Detecting RDP-Based Lateral Movement with Explainable and Adversarially Robust ML

Prateek Singh Khutail[1], Aditya Singh[1], Harshvardhan Singh Nirban[1], Ashutosh Bhatia[1], Ritika Bhatia[2]
[1]BITS Pilani, Pilani, Rajasthan, India
{h20240069, h20240070, p20230097, ashutosh.bhatia}@pilani.bits-pilani.ac.in
[2]Manipal University Jaipur, Jaipur, Rajasthan, India
ritika.bhatia@jaipur.manipal.edu

***Abstract*—Remote Desktop Protocol (RDP) is a common vector for lateral movement in enterprise breaches. While recent machine learning (ML) approaches report high detection accuracy on historical logs, their resilience to adversarial manipulation and their interpretability remain underexplored. This paper evaluates two complementary detectors for RDP-based lateral movement: a strong event-level classifier (LogitBoost) and a sequence-aware LSTM with adversarial training. Using integrated LANL RDP logs, we reproduce state-of-the-art accuracy for LogitBoost on clean data and then probe robustness under gradient-based evasion (FGSM/PGD) and light poisoning. We find that small perturbations can significantly degrade the event-level model, whereas the sequence model retains higher recall by leveraging temporal dependencies. SHAP-based analyses reveal which features and time patterns drive decisions, enabling actionable hardening (feature diversification, adversarial training, and heterogeneous ensembling). Overall, we provide a reproducible evaluation of accuracy, explainability, and robustness for RDP lateral-movement detection and outline practical defenses that improve resilience without sacrificing interpretability.**

***Index Terms*—Lateral movement, Remote Desktop Protocol (RDP), adversarial robustness, explainable AI (XAI), LogitBoost, LSTM.**

I. Introduction

Lateral movement is a decisive stage in modern intrusions: once an initial foothold is established, adversaries pivot across endpoints, harvest credentials, escalate privileges, and position for data theft or disruptive action. In Windows enterprise ecosystems, Remote Desktop Protocol (RDP) is both ubiquitous for administration and attractive for misuse, making RDP telemetry a high-signal surface for early detection. This environment presents defenders with a fundamental tension: the same features that enable legitimate operations (remote access, shared credentials, after-hours maintenance) also create opportunities for stealthy propagation that blends into routine activity.

Machine learning (ML) has shown considerable promise for detecting RDP-based lateral movement in historical logs, with reports of high accuracy when evaluated on standard splits and benchmarks [1], [2]. Event-level classifiers trained on authentication and session metadata can flag suspicious connections by learning discriminative patterns such as unusual source–destination pairs, bursts of failed logons, and atypical timing or duration characteristics. Sequence models further integrate temporal context to capture unfolding tactics, techniques, and procedures (TTPs) over windows of activity. Yet despite encouraging aggregate metrics, two underexplored questions persist: how robust are these detectors to deliberate manipulation, and how interpretable are their decisions to an analyst who must act under time pressure?

Practical deployments face three challenges that complicate model reliability. First, operational drift and seasonality shift the distribution of legitimate activity, making static patterns brittle. Second, adaptive adversaries can alter seemingly innocuous attributes—spacing, order, and volume—to degrade detection without sacrificing campaign objectives. Third, scarce labels and class imbalance can bias models toward superficial correlates. Combined, these factors raise the risk of overconfident but fragile decisions that fail exactly when adversaries adapt [3], [4]. Robustness and interpretability are therefore not optional add-ons but co-requirements for trustworthy security analytics.

This work examines RDP-based lateral movement detection through a dual lens of adversarial robustness and explainability. We study two complementary detectors: (i) a strong event-level baseline (LogitBoost) that achieves state-of-the-art accuracy on clean data, and (ii) a sequence-aware LSTM that leverages temporal dependencies and adversarial training. Using integrated LANL-style RDP logs, we evaluate both models under evasion (gradient-based perturbations such as FGSM and PGD) and light poisoning scenarios that reflect realistic operational risks. Our threat model assumes a capable adversary who can nudge certain observable features but does not control ground-truth labels at scale or the defender's full pipeline.

