACM Trans. Inf. Syst. Secur.* **2000**, *3*, 186–205. [CrossRef]
9. Elkan, C. The foundations of cost-sensitive learning. In Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, WA, USA, 4–10 August 2001; pp. 973–978.
10. Drummond, C.; Holte, R.C. Cost curves: An improved method for visualizing classifier performance. *Mach. Learn.* **2006**, *65*, 95–130. [CrossRef]
11. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [CrossRef]
12. Kweon, C.; Shin, S.; Jeon, S. NIST SP 800-161-Aligned Security Control Design and Quantitative Evaluation of a Deep-Learning-Based Malware Detection Model for Supply-Chain Threats. *J. Def. Ind. Secur.* **2026**, *2*, 59–80. (In Korean) [CrossRef]
13. Kweon, C.; Shin, S.; Ryu, Y.; Kim, G. Policy-Oriented Utilization of Supply Chain Malware Detection Results under Cost and Operational Constraints: An Industrial Security Decision Framework Based on NIST SP 800-161. *Korean J. Ind. Secur.* **2026**, *16*, 9–28. (In Korean) [CrossRef]
14. Mozannar, H.; Sontag, D. Consistent Estimators for Learning to Defer to an Expert. In Proceedings of the 37th International Conference on Machine Learning, Virtual Event, 13–18 July 2020; PMLR 119, pp. 7076–7087.
15. Raff, E.; Barker, J.; Sylvester, J.; Brandon, R.; Catanzaro, B.; Nicholas, C. Malware detection by eating a whole EXE. In Proceedings of the AAAI Workshop on Artificial Intelligence for Cyber Security, New Orleans, LA, USA, 2–7 February 2018.
16. Nataraj, L.; Karthikeyan, S.; Jacob, G.; Manjunath, B.S. Malware images: Visualization and automatic classification. In Proceedings of the 8th International Symposium on Visualization for Cyber Security, Pittsburgh, PA, USA, 20 July 2011.
17. Saxe, J.; Berlin, K. Deep neural network based malware detection using two dimensional binary program features. In Proceedings of the 10th International Conference on Malicious and Unwanted Software, Fajardo, Puerto Rico, 20–22 October 2015.
18. Arp, D.; Spreitzenbarth, M.; Hübner, M.; Gascon, H.; Rieck, K. DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket. In Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, USA, 23–26 February 2014.
19. Pendlebury, F.; Pierazzi, F.; Jordaney, R.; Kinder, J.; Cavallaro, L. TESSERACT: Eliminating experimental bias in malware classification across space and time. In Proceedings of the 28th USENIX Security Symposium, Santa Clara, CA, USA, 14–16 August 2019; pp. 729–746.
20. Anderson, H.S.; Kharkar, A.; Filar, B.; Evans, D.; Roth, P. Learning to evade static PE machine learning malware models via reinforcement learning. *arXiv* **2018**, arXiv:1801.08917.
21. Grosse, K.; Papernot, N.; Manoharan, P.; Backes, M.; McDaniel, P. Adversarial Examples for Malware Detection. In *Computer Security—ESORICS 2017*; Foley, S.N., Gollmann, D., Snekkenes, E., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2017; Volume 10493, pp. 62–79. [CrossRef]
22. Kolosnjaji, B.; Demontis, A.; Biggio, B.; Maiorca, D.; Giacinto, G.; Eckert, C.; Roli, F. Adversarial malware binaries: Evading deep learning for malware detection in executables. In Proceedings of the 26th European Signal Processing Conference, Rome, Italy, 3–7 September 2018.
23. Pierazzi, F.; Pendlebury, F.; Cortellazzi, J.; Cavallaro, L. Intriguing properties of adversarial ML attacks in the problem space. In Proceedings of the IEEE Symposium on Security and Privacy, San Francisco, CA, USA, 18–20 May 2020; pp. 1332–1349. [CrossRef]
24. Cappos, J.; Samuel, J.; Baker, S.; Hartman, J.H. A look in the mirror: Attacks on package managers. In Proceedings of the 15th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 27–31 October 2008.
25. Samuel, J.; Mathewson, N.; Cappos, J.; Dingledine, R. Survivable key compromise in software update systems. In Proceedings of the 17th ACM Conference on Computer and Communications Security, Chicago, IL, USA, 4–8 October 2010.
26. Kuppusamy, T.K.; Diaz, V.; Cappos, J. Mercury: Bandwidth-effective prevention of rollback attacks against community repositories. In Proceedings of the USENIX Annual Technical Conference, Santa Clara, CA, USA, 12–14 July 2017.
27. Open Source Security Foundation. *Supply-Chain Levels for Software Artifacts (SLSA) Specification*, version 1.2; OpenSSF: San Francisco, CA, USA, 2025.
28. Fourne, M.; Wermke, D.; Enck, W.; Fahl, S.; Acar, Y. It is like flossing your teeth: On the importance and challenges of reproducible builds for software supply chain security. In Proceedings of the IEEE Symposium on Security and Privacy, San Francisco, CA, USA, 21–25 May 2023; pp. 1527–1544. [CrossRef]
29. Ladisa, P.; Plate, H.; Martinez, M.; Barais, O. SoK: Taxonomy of Attacks on Open-Source Software Supply Chains. In Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 21–25 May 2023; pp. 1509–1526. [CrossRef]

30. Ohm, M.; Plate, H.; Sykosch, A.; Meier, M. Backstabber's knife collection: A review of open source software supply chain attacks. In Proceedings of the DIMVA, Lisbon, Portugal, 24–26 June 2020.
