

Attack-Type-Aware Dynamic Bayesian Defense: A Three-Level Non-Zero-Sum Game for Brute-Force and SQL Injection Attacks

Maqsood Hussain Wani

University of Kashmir

Burhan Ul Islam Khan

`burhankhan@um.edu.my`

University of Malaya

Miss Laiha Mat Kiah

University of Malaya

Dwi Sudarno Putra

State University of Padang

Amirrudin Kamsin

University of Malaya

Harprith Kaur

INTI International University

Mesith Chaimanee

Shinawatra University

Research Article

Keywords: Dynamic Bayesian cyber defense, non-zero-sum game, attack intensity, brute-force attack, SQL injection, adaptive decision support

Posted Date: October 6th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11092788/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Attack-Type-Aware Dynamic Bayesian Defense: A Three-Level Non-Zero-Sum Game for Brute-Force and SQL Injection Attacks

Abstract

Choosing cyber-defense intensity under uncertain, time-varying attacks requires balancing compromise risk against the cost and disruption of unnecessary intervention. The published three-level attacker–defender game keeps this trade-off interpretable but pools operationally different attacks into one payoff structure and replays a static equilibrium without updating the defender's beliefs. This paper proposes an attack-type-aware dynamic Bayesian defense (ATA-DBD) model that removes both limitations yet retains the three-level action scale. Its contributions are (i) a family-conditioned non-zero-sum stage game that assigns brute-force and SQL-injection attacks separate success matrices, losses, costs, and asymmetric penalties, and proves that a pooled game cannot preserve family-specific defense orderings; (ii) a five-state Bayesian belief process with a myopic posterior-utility defense rule that includes reconfiguration cost and is monotone under within-family escalation; and (iii) a fully declared dual-layer evaluation protocol that pairs a 10,000-cycle benchmark matched to the published fixed-level and mixed-strategy baselines with a held-out stress test conditioned on the public CSE-CIC-IDS2018 attack timetable. In the matched benchmark, the proposed policy attains defense-success rates of 99.15%, 98.59%, and 99.52% across three economic scenarios while retaining 80.40%, 84.96%, and 79.17% residual energy and limiting redundancy to 2.28%, 1.32%, and 3.93%. In the calendar-conditioned test, it achieves 84.48% expected containment, 0.948 mean defender utility, 1.07% benign activation, and 1.27% under-defense, with posterior macro-F1 of 0.939, and exceeds the outcome-matched reconstruction of the published mixed strategy in utility by 0.249 (95% confidence interval [0.245, 0.253]). The results are controlled-simulation evidence for attack-family-aware adaptive decision support rather than flow-level intrusion-detection accuracy.

Keywords: Dynamic Bayesian cyber defense; non-zero-sum game; attack intensity; brute-force attack; SQL injection; adaptive decision support

1 Introduction

Operational cyber defense requires a response decision before the attack state is known with certainty. Brute-force (BF) activity places sustained pressure on authentication services, whereas SQL injection (SQLi) targets application inputs and database-facing operations. An inadequate response increases compromise risk; an unnecessarily strong response consumes resources and can disrupt legitimate use (Alghawazi et al. 2022; Aslan et al. 2023; Kanta et al. 2024; Paul et al. 2024; Perwej et al. 2021; Wang et al. 2021). This allocation problem is strategic because attack intensity and defense intensity jointly determine both security loss and operating cost. Game-theoretic models expose those incentives and resource limits, provided that the action space remains interpretable and the attack representation is detailed enough for calibration and comparison (Do et al. 2017; Ho et al. 2022; Iqbal et al. 2019; Manshaei et al. 2013).

An intensity label is meaningful only in the context of its attack family. High-intensity brute-force activity is associated with repeated authentication attempts, request bursts, and account or IP reuse. SQL injection is expressed through hostile query structure, manipulated parameters, scripted probing, and database-facing exploitation. A pooled intensity variable would assign a common success and cost structure to mechanisms that respond differently to defense. The model therefore retains three ordinal action levels but specifies their success probabilities, losses, and operating costs separately for BF and SQLi (Alghawazi et al. 2022; Li et al. 2019; Paul et al. 2024; Wang et al. 2021).

The benchmark is the three-level non-zero-sum model of Adisa et al. (2025). It assigns no-, low-, and high-intensity actions to both players and compares its Level-M mixed strategy with fixed Level-0/1/2 defenses using residual energy, defense success rate, and defensive redundancy. The source paper discusses password guessing and SQL injection, reports 100,000 classified records from 275,700 records scraped from three websites, and evaluates attack–defense interaction over 10,000 cycles.

The compact Level-0/1/2 scale and legacy metrics remain useful for comparison. Three features, however, restrict the original framework's use as a sequential attack-family model: BF and SQLi share one payoff structure, attacker actions are sampled randomly across repeated trials, and the scraped data are not available for independent reuse. The present work retains the original action scale while adding family-conditioned payoffs, Bayesian state

estimation, and a fully declared simulation design (Adisa et al. 2025; Attiah et al. 2018; Leevy and Khoshgoftaar 2020; Zhang and Malacaria 2021).

These observations define four requirements for the extension. The stage game must be internally coherent, and its success, loss, and cost parameters must distinguish BF from SQLi. The defense rule must update from uncertain evidence rather than replay a scenario-level strategy. The evidence claim must also match the data used: Layer B combines the public attack timetable with synthetic observations and therefore evaluates the decision process under controlled uncertainty, not flow-level intrusion detection.

To address these gaps, the proposed mechanism combines the refined stage-game utility with a five-state Bayesian filter. Let $\mathcal{S} = \{B, BF_1, BF_2, SQLi_1, SQLi_2\}$ denote the operational state space and let $b_t(s)$ be the posterior probability of state s in window t . Given the preceding defense d_{t-1} , the selected myopic defense action is

$$d_t^* = \min * \arg \max_{d \in D} \left\{ \sum_{s \in \mathcal{S}} b_t(s) U_D(s, d) - \kappa |d - d_{t-1}| \right\}. \quad (1)$$

The corresponding prediction and correction stages are

$$\hat{b}_t(s) = \sum_{u \in \mathcal{S}} P(s|u) b_{t-1}(u), \quad b_t(s) = \frac{p(o_t|s) \hat{b}_t(s)}{\sum_{v \in \mathcal{S}} p(o_t|v) \hat{b}_t(v)}. \quad (2)$$

Equation (1) selects the lowest-index maximizer when exact ties occur. With $\kappa > 0$, it remains a myopic rule because it optimizes the current window only. Equation (2) supplies temporal adaptation through state prediction and posterior correction; Equation (28) later defines the cumulative expected-utility criterion used for evaluation.

Fig. 1 connects the five-state filter to family-conditioned utility evaluation and the myopic defense rule. Its input box distinguishes the type of evidence expected in a deployment from the Gaussian observations used in the controlled benchmark.

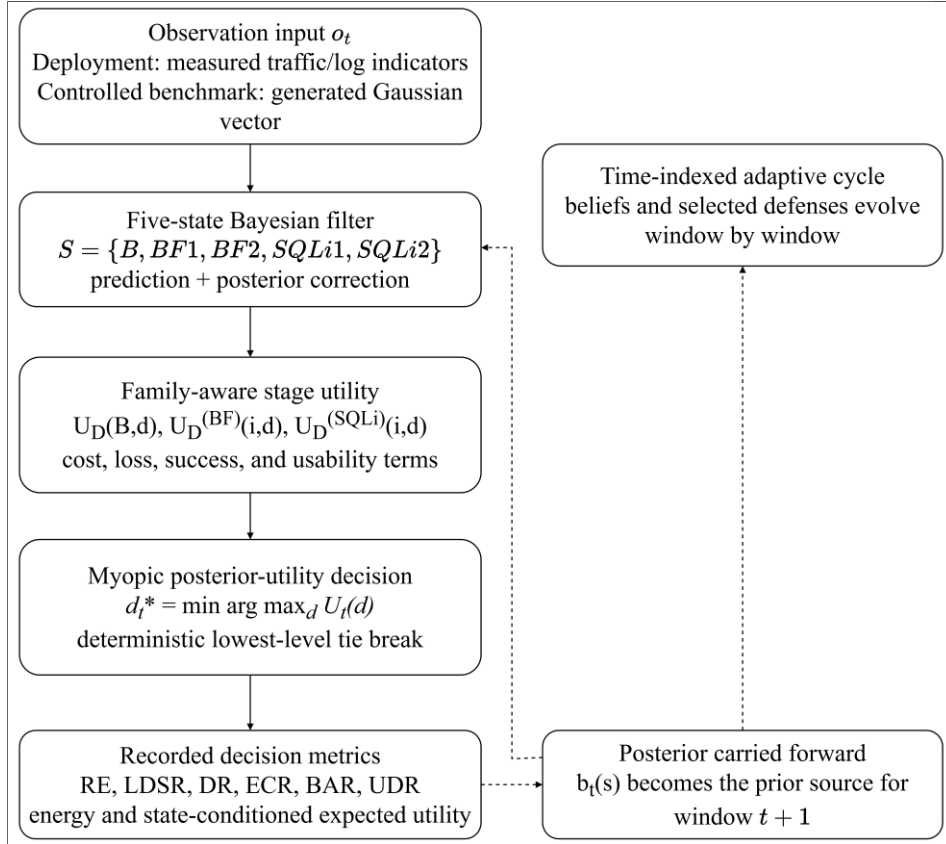


Fig. 1 Architecture of the attack-family-aware Bayesian defense cycle

The feedback path carries b_t into the prediction step for window $t + 1$. Beliefs and selected defense levels therefore evolve over time even though each action is chosen by a stage-wise maximization.

The resulting model is termed attack-type-aware dynamic Bayesian defense (ATA-DBD), and the paper makes three contributions.

(1) An attack-family-conditioned three-level non-zero-sum stage game. BF and SQLi receive separate success matrices, compromise values, losses, cost ladders, failed-attack penalties, and benign-use penalties within the same interpretable Level-0/1/2 action space (Section 3). Proposition 1 establishes equilibrium existence, Proposition 2 gives a marginal condition under which high-intensity defense is weakly suboptimal, and Proposition 3 shows that a pooled intensity-only payoff matrix cannot preserve both family-specific defense rankings. In the literature reviewed in Section 2, no three-level non-zero-sum cyber-defense game conditions its payoff semantics on the attack family.

(2) A dynamic Bayesian defense rule. The family-conditioned utilities are embedded in a five-state belief process over benign, low- and high-intensity BF, and low- and high-intensity SQLi states, and the defense level is selected by a myopic posterior-utility rule with an explicit reconfiguration cost (Section 4). Under the increasing-differences condition of Proposition 4, the rule is monotone under within-family escalation: stronger evidence that an attack is escalating never produces a weaker defense.

(3) A fully declared dual-layer evaluation protocol. Layer A reproduces the published 10,000-cycle benchmark with fixed Level-0/1/2 defenses, audits both transparent reconstructions of the published mixed strategy, and carries the outcome-matched reconstruction into the scenario-wise comparison, so that the comparison cannot rest on a single unexamined interpretation of a baseline whose data are not available. Layer B evaluates the decision mechanism against all comparators on a held-out calendar built from the public CSE-CIC-IDS2018 attack timetable under a declared stochastic observation model, with proper scoring rules, ablation, and sensitivity analyses (Sections 5 and 6). Every parameter, seed, and calendar is stated, so all results can be regenerated independently.

Section 2 positions the contribution relative to prior cyber-defense games. Sections 3 and 4 develop the stage game and Bayesian decision rule, Section 5 defines the two evaluation layers, Section 6 reports the results, and Section 7 concludes.

2 Related Work and Research Positioning

Prior work is compared along three dimensions that determine whether a model can support the proposed defense rule: payoff structure, information available to the defender, and attack-family resolution. These dimensions distinguish the roles of static games, sequential Bayesian models, and attack-specific evidence models.

2.1 Static and non-zero-sum attacker–defender games

Cybersecurity game models are commonly classified by payoff structure, timing, information, and number of players (Do et al. 2017; Iqbal et al. 2019; Manshaei et al. 2013). These choices determine whether a formulation can express resource trade-offs, sequential response, or uncertainty. Applications include advanced-manufacturing risk assessment, advanced persistent-threat defense, Bayesian incentives for distributed denial-of-service (DDoS) mitigation, and moving-target defense (Cho et al. 2020; Dahiya and Gupta 2021; Khalid et al. 2023; Sengupta et al. 2020; Zarreh et al. 2019). Static normal-form games make their action sets, assumptions, and payoffs explicit, which is useful when comparing a small number of costly defense options. Their one-stage structure does not, however, represent posterior revision as evidence arrives or tactical changes across monitoring windows (Attiah et al. 2018; Ho et al. 2022; Manshaei et al. 2013; Zhang and Malacaria 2021). Adisa et al. (2025) provide the compact benchmark used in this study: a simultaneous three-level non-zero-sum game evaluated against fixed defenses. Its action scale and resource metrics are retained for comparability; the extension adds family-specific payoffs and observation-conditioned defense selection.