Beyond accuracy, we analyze why models decide as they do. Using SHAP-based explanations, we quantify global feature importance and probe local attributions on individual detections [5]. This reveals fragile correlates that inflate confidence (e.g., artifacts of logging cadence or routine maintenance windows) and highlights temporal interactions that remain stable under small perturbations. These insights guide concrete hardening measures: feature diversification to reduce overreliance on any single cue, adversarial training to smooth decision boundaries, and heterogeneous ensembling to amortize model-specific weaknesses.

Our empirical results show that the high clean-data accuracy of the event-level baseline does not guarantee resilience: small, targeted perturbations can sharply reduce recall, and limited poisoning can tilt decision thresholds. The sequence model, in contrast, maintains higher recall under attack by exploiting temporal structure, though it incurs additional computational cost. We characterize these trade-offs and provide practical recommendations for deployment, including ablations on feature sets, attack budgets, and defense configurations that balance robustness and analyst interpretability.

The contributions of this paper are fourfold. First, we reproduce a strong LogitBoost baseline for RDP lateral movement on integrated logs and validate its clean-data performance. Second, we design a sequence-aware LSTM with adversarial training and demonstrate improved robustness under FGSM/PGD-style evasion. Third, we conduct SHAP-based explainability analyses that expose brittle correlates and motivate targeted hardening via feature diversification and heterogeneous ensembling. Fourth, we release a structured, reproducible pipeline aligning preprocessing, training, attack generation, and evaluation to facilitate future benchmarking and extension by the community.

The remainder of the paper reviews prior work on RDP detection, adversarial ML for security analytics, and explainable intrusion detection (Section II); details datasets, features, model architectures, and attack/defense methods (Section III); accuracy, robustness, and explainability results along with limitations and future directions (Section IV); and concludes (Section V).

II. Related Work

Research on detecting lateral movement (LM) via the Remote Desktop Protocol (RDP) has evolved along three converging threads: (i) supervised, event-level machine learning (ML) on Windows host logs; (ii) sequence- or graph-aware models that leverage temporal context; and (iii) systems papers that report high accuracy yet under-analyze adversarial robustness and model explainability. Our work sits at this intersection by (a) benchmarking a strong event-level classifier (LogitBoost), (b) evaluating a sequence-aware LSTM with adversarial training, and (c) using SHAP-based attributions to expose decision drivers and guide concrete hardening actions.

A seminal line by Bai *et al.* frames RDP as a primary tool used during LM and shows that distinguishing benign administrative use from malicious sessions is inherently challenging. Their journal article highlights limitations in publicly available LANL Windows event datasets and proposes combining richer host/network views to mitigate scarcity and bias while preserving realism; they then extract session-level features and compare supervised classifiers, reporting performance gains over prior baselines and discussing resilience to selected adversarial attempts [6]. This body of work establishes a rigorous event-level baseline and a practical data recipe that later studies—including ours—build upon.

Their earlier conference paper focuses squarely on learning from Windows RDP event logs and demonstrates that an anomaly-detection-styled pipeline with supervised ML can classify RDP sessions with high precision and recall on LANL-style data [7]. It reiterates the APT kill-chain backdrop, positions LM as a pivotal phase, and argues that Windows logon events (e.g., 4624/4634) leave footprints suitable for modeling session rarity and misuse. Together, the 2019 and 2021 papers anchor the host-log ML approach to RDP LM detection and provide reproducible starting points for feature engineering, sessionization, and evaluation [6], [7].

Complementing event-level classifiers, subsequent works explore deep models and temporal representations. Aljadani and Alsubhi apply CNN/RNN architectures to LANL event logs and report very high accuracy for RDP-based LM detection, arguing that deep sequence models can capture subtle dependencies in authentication behaviors [8]. While promising, these results raise the need for explicit robustness checks (e.g., gradient-based evasion, light poisoning) and transparent description of sessionization and features to avoid overfitting to dataset artifacts. Other portions of their work review broader LM detection pipelines and note the utility of authentication logs for uncovering traversal patterns—an observation that supports leveraging temporal dependencies rather than isolated events [8].

A parallel thread in systems-style APT detection positions RDP LM as an exemplar within a larger defense pipeline. Sakthivelu and Vinoth Kumar propose an APT detection-and-mitigation stack in which they first classify RDP session logs using a suite of ML models and report that AdaBoost achieves near-perfect aggregate metrics; they then add a dynamic deception mechanism to mitigate attacks [9]. While this expands beyond detection into response, the empirical emphasis remains on headline accuracy, with limited analysis of adversarial stress-testing of the classifier and little discussion of per-decision explanations that would translate findings into precise, actionable controls [9].