31. Zahan, N.; Lin, E.; Tamanna, M.; Enck, W.; Williams, L. Software bills of materials are required. Are we there yet? *IEEE Secur. Priv.* **2023**, *21*, 82–88. [CrossRef]
32. Brier, G.W. Verification of forecasts expressed in terms of probability. *Mon. Weather Rev.* **1950**, *78*, 1–3.
33. Niculescu-Mizil, A.; Caruana, R. Predicting good probabilities with supervised learning. In Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany, 7–11 August 2005; pp. 625–632.
34. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K.Q. On calibration of modern neural networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1321–1330.
35. Madras, D.; Pitassi, T.; Zemel, R. Predict Responsibly: Improving Fairness and Accuracy by Learning to Defer. In *Advances in Neural Information Processing Systems 31*; Curran Associates: Red Hook, NY, USA, 2018; pp. 6147–6157.
36. Alves, J.V.; Leitao, D.; Jesus, S.; Sampaio, M.O.P.; Liebana, J.; Saleiro, P.; Figueiredo, M.A.T.; Bizarro, P. Cost-Sensitive Learning to Defer to Multiple Experts with Workload Constraints. *Trans. Mach. Learn. Res.* 2024. Available online: https://openreview.net/forum?id=TAvGZm2Rqb (accessed on 13 October 2026).
37. Geifman, Y.; El-Yaniv, R. Selective Classification for Deep Neural Networks. In *Advances in Neural Information Processing Systems 30*; Curran Associates: Red Hook, NY, USA, 2017; pp. 4878–4887.
38. Tong, X.; Feng, Y.; Li, J.J. Neyman-Pearson Classification Algorithms and NP Receiver Operating Characteristics. *Sci. Adv.* **2018**, *4*, eaao1659. [CrossRef] [PubMed]
39. Angelopoulos, A.N.; Bates, S.; Candès, E.J.; Jordan, M.I.; Lei, L. Learn Then Test: Calibrating Predictive Algorithms to Achieve Risk Control. *Ann. Appl. Stat.* **2025**, *19*, 1641–1662. [CrossRef]
40. Angelopoulos, A.N.; Bates, S.; Fisch, A.; Lei, L.; Schuster, T. Conformal Risk Control. In Proceedings of the 12th International Conference on Learning Representations (ICLR 2024), Vienna, Austria, 7–11 May 2024. Available online: https://openreview.net/forum?id=33XGfHLtZg (accessed on 25 September 2026).
41. Alahmadi, B.A.; Axon, L.; Martinovic, I. 99% False Positives: A Qualitative Study of SOC Analysts' Perspectives on Security Alarms. In Proceedings of the 31st USENIX Security Symposium, Boston, MA, USA, 10–12 August 2022; pp. 2783–2800.
42. Barbero, F.; Pendlebury, F.; Pierazzi, F.; Cavallaro, L. Transcending TRANSCEND: Revisiting Malware Classification in the Presence of Concept Drift. In Proceedings of the 2022 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 22–26 May 2022. [CrossRef]
43. Joyce, R.J.; Miller, G.; Roth, P.; Zak, R.; Zaresky-Williams, E.; Anderson, H.; Raff, E.; Holt, J. EMBER2024—A Benchmark Dataset for Holistic Evaluation of Malware Classifiers. In Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, Toronto, ON, Canada, 3–7 August 2025. [CrossRef]
44. Elastic. Elastic Malware Benchmark for Empowering Researchers (EMBER). GitHub Repository. Available online: https://github.com/elastic/ember (accessed on 13 October 2026).
45. Efron, B. Bootstrap Methods: Another Look at the Jackknife. *Ann. Stat.* **1979**, *7*, 1–26. [CrossRef]
46. Fagerland, M.W.; Lydersen, S.; Laake, P. The McNemar Test for Binary Matched-Pairs Data: Mid-p and Asymptotic Are Better than Exact Conditional. *BMC Med. Res. Methodol.* **2013**, *13*, 91. [CrossRef] [PubMed]
47. Yang, L.; Ciptadi, A.; Laziuk, I.; Ahmadzadeh, A.; Wang, G. BODMAS: An Open Dataset for Learning Based Temporal Analysis of PE Malware. In Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW), Virtual, 27 May 2021; pp. 78–84. [CrossRef]
48. IT4LIA AI Factory. *BODMAS Cleaned*, version 1.0; Hugging Face Dataset; IT4LIA AI Factory: Bologna, Italy, 2021. Available online: https://huggingface.co/datasets/it4lia/BODMAS_cleaned (accessed on 8 September 2026).
49. National Institute of Standards and Technology. *NIST Cybersecurity Supply Chain Risk Management: Due Diligence Assessment Quick-Start Guide*; NIST Special Publication 1326; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2026. [CrossRef]
50. Licata, J.; McWhite, R.; Calloway, L.; Gilbert, D.; Anderson, M.; Snyder, J.; Miller, J. *Developing Security, Privacy, and Cybersecurity Supply Chain Risk Management Plans for Systems*; NIST Special Publication 800-18 Revision 2; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2026. [CrossRef]
51. Chandramouli, R.; Kautz, F.; Torres-Arias, S. *Strategies for the Integration of Software Supply Chain Security in DevSecOps CI/CD Pipelines*; NIST Special Publication 800-204D; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2024. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.