2.2 Bayesian and sequential cyber-defense games

Harsanyi’s Bayesian framework represents unknown game attributes through a probability distribution over latent states. In cyber defense, capability, objective, and current intensity are inferred from indirect indicators such as login failures, repeated source activity, anomalous parameters, request timing, and other noisy traces (Harsanyi 1967; Liu et al. 2021; Zhang and Malacaria 2021). Because those evidence distributions may change, concept-drift research distinguishes drift detection, interpretation, and adaptation rather than assuming stationarity (Lu et al. 2019). Bayesian Stackelberg models combine attack estimation with online defensive optimization (Zhang and Malacaria 2021), while dynamic-defense studies emphasize adaptation in heterogeneous infrastructures (Zheng et al. 2022). Bayesian inference has also been used in state-transition, mission-impact, and deception settings (Lee et al. 2024; Pawlick et al. 2019; Yao et al. 2024). None of the studies reviewed here retains the compact three-

level non-zero-sum action scale, assigns separate BF and SQLi payoff semantics, and conditions defense selection on sequentially updated beliefs. Within this literature, the proposed model is the first to combine these three properties in one decision framework, and the ablation study in Section 6.4 quantifies the separate contributions of temporal belief propagation, family conditioning, and Bayesian updating.

Equation (3) states the zero-switching-cost sequence objective motivated by these studies; Equation (28) adds the reconfiguration term used in the evaluation.

$$J_D^\pi = \mathbb{E}^\pi \left[\sum_{t=1}^T U_D(s_t, d_t) \right] \quad (3)$$

Policy value therefore depends on state-conditioned actions across windows rather than on a single equilibrium profile (Dahiya and Gupta 2021; Harsanyi 1967; Zhang and Malacaria 2021; Zheng et al. 2022).

2.3 Attack-family-conditioned intensity and evidence models

Evidence and defensive response differ by attack family. BF activity is associated with repeated guesses, failed-login bursts, source reuse, and rate anomalies; SQL injection is associated with suspicious tokens, crafted parameters, and query manipulation (Alghawazi et al. 2022; Paul et al. 2024; Tang et al. 2020; Wang et al. 2021). The defensive implications also differ: throttling or step-up authentication can address low-rate BF activity, whereas SQL injection requires input validation and request filtering. Family conditioning therefore changes $q_{ad}^{(k)}$, the associated loss and cost parameters, and the ordering of defense levels (Chen et al. 2024; Kasim 2021; McWhirter et al. 2018; Sepczuk and Kotulski 2018; Xie et al. 2019; Zhang et al. 2022). Section 3 compares those rankings with an equal-weight pooled game. The only CSE-CIC-IDS2018 content used in Layer B is the published attack timetable, from which BF and SQLi windows are reconstructed. Packet, flow, and CICFlowMeter records are not analyzed; the experiment is therefore calendar-conditioned rather than a flow-level intrusion detection system (IDS) evaluation (Canadian Institute for Cybersecurity n.d.; Catillo et al. 2021; Leevy and Khoshgoftaar 2020; Zuech et al. 2021).

2.4 Comparative research gap and research questions

The preceding comparison reveals a specific unresolved combination. Static non-zero-sum games preserve interpretable resource trade-offs but do not condition actions on sequential evidence; Bayesian and sequential games adapt to uncertainty but do not retain the same three-level, family-conditioned benchmark structure; and attack-specific detection studies distinguish observable patterns without integrating those distinctions into a strategic payoff system. The present study addresses this intersection through family-specific stage games, a five-state myopic Bayesian filter, and direct comparison with fixed defenses and both transparent Level-M interpretations under matched and controlled simulations (Adisa et al. 2025; Wang et al. 2021; Zhang and Malacaria 2021; Zheng et al. 2022).

Table 1 summarizes this positioning.

Table 1 Positioning of the proposed model relative to the three literature strands compared in Section 2

Literature strand (representative studies)	Three-level ordinal intensity scale for both players	BF- and SQLi- specific payoff semantics	Sequential belief updating for defense selection	Strategic (game- theoretic) payoff model	Fully declared, publicly reproducible evaluation
Static three-level non-zero-sum game (Adisa et al. 2025)	Yes	No (pooled)	No	Yes	No (scraped records withheld)
Bayesian and sequential defense games (Dahiya and Gupta 2021; Lee et al. 2024; Liu et al. 2021; Yao et al. 2024; Zhang and Malacaria 2021; Zheng et al. 2022)	No	No	Yes	Yes	Varies by study

Literature strand (representative studies)	Three-level ordinal intensity scale for both players	BF- and SQLi- specific payoff semantics	Sequential belief updating for defense selection	Strategic (game- theoretic) payoff model	Fully declared, publicly reproducible evaluation
Attack-family-specific detection and evidence studies (Alghawazi et al. 2022; McWhirter et al. 2018; Paul et al. 2024; Tang et al. 2020; Wang et al. 2021)	Not applicable	Yes (observable signatures only)	No	No	Varies by study
Proposed (ATA-DBD)	Yes	Yes	Yes	Yes	Yes

The resulting evaluation is organized around four questions that separately test mathematical consistency, sequential adaptation, family conditioning, and benchmark performance:

- RQ1: Can a three-level non-zero-sum stage game be formulated whose cost, success, and penalty structure is internally consistent and conditioned on the attack family?
- RQ2: Does adding Bayesian belief updates improve defense decisions under attack uncertainty?
- RQ3: Does attack-family-aware modeling outperform a generic intensity-only model?
- RQ4: Does the proposed policy improve the protection–resource trade-off relative to fixed defenses and both transparent interpretations of the published Level-M benchmark?

RQ1 is addressed by the payoff and equilibrium analysis in Section 3. RQ2 is tested by the fixed-prior and no-update ablations, while RQ3 is assessed through the family-conditioned versus pooled-game analyses. RQ4 is evaluated through the matched Layer-A benchmark and the Layer-B controlled calendar simulation, including the formula-faithful and reported-outcome-matched Level-M comparators.

3 Refined Three-Level Non-Zero-Sum Game Formulation

This section formalizes a refined and attack-family-aware three-level stage game that remains directly comparable with the baseline of Adisa et al. (2025) while being analytically consistent and extensible. The formulation preserves the ordinal structure of no attack / low-intensity attack / high-intensity attack and no defense / low-intensity defense / high-intensity defense, but it does so with explicit family-specific payoff design so that brute-force and SQL injection attacks are no longer treated as operationally identical.

3.1 Notation and action space

Let the attacker’s action set and the defender’s action set be $A = D = \{0,1,2\}$, where 0 denotes inactivity or no escalation, 1 denotes a low-intensity action, and 2 denotes a high-intensity action. In operational terms, defense Level 1 denotes low-friction controls, such as request throttling and step-up authentication against BF or input validation and parameter filtering against SQLi, whereas Level 2 denotes high-friction controls, such as source lockout or blocking against BF or query blocking and strict request filtering against SQLi. Let the attack family be $k_t \in K = \{BF, SQLi\}$, where *BF* denotes brute-force/credential-guessing behavior and *SQLi* denotes SQL injection behavior. The subscript *t* indexes a decision window; for now, however, we focus on the single-period stage game and postpone the dynamic extension to Section 4.

For each attack family k , let $V_k > 0$ denote the attacker’s value from a successful compromise, $R_k > 0$ the defender’s protection reward for maintaining safe service, and $L_k > 0$ the defender’s loss if the attack succeeds. Let $c_{k,a}^A$ be the attacker’s direct operational cost at intensity level a , and let $c_{k,d}^D$ be the defender’s cost of deploying defense level d . We also include two non-symmetric penalty terms: $\eta_{k,a}$, the attacker’s failure or detection penalty when an attempted attack does not succeed, and $\gamma_{k,d}$, the defender’s usability or false-alarm penalty from activating defense level $d > 0$ when no attack is present. This asymmetry is central to the non-zero-sum formulation adopted here (Adisa et al. 2025; Nash 1951). The notation used throughout this paper is collected in Table 2.

Table 2 Notation for the stage game and Bayesian model

Symbol	Meaning	Domain / definition
a	Attacker pure action in the stage game	$a \in A = \{0, 1, 2\}$
d	Defender pure action in the stage game	$d \in D = \{0, 1, 2\}$
d_t	Defense deployed in window t	$d_t \in D$
k	Attack family	$k \in K = \{\text{BF}, \text{SQLi}\}$
s_t	Hidden operational state	$s_t \in \mathcal{S} = \{B, \text{BF}_1, \text{BF}_2, \text{SQLi}_1, \text{SQLi}_2\}$
$f(s)$	Attack-family map	$f(\text{BF}_i) = \text{BF}; f(\text{SQLi}_i) = \text{SQLi}$
$\iota(s)$	Intensity map	$\iota(B) = 0; \iota(k_i) = i$
$Q^{(k)}, q_{ad}^{(k)}$	Family-specific success matrix and entry	$Q^{(k)} \in [0, 1]^{3 \times 3}$
V_k, R_k, L_k	Compromise value, protection reward, defender loss	Positive normalized design parameters
$c_{k,a}^A, c_{k,d}^D$	Attack and defense costs	Level-0 costs equal zero
$\eta_{k,a}, \gamma_{k,d}$	Failed-attack and benign-use penalties	Nonnegative
$U_A^{(k)}, U_D^{(k)}$	Family-conditioned stage utilities	Equations (5)–(6)
$U_D(B, d)$	Common benign-state defender utility	Equation (6a)
x, y	Attacker and defender mixed strategies	$x \in \Delta(A), y \in \Delta(D)$
$b_t(s), \hat{b}_t(s)$	Posterior and predicted beliefs	Elements of $\Delta(\mathcal{S})$
o_t	Six-dimensional standardized observation	$o_t \in \mathbb{R}^6$
P_A, P_B	Layer-A and Layer-B transition matrices	Equations (39a) and (39)
$u_D(s)$	Defender utility vector over the three pure defense levels	$[U_D(s, 0), U_D(s, 1), U_D(s, 2)] \tau$
κ	Defense switching-cost coefficient	$\kappa = 0.015$ in primary simulations
J_D^π	Undiscounted cumulative expected utility under policy π	Equation (28); mean utility is the corresponding per-window sample average
$\epsilon(d)$	Energy expenditure	$\epsilon(0) = 0, \epsilon(1) = 0.003, \epsilon(2) = 0.006$
LDSR	Legacy ordinal defense-success rate	Base-paper comparison only
ECR	Expected containment rate	Uses $1 - q_{i(s),d}^{(f(s))}$
BAR, UDR	Benign activation and under-defense rates	Action-allocation metrics
$G_k^{d,d'}(x)$	Expected payoff gain of defense level d over d' under family k	Equation (15)
$\hat{\pi}$	Layer-A marginal intensity law	Equation (41)
C_j, n_j, k_j	j -th contiguous attack-family spell, its window count, and its family	Equations (33)–(34)
$\tau_j, \bar{\tau}_{\text{onset}}$	Attack-spell onset delay and its mean	Equation (51)
$\bar{m}, s, n$	Mean paired difference, its sample standard deviation, and the number of paired replications	Equation (52)

The central interaction term is the family-specific success-probability matrix, introduced in Equation (4):

$$Q^{(k)} = [q_{ad}^{(k)}]_{a,d \in \{0,1,2\}}, \quad q_{0d}^{(k)} = 0 \forall d \quad (4)$$

The attacker's stage utility for family k follows from this matrix according to Equation (5):

$$U_A^{(k)}(a, d) = q_{ad}^{(k)} V_k - c_{k,a}^A - \eta_{k,a} (1 - q_{ad}^{(k)}) 1_{\{a>0\}} \quad (5)$$

The corresponding defender stage utility is given by Equation (6):

$$U_D^{(k)}(a, d) = (1 - q_{ad}^{(k)}) R_k - q_{ad}^{(k)} L_k - c_{k,d}^D - \gamma_{k,d} 1_{\{a=0, d>0\}} \quad (6)$$

A benign window has no attack-family label. To prevent the merged benign state from being evaluated through an arbitrary family, define its utility by a fixed equal-weight average of the two no-attack rows:

$$U_D(B, d) = \frac{1}{2}U_D^{(BF)}(0, d) + \frac{1}{2}U_D^{(SQLi)}(0, d) \quad (6a)$$

The dynamic state utility is therefore

$$U_D(s, d) = \begin{cases} U_D(B, d), & s = B, \\ U_D^{(BF)}(i, d), & s = BF_i, \\ U_D^{(SQLi)}(i, d), & s = SQLi_i \end{cases} \quad (6b)$$

Equations (5)–(6) are asymmetric by construction. A failed attack can still impose operating or exposure costs on the attacker, while a defense deployed in a benign window can consume resources and reduce usability. The players' payoffs are therefore not negatives of one another, which is consistent with a non-zero-sum formulation (Nash 1951).

3.2 Model assumptions and consistency conditions

The stage game is internally coherent only when cost ordering, success monotonicity, attack-family heterogeneity, marginal rationality, and the information pattern are specified jointly. The following assumptions impose these requirements.

A1: ordered cost of attack, $0 = c_{k,0}^A < c_{k,1}^A < c_{k,2}^A$ for each family k .

A2: ordered cost of defense, $0 = c_{k,0}^D < c_{k,1}^D < c_{k,2}^D$ and $0 = \gamma_{k,0} \leq \gamma_{k,1} \leq \gamma_{k,2}$. Stronger defenses therefore cost more and induce at least as much operational friction as weaker ones.

A3: ordered success with respect to attack and defense. For each family k , $q_{2d}^{(k)} \geq q_{1d}^{(k)} \geq q_{0d}^{(k)} = 0$ for all d , and $q_{a0}^{(k)} \geq q_{a1}^{(k)} \geq q_{a2}^{(k)}$ for all $a > 0$. Stronger attacks are at least as dangerous as weaker ones, while stronger defenses are at least as suppressive as weaker ones.

A4: family-specific heterogeneity. The matrices $Q^{(BF)}$ and $Q^{(SQLi)}$ are not assumed to be identical, and neither are the corresponding cost and loss parameters.

A5: marginal rationality rather than universal dominance. We do not require the asset value or protection reward to dominate every cost parameter. What matters is whether a marginal increase in attack or defense intensity improves expected utility.