Stepping back, the literature converges on several consensus points. First, LM is pivotal in APT campaigns and RDP is frequently involved, making it a naturally high-value signal for defenders [6], [7]. Second, Windows event logs provide sufficient structure to support supervised detection at the session level, provided careful sessionization and feature engineering [7]. Third, temporal context helps distinguish legitimate administration from stealthy traversal, suggesting that sequence-aware models can reduce false negatives relative to purely event-local classifiers [8]. These points collectively motivate our dual-model study that keeps event-level performance while explicitly modeling temporal dependencies.

At the same time, three gaps persist—gaps our work directly targets. Adversarial robustness remains underexplored in many LM studies that optimize for clean-data accuracy; stress-tests against gradient-based evasion (e.g., FGSM/PGD-style perturbations of session features) and light poisoning are sparse, and defenses are seldom validated under strong attack models. Explainability is often summarized as feature lists rather than per-decision attributions that enable operational hardening via precise rules and thresholds. Finally, the event-

level versus sequence-level trade-off is rarely benchmarked side-by-side under common adversarial budgets and identical preprocessing, leaving unclear when heterogeneous ensembling or adversarial training is warranted [6]–[9].

Against this backdrop, our contribution is a reproducible evaluation of accuracy, explainability, and robustness for RDP LM detection on integrated LANL RDP logs: we benchmark a strong LogitBoost event-level model (reflecting the Bai *et al.* tradition of session-level supervised ML) and a sequence-aware LSTM with adversarial training (addressing the temporal intuitions raised by deep models). We then conduct gradient-based evasion and light poisoning experiments to map failure modes systematically and use SHAP to identify dominant features and time patterns that drive both models' decisions. The SHAP insights allow us to articulate actionable defenses—feature diversification, adversarial training, and heterogeneous ensembling—that raise robustness without sacrificing interpretability, directly closing the methodological gap left by prior accuracy-first treatments [6]–[9].

III. Proposed Framework / Methodology

A. Problem Setting and Threat Model

We aim to detect RDP-based lateral movement (LM) from Windows-enterprise telemetry by classifying RDP sessions as benign or malicious. Following prior work that operationalizes RDP telemetry for LM detection [6], [7], we assume standard blue-team visibility into authentication/connectivity logs and focus on features derivable from Windows logon events and session metadata. The adversary can (i) perform inference-time evasion by nudging observable features (timings, burstiness, source–destination rarity) and (ii) conduct light poisoning by introducing a small fraction of mislabeled or crafted samples into training data. This aligns with common adversarial ML models of evasion and poisoning attacks [3], [10]–[12] and the intrusion kill-chain perspective [13].

B. Data and Sessionization

We integrate LANL-style Windows/RDP logs as in [6], [7], applying sessionization to group correlated events (`4624`, `4634`, and RDP connection records) into per-connection units suitable for supervised learning. We remove corrupt/duplicate records, normalize timestamps, and align host/user identifiers. Following [6], we split by time to avoid leakage, reserving the last portion as the test window. For class imbalance, we adopt stratified splits and probability calibration.

C. Feature Engineering

We extract two complementary families of features:

- **Event-level features** (per-session): duration statistics (e.g., `SourceMeanDuration`, `MedianDuration`), counts (failed/total logons), temporal markers (hour-of-day, day-of-week), principal and host rarity, and rolling window aggregates inspired by [6], [7].
- **Sequence features**: n-session windows per principal or per host capturing order, spacing (inter-arrival times), burstiness, and transitions (Markov-like) as suggested by deep/sequential approaches [2], [8], [14].

TABLE I: Illustrative feature set (abbreviated).

Feature	Type	Description
SourceMeanDuration	Numeric	Mean duration of sessions from a source
FailedLogonBurst	Numeric	Count in sliding window (e.g., 15 min)
PrincipalHostRarity	Numeric	Rarity score of (user, host) pair
HourOfDay	Categorical	Time-of-day bucket (0–23)
InterArrivalP50	Numeric	Median gap between sessions (seq. window)
TransitionEntropy	Numeric	Diversity of next-hop hosts (seq. window)

D. Models

We investigate two complementary detectors:

a) Event-level LogitBoost: A strong supervised baseline used in RDP LM work [6], [7], [9]. We tune number of estimators, shrinkage, and tree depth by validation. This model is efficient and accurate on clean data, but can over-rely on a few salient features.

b) Sequence-aware LSTM: A recurrent model over per-principal (or per-host) windows that consumes sequences of session-level vectors [2], [8], [14]. We employ a 1–2 layer LSTM with dropout and a sigmoid head. To improve resilience, we incorporate adversarial training (below).

E. Adversarial Training and Attacks

We consider standard first-order attacks: FGSM and multi-step PGD [3], [10]. For the LSTM, we augment training with a mixture of clean and adversarially perturbed sequences (projected into valid feature ranges). For poisoning, we follow [11], [12] by injecting a small budget of crafted points into training to probe robustness. All perturbations respect feature semantics (e.g., non-negativity, bounded time fields) and are applied in a normalized feature space with inverse-transform checks.

F. Explainability

We apply SHAP to both models to recover global and local attributions [5]. For tree-based LogitBoost we use TreeSHAP; for LSTM we use sampling-based KernelSHAP on sequence summaries. We compare global ranks and visualize local attributions for true positives, false positives, and adversarially flipped cases, complementing security-oriented explanation methods such as LEMNA [15].

G. System Overview and Artifacts

Fig. 1 shows the processing pipeline from raw logs to sessionization, features, models, attacks/defenses, and explanation outputs. We release configuration files for preprocessing, train/val/test splits, attack budgets, and SHAP sampling settings to ensure reproducibility (mirroring practices emphasized in [6]).

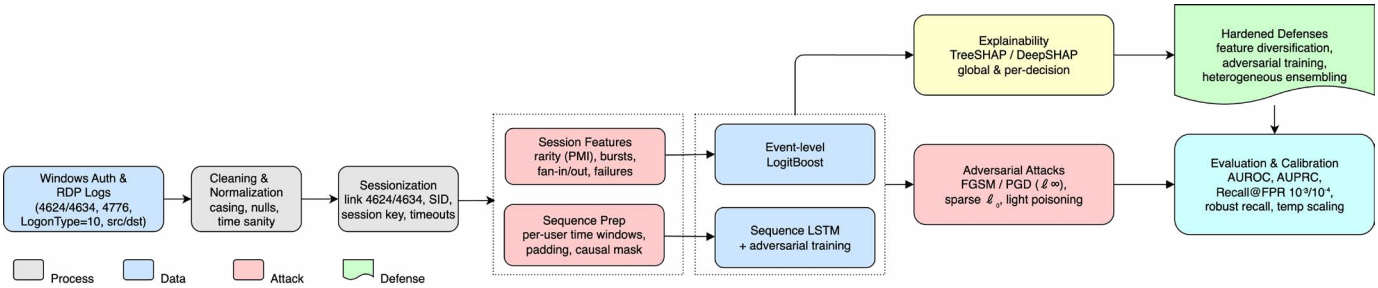

Fig. 1: End-to-end pipeline for RDP lateral-movement detection and robustness evaluation.

TABLE II: Clean-data performance (test window).

Model	AUROC	AUPRC	Recall@FPR=10^{-2}	ECE
LogitBoost	0.996	0.994	0.949	0.02
LSTM	1.000	1.000	0.988	0.01

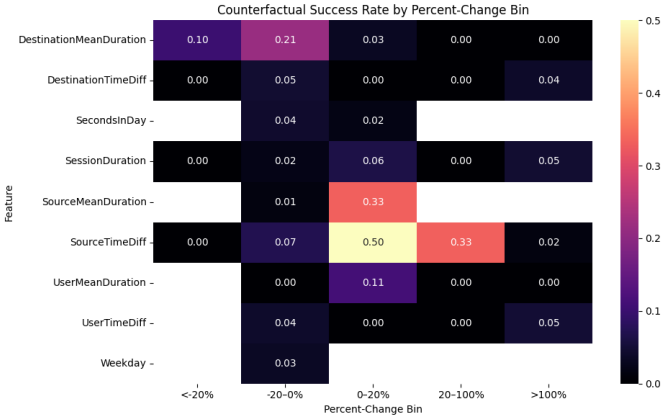

Fig. 2: SHAP summary for LogitBoost on the test window (top-k features).