A6: conditional common knowledge within a stage, incomplete information across stages. Conditional on family k , the stage-game structure is common knowledge; over time, however, the defender does not directly observe the true attack family and intensity. This assumption preserves stage-game tractability while motivating the Bayesian extension in Section 4 (Harsanyi 1967).

3.3 Attack-specific payoff design

Separate calibration is used because the same ordinal intensity need not leave the same residual success after defense. Throttling, lockouts, and stronger authentication directly suppress BF pressure, whereas crafted SQL injection can remain consequential when application-side validation is weak. Adaptive password guessing can also alter its candidate distribution after observing outcomes (Murray and Malone 2023; Paul et al. 2024; Wang et al. 2021). Tables 3 and 4 contain the normalized parameters used throughout the equilibrium and simulation calculations. They satisfy Assumptions A1–A4 but are not empirical estimates of compromise probability. Scenario, observation-noise, switching-cost, and intensity-mapping analyses test the influence of these choices.

Table 3 Normalized brute-force parameters

Parameter	Level 0	Level 1	Level 2
Attacker cost $c_{BF,a}^A$	0.00	0.18	0.34
Defender cost $c_{BF,d}^D$	0.00	0.14	0.33
Attacker failure penalty $\eta_{BF,a}$	0.00	0.04	0.08
Defender usability penalty $\gamma_{BF,d}$	0.00	0.05	0.12

The remaining BF reward, loss, and success-probability parameters are consolidated in Equation (6c):

$$V_{BF} = 1.00, \quad R_{BF} = 1.00, \quad L_{BF} = 1.20$$

$$Q^{(BF)} = \begin{bmatrix} 0 & 0 & 0 \\ 0.56 & 0.24 & 0.09 \\ 0.82 & 0.46 & 0.18 \end{bmatrix} \quad (6c)$$

Table 4 provides the corresponding calibration for SQL injection.

Table 4 Normalized SQL-injection parameters

Parameter	Level 0	Level 1	Level 2
Attacker cost $c_{SQLi,a}^A$	0.00	0.22	0.41
Defender cost $c_{SQLi,d}^D$	0.00	0.16	0.34
Attacker failure penalty $\eta_{SQLi,a}$	0.00	0.03	0.07
Defender usability penalty $\gamma_{SQLi,d}$	0.00	0.06	0.14

Note: All entries are normalized analytical design parameters and are not estimated from CSE-CIC-IDS2018.

The corresponding SQLi parameters are consolidated in Equation (6d):

$$V_{SQLi} = 1.20, R_{SQLi} = 1.20, L_{SQLi} = 1.55$$

$$Q^{(SQLi)} = \begin{bmatrix} 0 & 0 & 0 \\ 0.62 & 0.34 & 0.16 \\ 0.88 & 0.58 & 0.29 \end{bmatrix} \quad (6d)$$

The family-specific suppression profiles implied by the analytical matrices in Tables 3 and 4 are compared in Fig. 2.

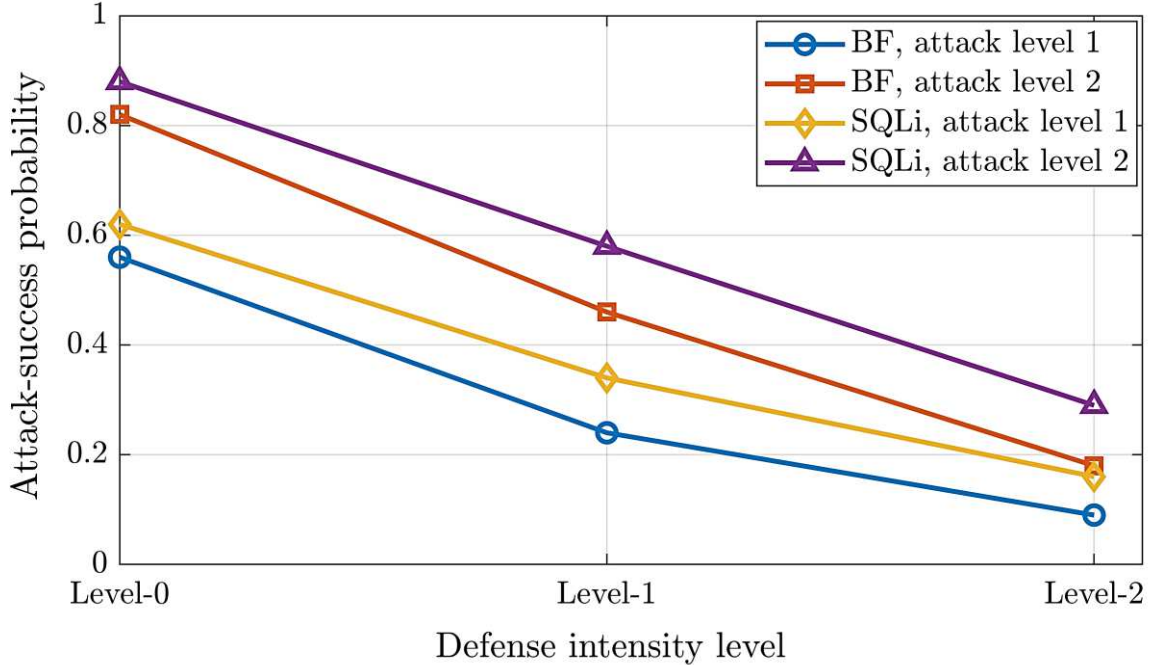


Fig. 2 Analytical attack-success profiles for brute-force and SQL injection

Fig. 2 shows that Level-1 defense reduces high-intensity BF success from 0.82 to 0.46 (43.9%), whereas the corresponding SQLi reduction is from 0.88 to 0.58 (34.1%). At attack level 1, the residual success probabilities under Level-1 defense are 0.24 for BF and 0.34 for SQLi. These different suppression profiles motivate family-conditioned equilibrium analysis; they are consequences of the normalized design matrices rather than measured traffic outcomes.

3.4 Payoff matrices

Using Equations (5)–(6), the attacker and defender payoff matrices for family k take the compact form of Equation (7):

$$A^{(k)} = (U_A^{(k)}(a, d))_{a,d \in \{0,1,2\}}, D^{(k)} = (U_D^{(k)}(a, d))_{a,d \in \{0,1,2\}} \quad (7)$$

For any given family k , the first row of $A^{(k)}$ is identically zero because $a = 0$ corresponds to no attack. By contrast, the first row of $D^{(k)}$ reflects an economically meaningful benign-state trade-off: the defender preserves safe service but may still incur friction if it deploys unnecessary defenses. Retaining this row is important because later benchmarking compares not only success probability, but also resource efficiency, residual utility, and defensive redundancy (Adisa et al. 2025).

3.5 Static equilibrium analysis

Let $x = (x_0, x_1, x_2)^\top \in \Delta(A)$ be the attacker's mixed strategy and $y = (y_0, y_1, y_2)^\top \in \Delta(D)$ the defender's mixed strategy. For attack family k , the expected utilities become Equation (8):

$$\Pi_A^{(k)}(x, y) = x^\top A^{(k)} y, \Pi_D^{(k)}(x, y) = x^\top D^{(k)} y \quad (8)$$

The expected payoff from each pure action is stated in Equations (9) and (10):

$$\bar{U}_{A,a}^{(k)}(y) = \sum_{d=0}^2 y_d U_A^{(k)}(a, d) \quad (9)$$

$$\bar{U}_{D,d}^{(k)}(x) = \sum_{a=0}^2 x_a U_D^{(k)}(a, d) \quad (10)$$

Under a full-support equilibrium, these payoffs satisfy the system in Equations (11)–(13):

$$A^{(k)} y^{*(k)} = \Pi_A^{*(k)} \mathbf{1} \quad (11)$$

$$(x^{*(k)})^\top D^{(k)} = \Pi_D^{*(k)} \mathbf{1}^\top \quad (12)$$

$$x_a^{*(k)} \geq 0, y_d^{*(k)} \geq 0, \quad \forall a, d \quad (13)$$

If any component of the candidate solution implied by Equations (11)–(13) is negative, the equilibrium lies on a boundary support rather than in the full interior, and the game should be solved on the relevant lower-dimensional support.

Proposition 1. For each attack family k and any admissible parameter set satisfying Assumptions A1–A6, the refined three-level stage game admits at least one mixed-strategy Nash equilibrium.

Proof. The game is finite: each player has three pure actions and finite payoffs in every cell. Therefore, by Nash's existence theorem for finite non-cooperative games, at least one mixed-strategy equilibrium exists. The admissibility conditions ensure that all payoff entries are well-defined and bounded (Nash 1951).

Proposition 2. Let x be any attacker mix for family k . High-intensity defense is weakly suboptimal whenever the condition in Equation (14) holds:

$$\sum_{a=1}^2 x_a [q_{a1}^{(k)} - q_{a2}^{(k)}] (R_k + L_k) \leq [c_{k,2}^D - c_{k,1}^D] + x_0 [\gamma_{k,2} - \gamma_{k,1}] \quad (14)$$

Proof. From Equation (6), $U_D^{(k)}(a, 2) - U_D^{(k)}(a, 1) = (q_{a1}^{(k)} - q_{a2}^{(k)}) (R_k + L_k) - (c_{k,2}^D - c_{k,1}^D) - (\gamma_{k,2} - \gamma_{k,1}) \mathbf{1}_{\{a=0\}}$. Taking expectation with respect to x gives the left-hand side of Equation (14) minus its right-hand side. When Equation (14) holds, the expected payoff difference between defense levels 2 and 1 is non-positive; hence high-intensity defense is weakly suboptimal.

Define the expected payoff gain of defense level d over d' under family k by Equation (15):

$$G_k^{(d,d')}(x) = \sum_{a=0}^2 x_a [U_D^{(k)}(a, d) - U_D^{(k)}(a, d')] \quad (15)$$

A strict family-dependent ranking reversal occurs when Equation (16) holds:

$$G_{BF}^{(d,d')}(x)G_{SQLi}^{(d,d')}(x) < 0 \quad (16)$$

Proposition 3 (family-dependent defense ranking). If Equation (16) holds for some attacker mixture x and pair $d \neq d'$, the two defense levels are ranked in opposite orders in the BF and SQLi games. A pooled payoff matrix assigns only one aggregate gain to the pair and therefore cannot preserve both strict family-specific rankings simultaneously. Different equilibrium supports may follow, but are not asserted without additional uniqueness assumptions.

Proof. A negative product in Equation (16) means that one gain is positive and the other is negative. Hence d is preferred for one family and d' for the other, whereas a pooled game produces only one aggregate ordering.

4 Dynamic Bayesian Defense Model

The one-period utilities become a sequential decision model after the defender assigns probabilities to the latent state and updates them as observations arrive. Logs, traffic statistics, request patterns, and authentication behavior provide indirect evidence rather than direct knowledge of attack family and intensity. The Bayesian filter maps that evidence to the state probabilities used in the defense rule (Harsanyi 1967; Liu et al. 2021; Zhang and Malacaria 2021; Zheng et al. 2022).

4.1 Sequential decision setting

Consider a finite horizon of discrete traffic windows $t \in \{1, 2, \dots, T\}$. Each window may correspond to a fixed time interval, a batch of network flows, or an application-layer monitoring slice. Within each window, the attacker has effectively chosen a latent attack family and intensity, but the defender does not observe this choice directly. Instead, the defender receives a feature vector extracted from the window and must decide which defense level to deploy. The latent state at each window is represented by Equation (17):

$$\mathcal{S} = \{B, BF_1, BF_2, SQLi_1, SQLi_2\} \quad (17)$$

The state B denotes a benign window. For an attack state k_i , $f(k_i) = k$ identifies the family and $\iota(k_i) = i$ identifies the intensity.

4.2 Hidden state and belief representation

Let $\mathcal{S}$ be the five-state space defined in Equation (17). The defender's posterior belief at time t is

$$b_t(s) = \Pr(s_t = s \mid o_{1:t}), \quad s \in \mathcal{S} \quad (18)$$

To model temporal persistence and escalation, we assume a first-order Markov law for the latent state. This yields the prediction step, shown in Equation (19):

$$\hat{b}_t(s) = \sum_{u \in \mathcal{S}} P(s \mid u) b_{t-1}(u) \quad (19)$$

For interpretation, the family and intensity marginals are defined in Equation (20):

$$\begin{aligned} b_t^K(BF) &= b_t(BF_1) + b_t(BF_2), \quad b_t^K(SQLi) = b_t(SQLi_1) + b_t(SQLi_2), \\ b_t^I(0) &= b_t(B), \quad b_t^I(i) = b_t(BF_i) + b_t(SQLi_i), \quad i \in \{1, 2\} \end{aligned} \quad (20)$$

4.3 Observation model

The observation vector associated with the current window is denoted by Equation (21). In a deployment, ϕ may extract this vector from monitored data; in the present controlled simulation, o_t is generated from the Gaussian model specified in Equations (37)–(38):

$$o_t = \phi(\omega_t) \in \mathbb{R}^F, \quad F = 6 \text{ in the controlled simulation} \quad (21)$$

The six coordinates are standardized synthetic indicators designed to distinguish authentication-oriented, volume-oriented, and request-content-oriented evidence. They preserve the qualitative semantics of brute-force and SQL-injection observations, but they are not CICFlowMeter fields and are not claimed to be measured traffic features. This abstraction deliberately separates evidence types: SQL injection has been detected from lightweight network-flow summaries, whereas query-string methods analyze application-layer syntax; the six synthetic coordinates should therefore be read as generic evidence channels rather than measured replicas of either feature space

(Crespo-Martínez et al. 2023; McWhirter et al. 2018). Conditional on the hidden state, the likelihood used by the filter is the product of independent Gaussian densities in Equation (22):

$$p(o_t | s_t = s) = \prod_{j=1}^F \mathcal{N}(o_{tj}; \mu_{sj}, \sigma^2), \quad s \in \mathcal{S} \quad (22)$$

4.4 Bayesian update rule

The posterior then follows from Bayes' rule as Equation (23):

$$b_t(s) = \frac{p(o_t | s) \hat{b}_t(s)}{\sum_{u \in \mathcal{S}} p(o_t | u) \hat{b}_t(u)} \quad (23)$$

The posterior expected attack intensity is obtained from Equation (24):

$$\rho_t = \sum_{s \in \mathcal{S}} \iota(s) b_t(s) \quad (24)$$

4.5 Myopic Bayesian defense rule

The defender evaluates each defense level through the posterior expected utility in Equation (25):

$$u_t(d) = \sum_{s \in \mathcal{S}} b_t(s) U_D(s, d) - \kappa |d - d_{t-1}| \quad (25)$$

The deployed action is then selected according to Equation (26):

$$d_t^* = \min * \arg \max_{d \in \mathcal{D}} u_t(d) \quad (26)$$

Equation (26) defines a myopic posterior-utility policy. The outer minimum is used only to resolve an exact tie by choosing the lowest defense level, matching the executable implementation. When $\kappa > 0$, the present action affects the next window's switching penalty, so the rule is not claimed to solve a Bellman finite-horizon problem.