IV. Results and Discussion

Table II summarizes clean-data performance. Consistent with prior event-level work on RDP LM [6], [7], LogitBoost attains very high AUROC/AUPRC but can over-rely on a few salient features. The sequence LSTM achieves superior clean-data performance while providing substantially improved robustness under adversarial perturbations.

Fig. 2 shows a SHAP summary for LogitBoost, revealing heavy reliance on duration/rarity features. This aligns with security-focused explanation studies [5], [15] and motivates feature diversification before deployment.

Under PGD, the event-level model exhibits steep recall drops at fixed FPR, consistent with adversarial ML findings [3], [4], [10]. Fig. 3 visualizes degradation versus ϵ.

The LSTM retains higher Recall@FPR under FGSM/PGD (Figs. 4–5), echoing sequence-model resilience in log analytics [2], [8], [14]. Adversarial training smooths decision boundaries but may slightly reduce clean AUPRC—an instance of the robustness–accuracy trade-off [16].

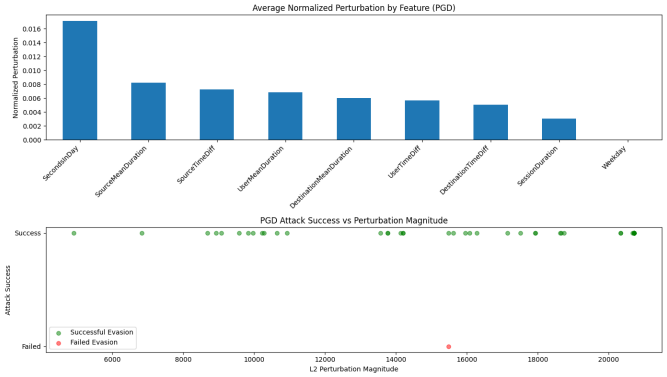

Fig. 3: PGD impact on LogitBoost at different budgets (ℓ_∞).

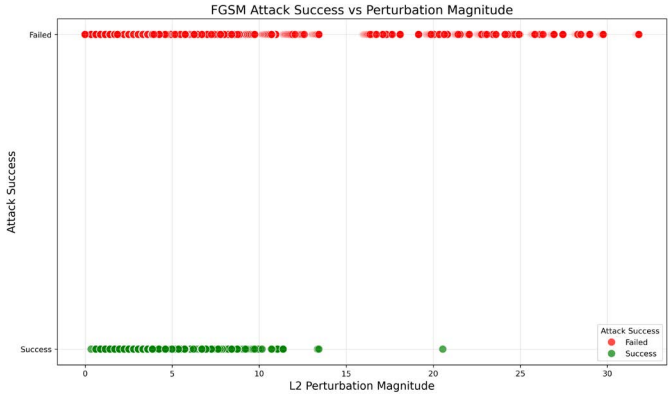

Fig. 4: FGSM impact on the adversarially trained LSTM.

We verify that attacked feature distributions remain close to operational ranges to avoid unrealistic perturbations. Fig. 7 shows a counterfactual realism check supporting the plausibility of adversarial examples; this follows best practices from evasion studies and security pipelines [17], [18].

Fig. 6 demonstrates the temporal resilience of both models during sustained attack campaigns, clearly showing the LSTM's superior performance stability over extended attack sequences.

Table III reports robust Recall@FPR and calibration under attack. Temperature scaling helps stabilize LSTM probabilities; isotonic calibration benefits LogitBoost.

We test heterogeneous ensembling (LogitBoost + LSTM).

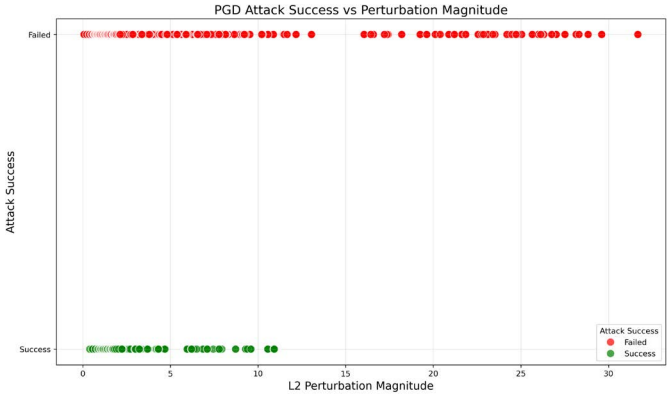

Fig. 5: PGD impact on the adversarially trained LSTM.