For completeness, define $\mathbf{u}_D(s) = [U_D(s, 0), U_D(s, 1), U_D(s, 2)]^\top$. Equation (27) gives an optional mixed-defense relaxation. The experiments do not use this relaxation; they implement the pure rule in Equation (26).

$$y_t^* \in * \arg \max_{y \in \Delta(\mathcal{D})} \left\{ \sum_{s \in \mathcal{S}} b_t(s) \mathbf{u}_D(s)^\top y - \kappa \sum_{d \in \mathcal{D}} y_d |d - d_{t-1}| \right\} \quad (27)$$

4.6 Repeated decision process

The dynamic model unfolds window by window. At each time step, the defender receives the next traffic window, extracts the feature vector, predicts the next prior from the previous posterior using the transition matrix, updates the posterior using Bayes' rule, evaluates the posterior expected utility of each defense level, deploys the maximizing action, and records the selected action and state-conditioned expected performance measures.

The complete repeated-decision objective across all windows is stated in Equation (28):

$$J_D^\pi = \mathbb{E}^\pi \left[\sum_{t=1}^T (U_D(s_t, d_t) - \kappa |d_t - d_{t-1}|) \right] \quad (28)$$

Equation (28) summarizes the state-conditioned expected stage utilities accumulated along the policy sequence. It is a retrospective performance criterion rather than an optimality claim for Equation (26). The mean utility reported in Equation (47) is the corresponding sample average, obtained by dividing the accumulated sum by T .

Proposition 4 (monotonicity under within-family escalation). Suppose that, for each $k \in \mathcal{K}$ and $d \in \{0, 1\}$, the defender utility satisfies increasing differences between attack intensities 1 and 2:

$$U_D^{(k)}(2, d+1) - U_D^{(k)}(2, d) \geq U_D^{(k)}(1, d+1) - U_D^{(k)}(1, d) \quad (29)$$

Hold $b_t(B)$ and the family masses $b_t^K(k)$ fixed, and shift conditional probability within each family from k_1 toward k_2 in the sense of first-order stochastic dominance. Then the selected action under Equation (26) weakly increases.

Proof sketch. Equation (29) gives increasing differences in attack intensity and defense level. A within-family upward shift therefore raises the marginal expected value of stronger defense; monotone comparative statics imply a nondecreasing stage-wise argmax (Milgrom and Shannon 1994).

4.7 Myopic Bayesian defense-selection algorithm

The complete window-by-window procedure is set out in Algorithm 1.

Algorithm 1. Window-wise myopic Bayesian defense selection

Input: prior b_0 over S ; transition matrix P ; observation means $\mu_{s \in S}$; noise standard deviation σ ; state-utility matrix U_D ; horizon T ; switching cost κ .

Output: $b_t, d_{t=1}^{*T}$ and the benchmark metrics.

Initialize: $d_0 \leftarrow 0$.

For $t = 1, \dots, T$:

1. Obtain o_t from the declared evidence source.

2. Predict $\hat{b}_t(s) = \sum_{s' \in S} P(s|s') b_{t-1}(s')$ for every $s \in S$.

3. Update $b_t(s) = p(o_t|s) \hat{b}_t(s) / \sum_{s' \in S} p(o_t|s') \hat{b}_t(s')$.

4. Compute $U_t(d) = \sum_{s \in S} b_t(s) U_D(s, d) - \kappa |d - d_{t-1}|$ for $d \in \{0, 1, 2\}$.

5. Deploy $d_t^* = \min \arg \max_{d \in D} U_t(d)$.

6. Record b_t, d_t^* , energy expenditure, the LDSR indicator, the ECR contribution, and state-conditioned expected stage utility.

End For.

Fig. 3 summarizes the five-state myopic Bayesian policy in Algorithm 1. Solid arrows show the within-window prediction, update, utility-evaluation, and action-selection sequence; the dashed arrow carries the posterior forward to the next window. Exact ties are resolved by selecting the lowest defense level.

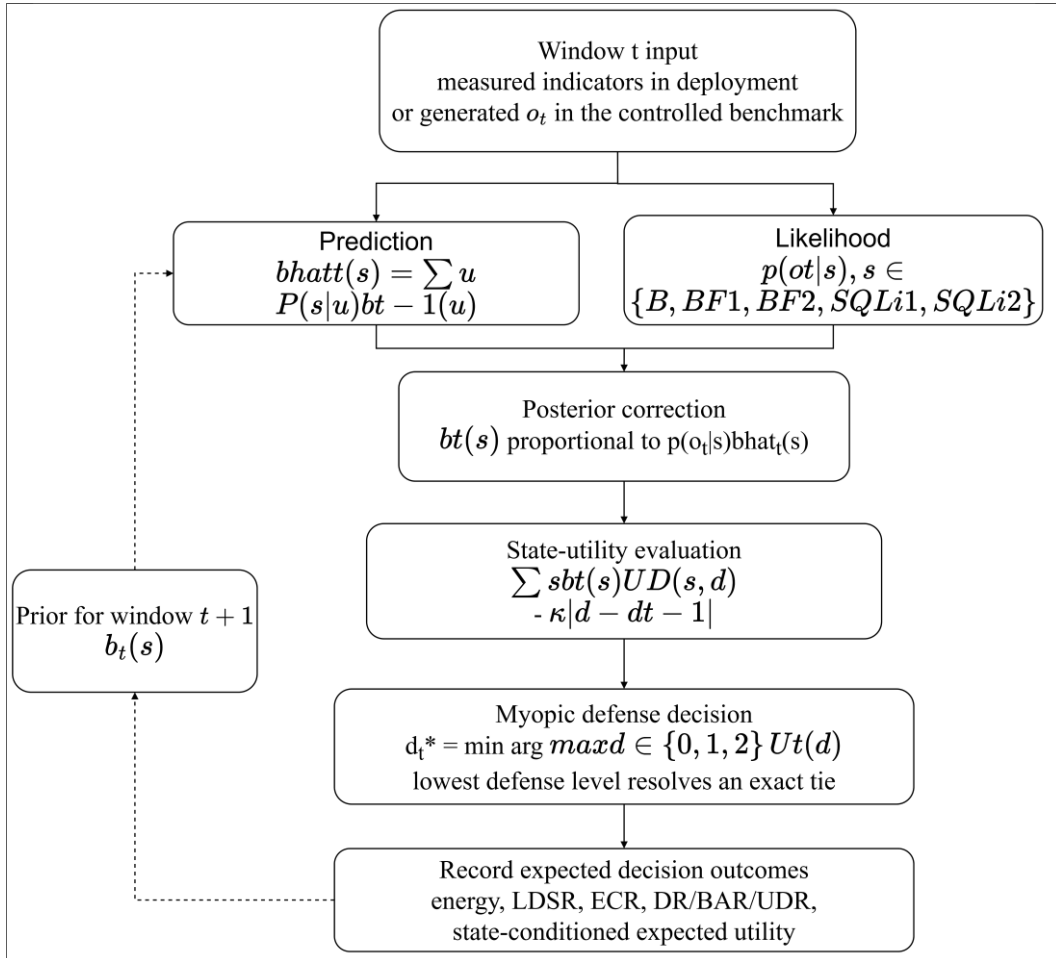


Fig. 3 Window-wise myopic Bayesian defense pipeline

Fig. 3 separates inference from decision evaluation. The transition and likelihood models determine the posterior b_t , whereas the family-conditioned utility matrix and switching coefficient κ determine the selected action. Observation uncertainty therefore affects defense through the posterior rather than by altering the payoff parameters.

5 Experimental Design and Benchmarking Protocol

The evaluation uses two layers because source-paper comparability and sequential state inference require different state processes. Layer A reproduces the 10,000-cycle random-attacker setting with fixed Level-0/1/2 defenses and the Level-M reconstructions. Layer B uses a calendar-defined hidden-state sequence and Gaussian observations to test belief updating under temporal persistence. Fig. 4 summarizes the two paths and the decision rules evaluated in each.

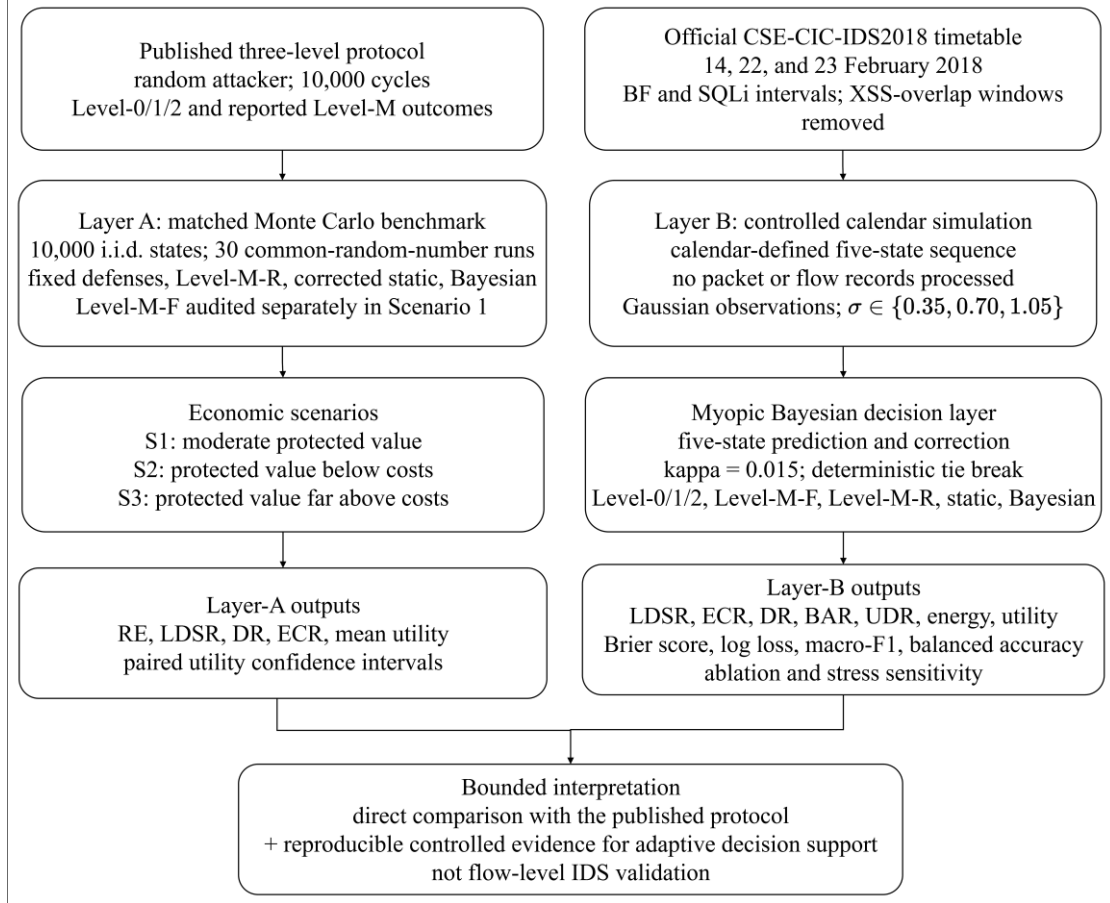


Fig. 4 Dual-layer evaluation workflow

Fig. 4 distinguishes the state and evidence sources. Layer A uses independent matched states, whereas Layer B uses a calendar-defined sequence and synthetic observations. Method definitions are held fixed within each applicable layer; the state process and evidence source are what change.

All simulations were run in Python 3.11.2 using NumPy 1.26.4, SciPy 1.13.1, pandas 2.2.2, and nashpy 0.0.40. Support enumeration generated the reported bimatrix equilibria, and the Lemke–Howson algorithm provided a numerical consistency check. Within each replication, common random numbers and matched seeds ensured that all methods encountered identical hidden-state and observation sequences.

5.1 Public attack calendar and window construction

The official CSE-CIC-IDS2018 release provides the documented brute-force and web-attack times used to construct the temporal calendars in this study. Only the published timetable is used; packet captures, flow records, and CICFlowMeter features are not processed. The source therefore supports an auditable calendar-conditioned simulation rather than a flow-level dataset experiment (Canadian Institute for Cybersecurity n.d.; Leevy and Khoshgoftaar 2020).

Layer B uses 5-minute non-overlapping windows over the official 09:00-17:00 capture horizon. Let the t -th window be denoted as in Equation (30):

$$\omega_t = [\theta_t, \theta_t + \Delta), \quad \Delta = 5 \text{ min} \quad (30)$$

Let $\mathcal{J}_{k,r}$ denote the r -th official interval for attack family $k \in \{\text{BF}, \text{SQLi}, \text{XSS}\}$. Windows that overlap the official cross-site scripting (XSS) spells are removed because XSS lies outside the scope of the proposed model. The retained usable window set is constructed via Equation (31):

$$\mathcal{W}_{\text{use}} = \{\omega_t: \omega_t \cap \mathcal{J}_{\text{XSS},r} = \emptyset \quad \forall r\} \quad (31)$$

For each retained window, the calendar-derived ground-truth family label z_t is assigned by Equation (32). This label is used to construct the hidden evaluation state and is not observed by the defender:

$$z_t = \begin{cases} 0, & \omega_t \cap \left(\bigcup_{k \in \{\text{BF}, \text{SQLi}\}} \bigcup_r \mathcal{J}_{k,r} \right) = \emptyset, \\ k, & \omega_t \cap \mathcal{J}_{k,r} \neq \emptyset \text{ for some } r, k \in \{\text{BF}, \text{SQLi}\} \end{cases} \quad (32)$$

The five-minute state calendars for 14 and 22 February are used only to calculate the development prior and transition counts. The 23 February calendar is not used in those calculations and is retained as the hold-out temporal sequence. The observation means remain analytical design parameters. A day-wise development/hold-out separation also reduces the risk that transition estimation or calibration will use information from the held-out calendar; intrusion-detection research shows more generally that preprocessing leakage can inflate apparent performance and weaken reliability (Bouke and Abdullah 2023).

Table 5 reports the state counts used to estimate the development prior and transition matrix and the separate counts retained for the hold-out calendar.

Table 5 Calendar-derived state counts

Date	Retained attack event(s)	Usable 5-min windows	Benign	Brute-force		SQL injection		Study use
				Level 1	Level 2	Level 1	Level 2	
Wed 14-02-2018	FTP-BruteForce, SSH-Bruteforce	96	57	24	15	0	0	Development
Thu 22-02-2018	Brute Force-Web, SQL Injection	88	71	10	4	1	2	Development
Fri 23-02-2018	Brute Force-Web, SQL Injection	82	66	8	5	1	2	Hold-out calendar

The two development calendars contain 184 windows, while the hold-out calendar contains 82 windows, 66 of which are benign. This class structure motivates reporting BAR and balanced accuracy alongside aggregate LDSR.

5.2 Calendar-to-intensity scenario mapping

The source paper distinguishes low- and high-intensity variants of both password guessing and SQL injection. In this study, intensity is assigned as a temporal scenario label rather than learned from traffic records.

Let $C_j = \{t_{j,s}, \dots, t_{j,e}\}$ denote the j th contiguous attack-family spell, let $k_j \in \{\text{BF}, \text{SQLi}\}$ be its family, and let $n_j = |C_j|$. For $t \in C_j$, the operational-state intensity satisfies $\iota(s_t) = \iota_t^{(k_j)}$.

$$\iota_t^{(\text{BF})} = \begin{cases} 1, & t - t_{j,s} < 0.30 \cdot n_j \text{ or } t_{j,e} - t < 0.30 \cdot n_j, \\ 2, & \text{otherwise,} \end{cases} \quad t \in C_j, k_j = \text{BF} \quad (33)$$

For SQL injection, the intensity assignment is given by Equation (34):

$$\iota_t^{(\text{SQLi})} = \begin{cases} 1, & t_{j,s} \leq t < t_{j,s} + \max\{1, \lfloor n_j/3 \rfloor\}, \\ 2, & \text{otherwise,} \end{cases} \quad t \in C_j, k_j = \text{SQLi} \quad (34)$$

Table 6 summarizes the declared calendar-to-intensity mapping generated by Equations (33)–(34).

Table 6 Calendar-to-intensity scenario mapping

State Family	Level-0 condition	Level-1 rule	Level-2 rule	Scenario semantics
Brute-force	No overlap with official <i>BF</i> interval	First/last 30% of a <i>BF</i> spell	Interior of a <i>BF</i> spell	Boundary-phase Level 1 versus interior-phase Level 2 (declared scenario semantics).
SQL injection	No overlap with official <i>SQLi</i> interval	First one-third of a <i>SQLi</i> spell	Remaining windows of the <i>SQLi</i> spell	First-window Level 1 versus remaining-window Level 2 (declared scenario semantics).
Benign	No overlap with any retained <i>BF</i> or <i>SQLi</i> spell	-	-	No modeled attack activity

The Level-1 and Level-2 descriptions in Table 6 are declared scenario semantics. Temporal position within an attack spell does not by itself establish whether the underlying activity was manual or automated.

5.3 Observation model and benchmark parameterization

Implementation uses the five-state process $\{s_t\}_{t=1}^T$ defined in Equation (17). The development prior, computed from the 14 and 22 February calendar counts in Table 5, is given in Equation (35):

$$b_0 = (128, 34, 19, 1, 2)/184 = (0.695652, 0.184783, 0.103261, 0.005435, 0.010870) \quad (35)$$

Each window generates the six-dimensional sensor vector defined by Equation (36):

$$o_t = (l_t, r_t, u_t, v_t, \psi_t, h_t) \quad (36)$$

where l_t is failed-login burst score, r_t account/IP reuse score, u_t request-rate anomaly score, v_t SQL-token risk score, ψ_t parameter-shape anomaly score, and h_t payload-entropy score. Conditional on the hidden state, the observation model takes the Gaussian form of Equation (37):

$$o_t | s_t = s \sim \mathcal{N}(\mu_s, \sigma^2 I_6), \quad \sigma = 0.70, \quad (37)$$

The state-specific means are fixed analytically, as listed in Equation (38):

$$\begin{aligned} \mu_B &= (0, 0, 0, 0, 0, 0), \\ \mu_{BF_1} &= (1.8, 1.5, 1.3, 0.1, 0.1, 0.2), \\ \mu_{BF_2} &= (3.0, 2.7, 2.8, 0.1, 0.1, 0.3), \\ \mu_{SQLi_1} &= (0.1, 0.1, 0.8, 1.9, 1.7, 1.3), \\ \mu_{SQLi_2} &= (0.1, 0.1, 1.4, 3.1, 2.8, 2.3) \end{aligned} \quad (38)$$

The primary $\sigma = 0.70$ is the midpoint of the predeclared stress grid $\{0.35, 0.70, 1.05\}$; it is not estimated from CSE-CIC-IDS2018. The low-, moderate-, and high-noise results are all reported in Section 6.5.

The Layer-B transition counts are calculated within the 14 and 22 February development calendars. One unit pseudocount is added only to admissible transitions $B \rightarrow \{B, BF1, SQLi1\}$, $BF_i \rightarrow \{B, BF1, BF2\}$, and $SQLi_i \rightarrow \{B, SQLi1, SQLi2\}$. Equation (39) is then the row-normalized matrix shown immediately below.

$$P_B = \begin{bmatrix} 0.953488 & 0.031008 & 0.000000 & 0.015504 & 0.000000 \\ 0.108108 & 0.783784 & 0.108108 & 0.000000 & 0.000000 \\ 0.045455 & 0.181818 & 0.772727 & 0.000000 & 0.000000 \\ 0.250000 & 0.000000 & 0.000000 & 0.250000 & 0.500000 \\ 0.400000 & 0.000000 & 0.000000 & 0.200000 & 0.400000 \end{bmatrix} \quad (39)$$

$$P_A = \mathbf{1}p_A^\top, \quad \text{where } p_A = (0.6774, 0.1593, 0.0020, 0.1593, 0.0020) \quad (39a)$$

The switching coefficient is fixed at $\kappa = 0.015$, equal to 10% of the mean Level-1 defense cost: $0.10 \times (0.14 + 0.16)/2$. It is not optimized on the hold-out sequence. Section 6.5 reports $\kappa \in \{0, 0.015, 0.03\}$.

To benchmark fairly against Adisa et al. (2025), Layer A preserves the same three scenario families used in the source paper: the protected resource exceeds attack and defense costs; the protected resource is lower than attack and defense costs; and the protected resource is significantly greater than attack and defense costs. The three

matched economic regimes are indexed by $g \in \{1,2,3\}$; their scenario-specific rewards and losses are defined by Equation (40):

$$(V_k^{(g)}, R_k^{(g)}, L_k^{(g)}) = \lambda_g(V_k, R_k, L_k), (\lambda_1, \lambda_2, \lambda_3) = (1.00, 0.55, 2.20) \quad (40)$$

Let $\hat{\pi}$ denote the fixed marginal intensity law calibrated for Layer-A matching. It is not estimated from the Layer-B calendar and is specified in Equation (41):

$$\hat{\pi} = (\pi_0, \pi_1, \pi_2) = (0.6774, 0.3186, 0.0040) \quad (41)$$

Conditional on an attack, BF and SQLi are assigned equal probability because the source paper does not report a family mixture. This gives $p_A = (0.6774, 0.1593, 0.0020, 0.1593, 0.0020)$. The Layer-A states are independent across cycles, every row of P_A equals $p_A^\top$, observations use $\sigma = 0.70$ and the declared means, and the Bayesian policy uses $\kappa = 0.015$.

The source article provides printed strategy expressions and reported aggregate outcomes that do not reproduce one another under the declared Scenario-1 parameters. We therefore distinguish two interpretations. Level-M-F follows the printed formula and gives $y_{MF} = (0.2, 0.2, 0.6)$. Level-M-R is the reported-outcome-matched proxy used for scenario-wise comparison. The scenario-wise Level-M-R strategy vectors are given in Equation (42):

$$y_{MR}^{(1)} = (0.669, 0, 0.331), \quad y_{MR}^{(2)} = (1, 0, 0), \quad y_{MR}^{(3)} = (0, 0.011, 0.989) \quad (42)$$

The corrected static comparator is generated by averaging the BF and SQLi payoff matrices with fixed equal family weights and solving the resulting pooled game separately in each economic scenario. The corrected static equilibrium strategies are reported in Equation (43):

$$y_{CS}^{(1)} = (0, 0.475888, 0.524112), \quad y_{CS}^{(2)} = (0.245550, 0.754450, 0), \quad y_{CS}^{(3)} = (0, 0, 1) \quad (43)$$

The seven benchmark methods and their executable decision rules are listed in Table 7; the proposed ATA-DBD policy is labeled "Bayesian" in all tables and figures.

Table 7 Benchmark methods and decision rules

Method	Decision Rule	Uses posterior evidence?	Layer A	Layer B
Level-0	Set $d_t = 0$ in every window	No	Yes	Yes
Level-1	Set $d_t = 1$ in every window	No	Yes	Yes
Level-2	Set $d_t = 2$ in every window	No	Yes	Yes
Level-M-F (formula)	Sample $y_{MF} = (0.2, 0.2, 0.6)$	No	Scenario-1 audit only	Yes
Level-M-R (reported-outcome proxy)	Sample the declared scenario-wise y_{MR} vector	No	Yes	Yes (S1 vector)
Corrected static	Sample the equal-weight pooled equilibrium y_{CS}	No	Yes	Yes (S1 vector)
Bayesian	Choose the lowest-index defense in $* \arg \max_{d \in D} \mathcal{U}_t(d)$.	Yes	Yes	Yes

Table 8 summarizes the reward/loss scaling used to construct the three Layer-A economic scenarios; the attack and defense cost ladders remain fixed.