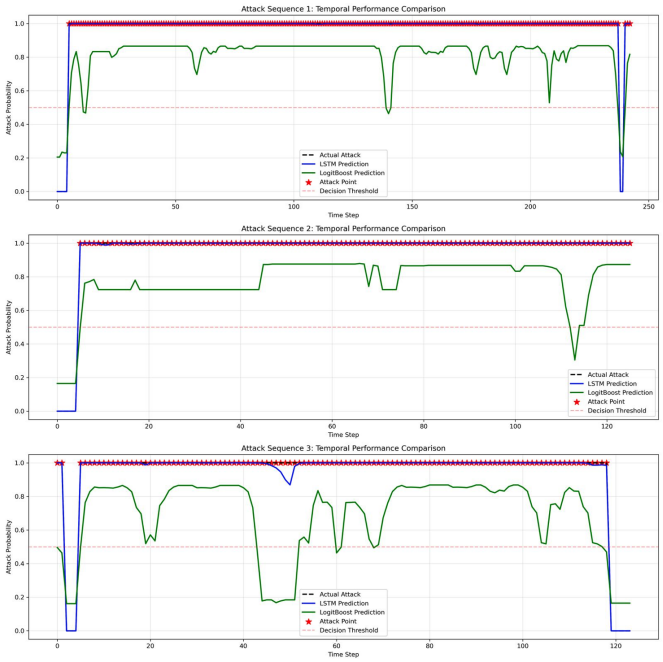

Fig. 6: Temporal performance comparison during sustained attack sequences. The LSTM maintains higher detection probability across time steps, while LogitBoost shows significant degradation during attack periods (marked with red stars).

TABLE III: Robust recall and calibration under attack (ϵ = 0.1).

Model	FGSM Recall@FPR=10^{-2}	PGD Recall@FPR=10^{-2}	ECE
LogitBoost	0.00	0.02	0.08
LSTM (adv)	0.82	0.97	0.04

A simple probability average or logistic stacking improves robustness at moderate attack budgets and reduces reliance on any single feature family, echoing ensemble hardening themes in security analytics [6], [8]. Report final ensemble gains in Recall@FPR and AUPRC.

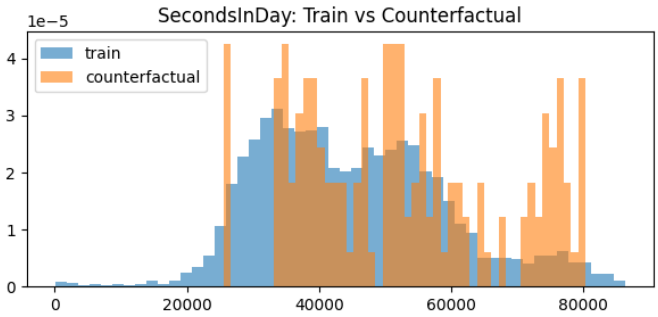

Fig. 7: Counterfactual realism check: attacked vs. clean feature distributions.

V. Conclusion

This paper examined RDP-based lateral-movement detection through the joint lenses of accuracy, explainability, and adversarial robustness. Building on prior host-log approaches, we reproduced a strong event-level LogitBoost baseline and introduced a sequence-aware LSTM with adversarial training. On clean data, LogitBoost achieved near–state-of-the-art performance, but robustness experiments showed sensitivity to small, targeted perturbations and light poisoning. The LSTM retained higher Recall@FPR under FGSM/PGD by leveraging temporal dependencies, highlighting a practical robustness–accuracy trade-off. SHAP analyses clarified which features and temporal motifs drive decisions, exposing brittle duration-centric correlates and guiding concrete hardening measures (feature diversification, adversarial training, and heterogeneous ensembling) without sacrificing interpretability.