Table 8 Layer-A economic scenarios and reward/loss scaling

Scenario	Economic regime	Reward/loss multiplier, λ_g	Expected defense posture
S1	Asset value exceeds cost, but not overwhelmingly	1.00	Economical mixed defense
S2	Asset value is below the attack/defense cost ladder	0.55	Cost-sensitive restraint
S3	Asset value greatly exceeds the cost ladder	2.20	Aggressive protection becomes attractive

5.4 Evaluation metrics

To preserve direct comparability with Adisa et al. (2025), residual energy (RE), legacy defense-success rate (LDSR), and defensive redundancy (DR) are retained and defined in Equations (44)–(46). LDSR is explicitly labeled legacy because it measures ordinal adequacy rather than probabilistic attack success:

$$\text{RE} = 100 - \sum_{t=1}^T \epsilon(d_t), \quad T = 10,000, \quad (44)$$

$$\text{LDSR} = \frac{100}{T} \sum_{t=1}^T \mathbf{1}\{d_t \geq \iota(s_t)\}, \quad (45)$$

$$\text{DR} = \frac{100}{T} \sum_{t=1}^T \mathbf{1}\{d_t > 0, s_t = B\} \quad (46)$$

The remaining allocation, containment, delay, and posterior-quality metrics are defined in Equations (47)–(51c). Brier score and log loss are retained because they are proper scoring rules for probabilistic predictions and therefore complement the discrete state-classification summaries reported elsewhere (Silva Filho et al. 2023).

$$\bar{U}_D = \frac{1}{T} \sum_{t=1}^T [U_D(s_t, d_t) - \kappa |d_t - d_{t-1}|], \quad (47)$$

$$\text{BAR} = 100 \frac{\sum_{t=1}^T \mathbf{1}\{s_t = B, d_t > 0\}}{\sum_{t=1}^T \mathbf{1}\{s_t = B\}}, \quad (48)$$

$$\text{UDR} = 100 \frac{\sum_{t=1}^T \mathbf{1}\{s_t \neq B, d_t < \iota(s_t)\}}{\sum_{t=1}^T \mathbf{1}\{s_t \neq B\}}, \quad (49)$$

$$\bar{C}_E = \frac{1}{T} \sum_{t=1}^T \epsilon(d_t), \quad (50)$$

$$\tau_j = \begin{cases} \min\{t - t_{j,s}^0 : t \in C_j, d_t \geq \iota(s_t)\}, & \text{if an adequate action occurs in } C_j, \\ |C_j|, & \text{otherwise,} \end{cases}$$

$$\bar{\tau}_{\text{onset}} = \frac{1}{N_{\text{spells}}} \sum_{j=1}^{N_{\text{spells}}} \tau_j \quad (51)$$

This metric measures the first ordinal adequate response after attack-family spell onset. It does not measure a second delay when intensity later escalates within the same family spell; UDR captures any such under-defense windows. Here $T_A = \sum_{t=1}^T \mathbf{1}\{s_t \neq B\}$ is the number of attack windows.

$$\text{ECR} = \frac{100}{T_A} \sum_{t: s_t \neq B} [1 - q_{\iota(s_t), d_t}^{(f(s_t))}], \quad (51a)$$

$$\text{Brier} = \frac{1}{T} \sum_{t=1}^T \sum_{s \in \mathcal{S}} [b_t(s) - \mathbf{1}\{s_t = s\}]^2, \quad (51b)$$

$$\text{LogLoss} = -\frac{1}{T} \sum_{t=1}^T \log b_t(s_t) \quad (51c)$$

For posterior classification summaries, define $\hat{s}_t = \arg \max_{s \in \mathcal{S}} b_t(s)$. State accuracy is the percentage of windows for which $\hat{s}_t = s_t$; macro-F1 is the unweighted mean of the five state-wise F1 scores; and balanced accuracy is the unweighted mean of the five state-wise recalls. The reported attack-spell onset delay is the mean of τ_j across attack spells C_j . If a spell is never adequately defended, the delay is right-censored at its length. Because fixed rules and memoryless mixed rules do not react to evidence over time, attack-spell onset delay is reported only for the Bayesian policy. A consolidated overview of all evaluation metrics is given in Table 9.

Table 9 Evaluation metrics

Metric	Operational definition	Directly comparable with Adisa et al. (2025)
Residual energy (RE)	Resource remaining after the Layer-A energy schedule	Yes
Legacy defense-success rate (LDSR)	Ordinal adequacy: d_t is at least the scenario intensity	Yes
Defensive redundancy (DR)	Percentage of all windows with active defense during a benign state	Yes
Mean defender utility	Per-window state-conditioned expected stage utility including switching cost	No
Benign activation rate (BAR)	Conditional percentage of benign windows with $d_t > 0$	No
Under-defense rate (UDR)	Conditional percentage of attack windows with d_t below the scenario intensity	No
Mean energy expenditure	Average $\epsilon(d_t)$ per window	No
Expected containment rate (ECR)	Mean of $1 - q_{i,d_t}^{(k)}$ across attack windows	No
Attack-spell onset delay	Windows from family-spell onset to the first ordinally adequate action, right-censored at spell length	No
Brier score	Mean squared error of the five-state posterior probability vector	No
Log loss	Negative log posterior probability assigned to the true scenario state	No
State accuracy	Percentage of windows for which $\hat{s}_t = s_t$	No
Macro-F1	Unweighted mean F1 over the five scenario states	No
Balanced accuracy	Unweighted mean recall over the five scenario states	No

5.5 Statistical analysis

Layer A uses 30 common-random-number replications with seeds 60000–60029. Each replication contains 10,000 hidden states and one common observation sequence shared by all methods. Means and standard deviations (SD) are reported, and the Bayesian–Level-M-R utility difference is summarized by the paired 95% t-confidence interval (CI) in Equation (52).

$$\bar{m} \pm t_{(0.975, n-1)} \frac{s}{\sqrt{n}} \quad (52)$$

Here $\bar{m}$ is the mean paired difference, s is the sample standard deviation of the paired differences, n is the number of paired replications, and $t_{\{0.975, n-1\}}$ is the 97.5th percentile of the Student’s t distribution with $n - 1$ degrees of freedom.

Layer B uses 300 stochastic observation replications with seeds 40000–40299 over the same held-out 23 February state calendar. These replications quantify simulation uncertainty and are not treated as 300 independent public test days. Means and standard deviations are reported. Paired confidence intervals refer only to the simulated observation mechanism. Ablation uses seeds 50000–50299; sensor-noise sensitivity uses 71000–71299; switching-cost sensitivity uses 72000–72299; and BF-label sensitivity uses 74000–74299. All seeds and output tables are generated by the simulation code provided as Online Resource 1.

6 Results and Discussion

The results separate internal model properties from comparative policy performance. The equilibrium analysis first establishes feasibility and family-dependent support behavior; Layer A then tests benchmark comparability, and Layer B evaluates posterior-conditioned allocation under a persistent calendar-driven state process.

6.1 Equilibrium properties of the refined stage game

The refined stage games were first evaluated for simplex feasibility, support consistency, and economically interpretable best responses. Across S1–S3, every reported mixed strategy is non-negative and sums to one. Under

S2, high-intensity defense leaves the equilibrium support, whereas the larger reward/loss scale in S3 shifts mass toward stronger protection, particularly for SQLi. Table 10 reports the complete scenario–family equilibrium set.

Table 10 Family-conditioned equilibria

Scenario	Attack Family	Attacker equilibrium, $\mathbf{x}^*(\mathbf{k})$	Defender equilibrium, $\mathbf{y}^*(\mathbf{k})$	Defender equilibrium utility, $\Pi_D^*(\mathbf{k})$
S1	BF	(0.621, 0.000, 0.379)	(0.000, 0.746, 0.254)	0.445
S1	SQLi	(0.548, 0.452, 0.000)	(0.000, 0.240, 0.760)	0.584
S2	BF	(0.565, 0.435, 0.000)	(0.415, 0.585, 0.000)	0.256
S2	SQLi	(0.545, 0.455, 0.000)	(0.080, 0.920, 0.000)	0.233
S3	BF	(0.818, 0.000, 0.182)	(0.000, 0.015, 0.985)	1.613
S3	SQLi	(0.000, 0.000, 1.000)	(0.000, 0.000, 1.000)	0.545

Whenever a full-support candidate is infeasible, the selected boundary support follows the payoff ordering. Family conditioning also changes the support: in S1, the BF defender mix is concentrated on Level 1, whereas the SQLi mix places most mass on Level 2. Proposition 3 explains why a pooled game can retain only one of these strict family-specific rankings.

6.2 Matched benchmark against the published three-level framework

Layer A retains the source paper’s 10,000-cycle setting, fixed Level-0/1/2 baselines, three economic regimes, and three legacy metrics, while adding mean defender utility and expected containment. Because the published Level-M expressions and the reported Scenario 1 aggregate outcomes do not reproduce one another under the declared parameters, Table 11 reports the formula-faithful and outcome-matched interpretations. Table 12 then compares the six scenario-wise methods across the three economic regimes.

Table 11 Audit of Level-M under Scenario 1

Level-M representation	Defender strategy, $\mathbf{y}$	RE (%)	LDSR (%)	DR (%)
Published aggregate result	not reported	80.21	78.300	22.430
Level-M-F: printed formula	(0.200, 0.200, 0.600)	58.00	93.468	54.192
Level-M-R: outcome-matched proxy	(0.669, 0.000, 0.331)	80.14	78.418	22.416

The formula-faithful strategy does not reproduce the published aggregate profile under the declared Scenario-1 parameters. Level-M-R is therefore used for scenario-wise outcome comparison, while Level-M-F is retained as a separate transparency audit and Layer-B comparator.

Table 12 Layer-A benchmark results

Scenario	Method	RE (%)	LDSR (%)	DR (%)	Mean defender utility	ECR (%)
S1	Level-0	100.00 ± 0.00	67.61 ± 0.48	0.00 ± 0.00	0.622 ± 0.007	40.65 ± 0.06
S1	Level-1	70.00 ± 0.00	99.58 ± 0.07	67.61 ± 0.48	0.673 ± 0.003	70.69 ± 0.08
S1	Level-2	40.00 ± 0.00	100.00 ± 0.00	67.61 ± 0.48	0.573 ± 0.001	87.35 ± 0.05
S1	Level-M-R	80.25 ± 0.31	78.32 ± 0.44	22.22 ± 0.41	0.593 ± 0.006	56.09 ± 0.44
S1	Corrected static	54.29 ± 0.14	99.80 ± 0.04	67.61 ± 0.48	0.613 ± 0.003	79.40 ± 0.14
S1	Bayesian	80.40 ± 0.31	99.15 ± 0.07	2.28 ± 0.17	0.858 ± 0.003	85.64 ± 0.12
S2	Level-0	100.00 ± 0.00	67.61 ± 0.48	0.00 ± 0.00	0.342 ± 0.004	40.65 ± 0.06
S2	Level-1	70.00 ± 0.00	99.58 ± 0.07	67.61 ± 0.48	0.286 ± 0.002	70.69 ± 0.08
S2	Level-2	40.00 ± 0.00	100.00 ± 0.00	67.61 ± 0.48	0.125 ± 0.000	87.35 ± 0.05
S2	Level-M-R	100.00 ± 0.00	67.61 ± 0.48	0.00 ± 0.00	0.342 ± 0.004	40.65 ± 0.06
S2	Corrected static	77.38 ± 0.16	91.66 ± 0.28	51.04 ± 0.55	0.294 ± 0.002	63.25 ± 0.26
S2	Bayesian	84.96 ± 0.24	98.59 ± 0.11	1.32 ± 0.12	0.422 ± 0.003	79.08 ± 0.18
S3	Level-0	100.00 ± 0.00	67.61 ± 0.48	0.00 ± 0.00	1.367 ± 0.016	40.65 ± 0.06
S3	Level-1	70.00 ± 0.00	99.58 ± 0.07	67.61 ± 0.48	1.706 ± 0.008	70.69 ± 0.08

S3	Level-2	40.00 $\pm$ 0.00	100.00 $\pm$ 0.00	67.61 $\pm$ 0.48	1.767 $\pm$ 0.003	87.35 $\pm$ 0.05
S3	Level-M-R	40.33 $\pm$ 0.03	99.99 $\pm$ 0.01	67.61 $\pm$ 0.48	1.766 $\pm$ 0.003	87.17 $\pm$ 0.06
S3	Corrected static	40.00 $\pm$ 0.00	100.00 $\pm$ 0.00	67.61 $\pm$ 0.48	1.767 $\pm$ 0.003	87.35 $\pm$ 0.05
S3	Bayesian	79.17 $\pm$ 0.33	99.52 $\pm$ 0.06	3.93 $\pm$ 0.20	2.042 $\pm$ 0.005	86.44 $\pm$ 0.10

Table 12 shows that the Bayesian policy improves the tested utility–resource trade-off rather than dominating every individual metric. In S1, it retains 80.40% residual energy, achieves 99.15% LDSR, and limits redundancy to 2.28%, compared with 80.25%, 78.32%, and 22.22% for Level-M-R. In S2, the policy sacrifices part of Level-M-R's no-defense energy reserve but raises LDSR from 67.61% to 98.59% and raises mean utility from 0.342 to 0.422. In S3, it retains 79.17% energy with 99.52% LDSR, whereas Level-M-R retains 40.33% with 99.99%. The Bayesian–Level-M-R utility differences are 0.266 (95% CI [0.264, 0.267]), 0.080 ([0.080, 0.081]), and 0.275 ([0.274, 0.276]); the difference is positive in all 30 paired runs in each scenario.

Fig. 5a–c visualizes the RE, LDSR, and DR columns of Table 12. Bars show 30-run means, and error bars show one standard deviation.

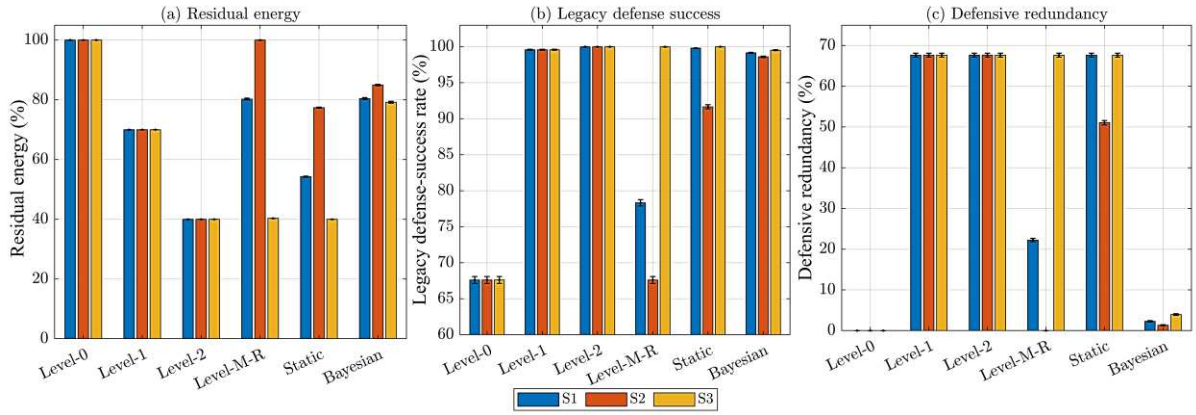


Fig. 5 Layer-A results: (a) residual energy, (b) LDSR, and (c) defensive redundancy

The graphs make the allocation pattern visible. In S1 and S3, the Bayesian rule avoids the energy cost of permanent Level-2 defense while maintaining near-complete LDSR; in S2, it gives up the no-defense energy maximum to reduce under-protection.

6.3 Controlled calendar-simulation results (Layer B)

Layer B evaluates the same decision rules on a fixed hidden-state calendar anchored to the official 23 February 2018 attack timetable while generating observations from the declared Gaussian model. It is therefore a controlled sequential simulation rather than a packet- or flow-level test. Table 13 reports the moderate-noise results.

Table 13 Layer-B results at $\sigma = 0.70$

Method	LDSR (%)	ECR (%)	DR (%)	BAR (%)	UDR (%)	Mean energy per window	Mean defender utility
Level-0	80.49	31.50	0.00	0.00	100.00	0.0000	0.778
Level-1	91.46	64.25	80.49	100.00	43.75	0.0030	0.731
Level-2	100.00	85.25	80.49	100.00	0.00	0.0060	0.580
Level-M-F	94.41 $\pm$ 2.08	70.30 $\pm$ 5.31	64.21 $\pm$ 3.86	79.77 $\pm$ 4.79	28.67 $\pm$ 10.65	0.0042 $\pm$ 0.0003	0.639 $\pm$ 0.024
Level-M-R	86.94 $\pm$ 2.42	49.28 $\pm$ 6.71	26.66 $\pm$ 4.77	33.13 $\pm$ 5.93	66.92 $\pm$ 12.40	0.0020 $\pm$ 0.0003	0.699 $\pm$ 0.032
Corrected static	95.94 $\pm$ 1.59	75.38 $\pm$ 2.71	80.49 $\pm$ 0.00	100.00 $\pm$ 0.00	20.79 $\pm$ 8.13	0.0046 $\pm$ 0.0002	0.646 $\pm$ 0.015
Bayesian	99.75 $\pm$ 0.53	84.48 $\pm$ 1.29	0.86 $\pm$ 1.12	1.07 $\pm$ 1.39	1.27 $\pm$ 2.72	0.0012 $\pm$ 0.0001	0.948 $\pm$ 0.006

Under the moderate-noise condition $\sigma = 0.70$, the Bayesian policy achieves 99.75% LDSR, 84.48% ECR, 1.07% BAR, 1.27% UDR, and mean utility 0.948. Level-M-R attains 86.94% LDSR, 49.28% ECR, 33.13% BAR, 66.92% UDR, and utility 0.699. The paired utility difference is 0.249, with a simulation-based 95% CI of [0.245, 0.253], and is positive in 300/300 replications. Level-2 retains the highest ECR (85.25%) but activates defense in every benign window and consequently has lower utility (0.580). Accordingly, the Bayesian advantage is a joint utility-and-allocation gain; Level-2 remains slightly better on ECR. To separate posterior quality from defense outcomes, Table 14 reports proper scoring rules, discrete state summaries, and attack-spell onset delay.

Table 14 Bayesian posterior quality and onset delay

Metric	Mean $\pm$ SD
Brier score	0.023 $\pm$ 0.019
Log loss	0.043 $\pm$ 0.034
State accuracy (%)	98.51 $\pm$ 1.44
Macro-F1	0.939 $\pm$ 0.081
Balanced accuracy	0.945 $\pm$ 0.079
Attack-spell onset delay (windows)	0.08 $\pm$ 0.20

At $\sigma = 0.70$, the posterior achieves macro-F1 0.939 ± 0.081 , balanced accuracy 0.945 ± 0.079 , Brier score 0.023 ± 0.019 , and log loss 0.043 ± 0.034 . The mean attack-spell onset delay to the first ordinally adequate defense is 0.08 ± 0.20 windows. These values characterize posterior performance under the declared synthetic observation model and do not constitute flow-level IDS accuracy.

Fig. 6 visualizes the utility column of Table 13. Bars show the mean and error bars show one standard deviation over 300 stochastic observation replications.

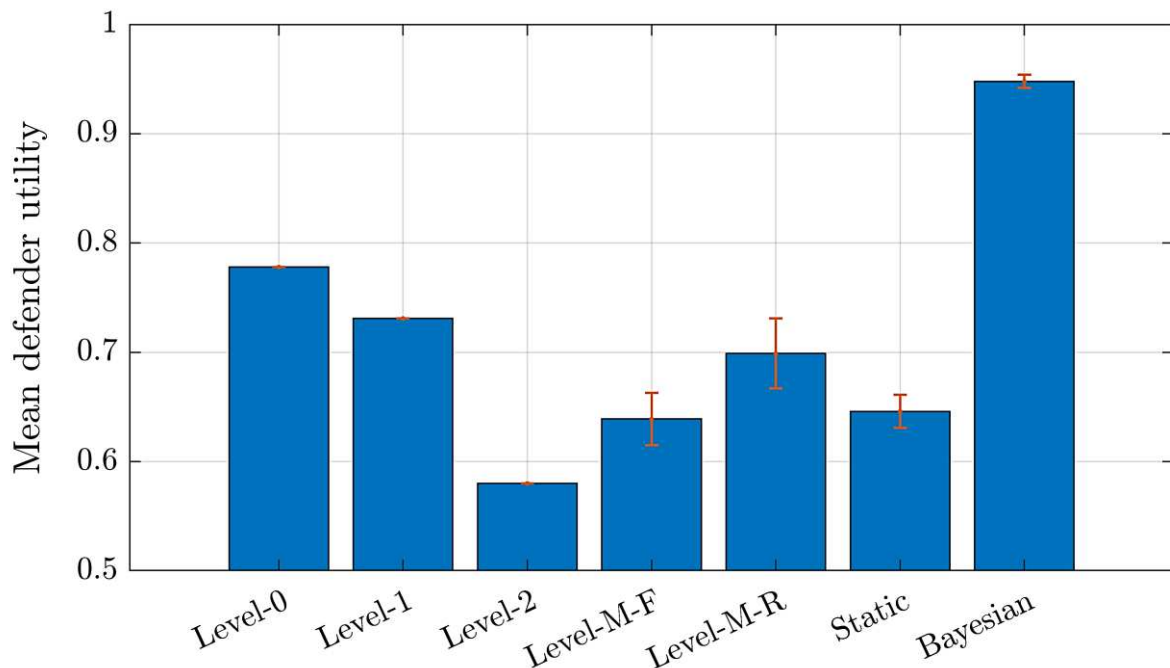


Fig. 6 Mean defender utility in Layer B ($\sigma = 0.70$)

Fig. 6 shows the highest mean utility for the Bayesian policy (0.948), followed by Level-0 (0.778) and Level-1 (0.731). The advantage reflects selective defense allocation; it should not be interpreted as higher raw containment than always-on Level-2. Because the success matrices are family-specific, Table 15 disaggregates LDSR and ECR for BF and SQLi.

Table 15 Attack-family-specific Layer-B results

Method	BF-LDSR (%)	BF-ECR (%)	SQLi-LDSR (%)	SQLi-ECR (%)
Level-0	0.00	34.00	0.00	20.67

Level-1	61.54	67.54	33.33	50.00
Level-2	100.00	87.54	100.00	75.33
Level-M-F	72.21 $\pm$ 11.86	72.73 $\pm$ 5.99	67.56 $\pm$ 27.92	59.74 $\pm$ 13.39
Level-M-R	33.03 $\pm$ 13.93	51.68 $\pm$ 7.55	33.33 $\pm$ 28.46	38.92 $\pm$ 15.64
Corrected static	81.33 $\pm$ 8.57	78.10 $\pm$ 2.91	70.00 $\pm$ 22.89	63.59 $\pm$ 7.34
Bayesian	99.00 $\pm$ 2.74	86.91 $\pm$ 1.33	97.56 $\pm$ 8.70	73.95 $\pm$ 4.12

Table 15 reports BF/SQLi ECRs of 86.91%/73.95% for the Bayesian rule and 87.54%/75.33% for Level-2. Fig. 7 plots those columns with one-standard-deviation error bars.

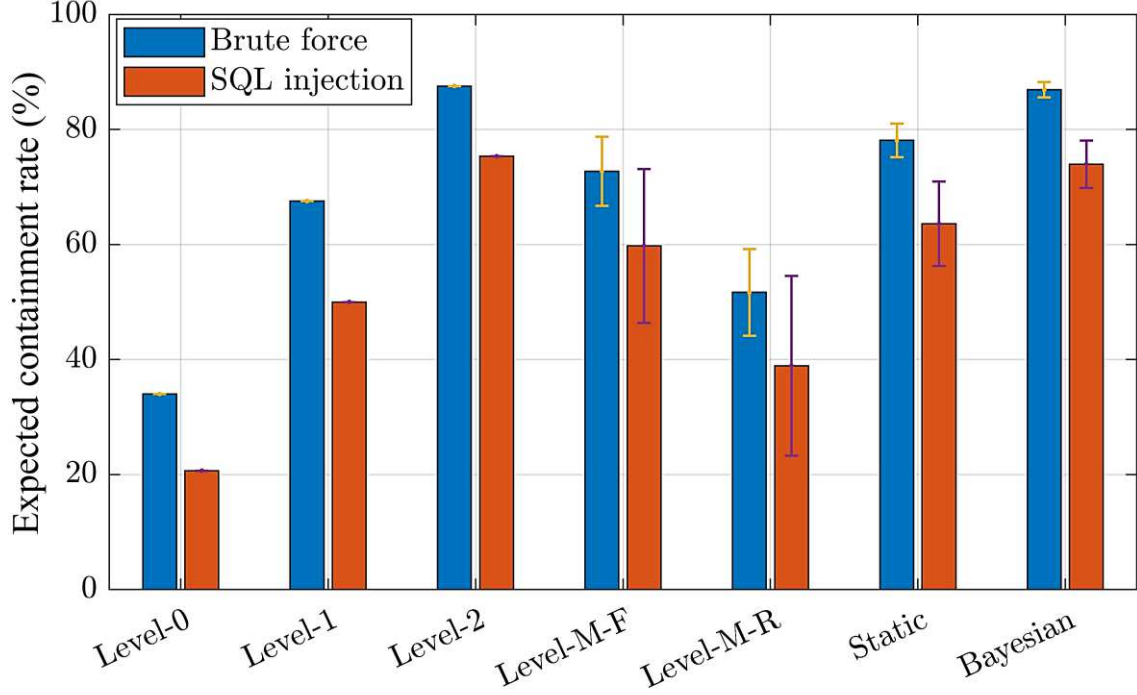


Fig. 7 Attack-family-specific ECR in Layer B ($\sigma = 0.70$)

Across both families, the Bayesian rule approaches Level-2 containment while avoiding continuous defense during benign windows.

6.4 Ablation study

To isolate the contributions of temporal prediction, attack-family separation, and Bayesian updating, Table 16 compares the full policy with three precisely defined ablations.

Table 16 Layer-B ablation results

Policy variant	LDSR (%)	ECR (%)	BAR (%)	UDR (%)	Mean defender utility
Full model	99.76 $\pm$ 0.55	84.44 $\pm$ 1.42	1.04 $\pm$ 1.25	1.23 $\pm$ 2.83	0.948 $\pm$ 0.005
Fixed prior	99.53 $\pm$ 0.75	83.77 $\pm$ 1.85	2.22 $\pm$ 1.91	2.40 $\pm$ 3.84	0.943 $\pm$ 0.008
Pooled family	99.27 $\pm$ 0.97	83.07 $\pm$ 2.43	1.82 $\pm$ 1.77	3.75 $\pm$ 4.96	0.942 $\pm$ 0.009
No Bayesian update	95.88 $\pm$ 1.69	75.30 $\pm$ 2.77	100.00 $\pm$ 0.00	21.12 $\pm$ 8.65	0.645 $\pm$ 0.016

Removing Bayesian updating produces the largest decline: utility falls from 0.948 to 0.645, ECR from 84.44% to 75.30%, and BAR rises from 1.04% to 100%. Temporal prediction and family separation provide smaller but consistent additional gains.

Because Table 16 uses the independent seed set 50000–50299, its full-model mean differs slightly from Table 13 through Monte Carlo variation rather than a change in model specification.

6.5 Sensitivity analysis

Sensitivity to observation noise, switching cost, and the declared brute-force intensity boundary is summarized in Table 17.

Table 17 Sensitivity analysis of the Bayesian policy

Sensitivity factor	Parameter value	LDSR (%)	ECR (%)	BAR (%)	UDR (%)	Mean defender utility	Additional diagnostic
Sensor noise σ	0.35	100.00	85.25	0.01	0.00	0.953	Macro-F1 1.000
Sensor noise σ	0.70	99.79	84.51	0.86	1.08	0.949	Macro-F1 0.936
Sensor noise σ	1.05	98.98	82.09	3.71	5.23	0.935	Macro-F1 0.808
Switching cost κ	0.000	99.70	84.32	1.05	1.56	0.949	Not applicable
Switching cost κ	0.015	99.69	84.29	1.03	1.60	0.948	Not applicable
Switching cost κ	0.030	99.68	84.26	0.95	1.65	0.946	Not applicable
BF edge fraction	0.20	99.78	83.39	1.03	1.10	0.943	BF1/BF2=6/7
BF edge fraction	0.30	99.78	84.52	1.09	1.12	0.948	BF1/BF2=8/5
BF edge fraction	0.40	99.78	86.75	1.12	1.15	0.958	BF1/BF2=12/1

Increasing observation noise from $\sigma = 0.35$ to 1.05 reduces macro-F1 from 1.000 to 0.808 and ECR from 85.25% to 82.09%, while BAR rises from 0.01% to 3.71%. Increasing κ from 0 to 0.03 slightly lowers BAR but raises UDR and reduces mean utility from 0.949 to 0.946. Changing the BF edge fraction alters the BF1/BF2 composition and ECR, whereas LDSR remains about 99.78%. These are sensitivity results for the declared simulation, not evidence about unmodeled traffic drift. The moderate-noise rows in Tables 13 and 17 use different fixed seed sets, which accounts for their small numerical differences.