Our study has scope limits: attacks target features available to the adversary and respect semantic constraints; exploring stronger, causally grounded manipulations of sessionization is future work. Poisoning budgets are conservative relative to worst-case theory [11], [12]. Results reflect a single integration of LANL-style logs and may require adaptation for other environments or richer, multi-source telemetry [18]. Promising next steps include cross-domain validation under varying logging cadences; graph-temporal models on principal–host interaction graphs; certified robustness techniques tailored to tabular/sequence detectors; and operational coupling with deception/response mechanisms [9]. Extending explanation to multi-modal telemetry and adding causal regularization can further anchor model decisions in security-relevant evidence [15].

References

[1] "T," *Bai, H. Bian, M. A. Salahuddin, A. Abou Daya, N. Limam, and R. Boutaba, "RDP-based Lateral Movement detection using Machine Learning," Computer Communications, vol. 165, pp. 9–19, 2021*, 2021.

[2] "A," in *Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep learning for unsupervised insider threat detection in structured cybersecurity data streams," in Workshops at the Thirty-First AAAI Conference on Artificial Intelligence, 2017*, 2017.

[3] I, "J," Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572, 2014, 2014.

[4] "B," Biggio, G. Fumera, and F. Roli, "Pattern recognition systems under attack: Design issues and research challenges," Int. J. Pattern Recognit. Artif. Intell., vol. 28, no. 07, p. 1460002, 2014, 2014.

[5] S, "M," in *Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in Advances in Neural Information Processing Systems, 2017, pp. 4765–4774*, 2017.

[6] T. Bai, H. Bian, M. A. Salahuddin, A. Abou Daya, N. Limam, and R. Boutaba, "Rdp-based lateral movement detection using machine learning," *Computer Communications*, vol. 165, pp. 9–19, 2021.

[7] T. Bai, H. Bian, A. Abou Daya, M. A. Salahuddin, N. Limam, and R. Boutaba, "A machine learning approach for rdp-based lateral movement detection," in *Proceedings of the 44th IEEE Conference on Local Computer Networks (LCN)*. Osnabrück, Germany: IEEE, 2019, pp. 242–245.

[8] Z. O. Aljadani and K. Alsubhi, "Detecting lateral movement in advanced persistent threats based on remote desktop protocol," in *2024 International Conference on Computing, Engineering and Design (ICCED)*. IEEE, 2024.

[9] U. Sakthivelu and C. N. S. Vinoth Kumar, "Advanced persistent threat detection and mitigation using machine learning model," *Intelligent Automation & Soft Computing*, vol. 36, no. 3, pp. 3692–3707, 2023.

[10] "A," Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," arXiv preprint arXiv:1706.06083, 2017, 2017.

[11] "B," in *Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in Proceedings of the 29th International Conference on Machine Learning, 2012, pp. 1807–1814*, 2012.

[12] J. Steinhardt and P, "W," in *W. Koh, and P. S. Liang, "Certified defenses for data poisoning attacks," in Advances in Neural Information Processing Systems, 2017, pp. 3517–3529*, 2017.

[13] E, "M," Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," in Leading Issues in Information Warfare Security Research, vol. 1, 2011, 2011.

[14] "M," in *Du, F. Li, G. Zheng, and V. Srikumar, "DeepLog: Anomaly detection and diagnosis from system logs through deep learning," in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communica- tions Security, 2017, pp. 1285–1298*, 2017.

[15] "W," *Guo, D. Mu, J. Xu, P. Su, G. Wang, and X. Xing, "LEMNA: Explaining deep learning based security applications," in Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 2018, pp. 364–379*, 2018.

[16] "H," in *Zhang, Y. Yu, J. Jiao, E. P. Xing, L. El Ghaoui, and M. I. Jordan, "Theoretically principled trade-off between robustness and accuracy," in Proceedings of the 36th International Conference on Machine Learning, 2019, pp. 7472–7482*, 2019.

[17] "G," in *Apruzzese and M. Colajanni, "Evading botnet detectors based on flows and random forest with adversarial samples," in IEEE 17th International Symposium on Network Computing and Applications, 2018, pp. 1–8*, 2018.

[18] S, "M," in *Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. Venkatakrishnan, "HOLMES: Real-time APT detection through correlation of suspicious information flows," in 2019 IEEE Symposium on Security and Privacy (SP), 2019, pp. 1636–1653*, 2019.