6.6 Discussion

Family conditioning affects the stage game before the Bayesian update is applied. BF and SQLi can rank the same defense pair differently, while an equal-weight pooled matrix produces only one aggregate ranking. The equilibrium results therefore support separate payoff calibration rather than a common intensity model. The matched benchmark shows that Bayesian selection improves mean utility over Level-M-R in all three scenarios without dominating every individual metric. In S2, the policy spends energy to avoid the no-defense outcome; in S3, it accepts a small reduction in LDSR relative to Level-2 while retaining substantially more energy. Layer B provides a complementary result: at $\sigma = 0.70$, posterior macro-F1 is 0.939, ECR is 84.48%, BAR is 1.07%, and UDR is 1.27%. The ablation study identifies posterior updating as the largest contributor, with smaller gains from temporal propagation and family separation. The scope remains controlled and simulation-based. The CSE-CIC-IDS2018 timetable supplies attack timing, but the likelihood means, success matrices, and cost coefficients are analytical design parameters. The findings support the proposed decision rule under declared uncertainty; they do not establish flow-level detection accuracy, live-system containment, or finite-horizon optimal control.

7 Conclusion

This paper introduced ATA-DBD, an attack-type-aware dynamic Bayesian defense model that retains the interpretable Level-0/1/2 action scale while assigning BF and SQLi separate success matrices, losses, costs, and penalties. The resulting stage games can rank the same defense pair differently, which an equal-weight pooled model cannot preserve (Proposition 3). The family-conditioned utilities are embedded in a five-state Bayesian process that predicts and updates the latent state before selecting the lowest-index defense that maximizes current posterior utility; under the increasing-differences condition of Proposition 4, within-family escalation never lowers the selected level. Evaluating both the formula-faithful and the outcome-matched reconstructions of the published mixed strategy (Level-M-F and Level-M-R) keeps the benchmark comparison transparent where the printed strategy and the reported aggregate profile do not coincide. In the matched benchmark, LDSR is 99.15%,

98.59%, and 99.52%, with residual energy of 80.40%, 84.96%, and 79.17% across S1–S3. In Layer B, ECR is 84.48%, mean utility is 0.948, BAR is 1.07%, UDR is 1.27%, and posterior macro-F1 is 0.939. Level-2 retains slightly higher raw containment, whereas the Bayesian rule avoids continuous high-intensity defense and achieves higher utility. These results apply to the declared simulation. Attack timing comes from the public timetable, while observation means and payoff parameters remain analytical. Future work should estimate these quantities from packet or flow data and controlled service experiments, extend the state space to additional attack families, and evaluate finite-horizon or multi-defender policies.

Statements and Declarations

Competing Interests: The authors have no relevant financial or non-financial interests to disclose.

Ethics Approval and Consent to Participate: Not applicable. The study involved no human participants, animals, or personal data.

Data Availability: The official CSE-CIC-IDS2018 attack timetable used in this study is publicly available from the Canadian Institute for Cybersecurity (<https://www.unb.ca/cic/datasets/ids-2018.html>); no packet-capture or flow-level data were processed. The derived state calendars, simulation parameters, random seeds, and result tables are available from the corresponding author upon reasonable request and permission from the funding agency.

References

- Adisa SP, Akanbi CO, Ogundoyin IK (2025) Classification and analysis of cyber-attack intensities: a game theoretic approach. *J King Saud Univ Comput Inf Sci* 37:244. <https://doi.org/10.1007/s44443-025-00043-0>
- Alghawazi M, Alghazzawi D, Alarifi S (2022) Detection of SQL injection attack using machine learning techniques: a systematic literature review. *J Cybersecur Priv* 2(4):764–777. <https://doi.org/10.3390/jcp2040039>
- Aslan Ö, Aktuğ SS, Ozkan-Okay M, Yilmaz AA, Akin E (2023) A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. *Electronics* 12(6):1333. <https://doi.org/10.3390/electronics12061333>
- Attiah A, Chatterjee M, Zou CC (2018) A game theoretic approach to model cyber attack and defense strategies. In: 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018, pp 1–7. IEEE. <https://doi.org/10.1109/ICC.2018.8422719>
- Bouke MA, Abdullah A (2023) An empirical study of pattern leakage impact during data preprocessing on machine learning-based intrusion detection models reliability. *Expert Syst Appl* 230:120715. <https://doi.org/10.1016/j.eswa.2023.120715>
- Canadian Institute for Cybersecurity (n.d.) CSE-CIC-IDS2018 on AWS. University of New Brunswick. <https://www.unb.ca/cic/datasets/ids-2018.html>. Accessed 17 Sep 2026
- Catillo M, Pecchia A, Rak M, Villano U (2021) Demystifying the role of public intrusion datasets: a replication study of DoS network traffic data. *Comput Secur* 108:102341. <https://doi.org/10.1016/j.cose.2021.102341>
- Chen J, Hengartner U, Khan H (2024) MRAAC: a multi-stage risk-aware adaptive authentication and access control framework for Android. *ACM Trans Priv Secur* 27(2):17. <https://doi.org/10.1145/3648372>
- Cho JH, Sharma DP, Alavizadeh H, Yoon S, Ben-Asher N, Moore TJ, Kim DS, Lim H, Nelson FF (2020) Toward proactive, adaptive defense: a survey on moving target defense. *IEEE Commun Surv Tutor* 22(1):709–745. <https://doi.org/10.1109/COMST.2019.2963791>
- Crespo-Martínez IS, Campazas-Vega A, Guerrero-Higueras ÁM, Riego-DelCastillo V, Álvarez-Aparicio C, Fernández-Llamas C (2023) SQL injection attack detection in network flow data. *Comput Secur* 127:103093. <https://doi.org/10.1016/j.cose.2023.103093>
- Dahiya A, Gupta BB (2021) A reputation score policy and Bayesian game theory based incentivized mechanism for DDoS attacks mitigation and cyber defense. *Future Gener Comput Syst* 117:193–204. <https://doi.org/10.1016/j.future.2020.11.027>
- Do CT, Tran NH, Hong C, Kamhoua CA, Kwiat KA, Blasch E, Ren S, Pissinou N, Iyengar SS (2017) Game theory for cyber security and privacy. *ACM Comput Surv* 50(2):30. <https://doi.org/10.1145/3057268>
- Harsanyi JC (1967) Games with incomplete information played by "Bayesian" players, I–III. Part I. The basic model. *Manag Sci* 14(3):159–182. <https://doi.org/10.1287/mnsc.14.3.159>

- Ho E, Rajagopalan A, Skvortsov A, Arulampalam S, Piraveenan M (2022) Game theory in defence applications: a review. *Sensors* 22(3):1032. <https://doi.org/10.3390/s22031032>
- Iqbal A, Gunn LJ, Guo M, Babar MA, Abbott D (2019) Game theoretical modelling of network/cyber security. *IEEE Access* 7:154167-154179. <https://doi.org/10.1109/ACCESS.2019.2948356>
- Kanta A, Coisel I, Scanlon M (2024) A comprehensive evaluation on the benefits of context based password cracking for digital forensics. *J Inf Secur Appl* 84:103809. <https://doi.org/10.1016/j.jisa.2024.103809>
- Kasım Ö (2021) An ensemble classification-based approach to detect attack level of SQL injections. *J Inf Secur Appl* 59:102852. <https://doi.org/10.1016/j.jisa.2021.102852>
- Khalid MNA, Al-Kadhimi AA, Singh MM (2023) Recent developments in game-theory approaches for the detection and defense against advanced persistent threats (APTs): a systematic review. *Mathematics* 11(6):1353. <https://doi.org/10.3390/math11061353>
- Lee D, Kim D, Ahn MK, Lee S (2024) Bayesian Stackelberg game approach for cyber mission impact assessment. *ICT Express* 10(2):299-305. <https://doi.org/10.1016/j.ict.2023.11.003>
- Leevy JL, Khoshgoftaar TM (2020) A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 big data. *J Big Data* 7:104. <https://doi.org/10.1186/s40537-020-00382-x>
- Li Q, Wang F, Wang J, Li W (2019) LSTM-based SQL injection detection method for intelligent transportation system. *IEEE Trans Veh Technol* 68(5):4182-4191. <https://doi.org/10.1109/TVT.2019.2893675>
- Liu X, Zhang J, Zhu P, Tan Q, Yin W (2021) Quantitative cyber-physical security analysis methodology for industrial control systems based on incomplete information Bayesian game. *Comput Secur* 102:102138. <https://doi.org/10.1016/j.cose.2020.102138>
- Lu J, Liu A, Dong F, Gu F, Gama J, Zhang G (2019) Learning under concept drift: a review. *IEEE Trans Knowl Data Eng* 31(12):2346-2363. <https://doi.org/10.1109/TKDE.2018.2876857>
- Manshaei MH, Zhu Q, Alpcan T, Başar T, Hubaux JP (2013) Game theory meets network security and privacy. *ACM Comput Surv* 45(3):25. <https://doi.org/10.1145/2480741.2480742>
- McWhirter PR, Kifayat K, Shi Q, Askwith B (2018) SQL injection attack classification through the feature extraction of SQL query strings using a gap-weighted string subsequence kernel. *J Inf Secur Appl* 40:199-216. <https://doi.org/10.1016/j.jisa.2018.04.001>
- Milgrom P, Shannon C (1994) Monotone comparative statics. *Econometrica* 62(1):157-180. <https://doi.org/10.2307/2951479>
- Murray H, Malone D (2023) Adaptive password guessing: learning language, nationality and dataset source. *Ann Telecommun* 78(7-8):385-400. <https://doi.org/10.1007/s12243-023-00969-4>
- Nash JF (1951) Non-cooperative games. *Ann Math* 54(2):286-295. <https://doi.org/10.2307/1969529>
- Paul A, Sharma V, Olukoya O (2024) SQL injection attack: detection, prioritization & prevention. *J Inf Secur Appl* 85:103871. <https://doi.org/10.1016/j.jisa.2024.103871>
- Pawlick J, Colbert E, Zhu Q (2019) A game-theoretic taxonomy and survey of defensive deception for cybersecurity and privacy. *ACM Comput Surv* 52(4):82. <https://doi.org/10.1145/3337772>
- Perwej Y, Abbas SQ, Dixit JP, Akhtar N, Jaiswal AK (2021) A systematic literature review on the cyber security. *Int J Sci Res Manag* 9(12):669-710. <https://doi.org/10.18535/ijssrm/v9i12.ec04>
- Sengupta S, Chowdhary A, Sabur A, Alshamrani A, Huang D, Kambhampati S (2020) A survey of moving target defenses for network security. *IEEE Commun Surv Tutor* 22(3):1909-1941. <https://doi.org/10.1109/COMST.2020.2982955>
- Sepczuk M, Kotulski Z (2018) A new risk-based authentication management model oriented on user's experience. *Comput Secur* 73:17-33. <https://doi.org/10.1016/j.cose.2017.10.002>
- Silva Filho T, Song H, Perello-Nieto M, Santos-Rodriguez R, Kull M, Flach PA (2023) Classifier calibration: a survey on how to assess and improve predicted class probabilities. *Mach Learn* 112(9):3211-3260. <https://doi.org/10.1007/s10994-023-06336-7>
- Tang P, Qiu W, Huang Z, Lian H, Liu G (2020) Detection of SQL injection based on artificial neural network. *Knowl-Based Syst* 190:105528. <https://doi.org/10.1016/j.knosys.2020.105528>

- Wang X, Yan Z, Zhang R, Zhang P (2021) Attacks and defenses in user authentication systems: a survey. *J Netw Comput Appl* 188:103080. <https://doi.org/10.1016/j.jnca.2021.103080>
- Xie X, Ren C, Fu Y, Xu J, Guo J (2019) SQL injection detection for web applications based on elastic-pooling CNN. *IEEE Access* 7:151475-151481. <https://doi.org/10.1109/ACCESS.2019.2947527>
- Yao P, Jiang Z, Yan B, Yang Q, Wang W (2024) Bayesian and stochastic game joint approach for cross-layer optimal defensive decision-making in industrial cyber-physical systems. *Inf Sci* 662:120216. <https://doi.org/10.1016/j.ins.2024.120216>
- Zarreh A, Wan H, Lee Y, Saygin C, Al Janahi R (2019) Risk assessment for cyber security of manufacturing systems: a game theory approach. *Procedia Manuf* 38:605-612. <https://doi.org/10.1016/j.promfg.2020.01.077>
- Zhang W, Li Y, Li X, Shao M, Mi Y, Zhang H, Zhi G (2022) Deep neural network-based SQL injection detection method. *Secur Commun Netw* 2022:4836289. <https://doi.org/10.1155/2022/4836289>
- Zhang Y, Malacaria P (2021) Bayesian Stackelberg games for cyber-security decision support. *Decis Support Syst* 148:113599. <https://doi.org/10.1016/j.dss.2021.113599>
- Zheng Y, Li Z, Xu X, Zhao Q (2022) Dynamic defenses in cyber security: techniques, methods and challenges. *Digit Commun Netw* 8(4):422-435. <https://doi.org/10.1016/j.dcan.2021.07.006>
- Zuech R, Hancock J, Khoshgoftaar TM (2021) Investigating rarity in web attacks with ensemble learners. *J Big Data* 8:71. <https://doi.org/10.1186/s40537-021-00462-6>