Adaptive Network Defense: A Comparative Study of Security-Focused vs. Availability-Focused Reinforcement Learning Agents

Noah Weaver, Kadri Mufti, Loic Thibault Josue Martins

September 15, 2025

1 Problem Statement

The cybersecurity landscape has a surge in both the frequency and sophistication of cyberattacks in recent years. Traditional network defense systems rely on static, rule-based approaches that struggle to adapt to evolving threats and novel attack vectors. These static systems often result in either overly restrictive policies that disrupt legitimate network traffic or permissive configurations that fail to detect sophisticated attacks.

While the last few years have seen an explosion of interest in autonomous cyber defense agents based on deep reinforcement learning [1], with at least 32 cyber simulator environments already developed for training such agents, a critical gap remains in understanding how different RL philosophies perform in real-world network defense scenarios.

Network administrators face a fundamental trade-off: aggressive defensive measures may successfully block attacks but can also disrupt legitimate users, while conservative approaches maintain service quality but may miss sophisticated threats. This challenge is particularly acute as attackers continuously adapt their strategies to evade detection systems.

Research Question: How do security-focused versus availability-focused reinforcement learning agents compare in their ability to defend networks against diverse attack patterns while maintaining different operational priorities?

Input: Real-time network traffic features (bandwidth utilization, connection patterns, packet characteristics)

Output: Adaptive defensive actions (rate limiting, traffic filtering, connection monitoring)

Why is this problem important and challenging? Modern networks must balance competing objectives of security and availability under constantly evolving threat landscapes. The problem is challenging because: (1) attack patterns continuously evolve to evade detection, (2) legitimate traffic patterns vary significantly across different network environments, (3) defensive actions have delayed and cascading effects that are difficult to predict, and (4) false positives in security can have severe business consequences.

2 Proposed Approach

We will develop and compare two distinct RL defense philosophies within a controlled network simulation environment using Mininet [4], a mature network emulation platform that creates virtual networks with realistic behavior.

Security-Focused Agent: Prioritizes attack prevention with tolerance for false positives. Uses reward functions that heavily weight attack prevention over service disruption (e.g., $R = 5 \times \text{attacks\_blocked} - 1 \times \text{false\_positives}$, though exact weights will be tuned empirically) and employs quick restriction policies with evidence-based restoration.

Availability-Focused Agent: Prioritizes service continuity while maintaining reasonable security. Uses reward functions that heavily penalize service disruption (e.g., $R = 2 \times \text{attacks\_blocked} - 10 \times \text{service\_disruption}$, subject to experimental optimization) and requires high confidence thresholds before implementing restrictions.

Both agents will use reinforcement learning algorithms (likely Deep Q-Networks or Policy Gradient methods, to be determined based on preliminary experiments) with identical network architectures, differing primarily in reward functions and action selection policies.

Software-Defined Networking (SDN) is a network architecture approach that separates the network control plane (decision-making logic) from the data plane (packet forwarding), enabling centralized, programmable network management through software controllers rather than distributed hardware-based configurations.

2.1 Technical Implementation

- **Environment:** Mininet-based network emulation with SDN controllers that allow centralized, programmable network management
- **State Space:** Network metrics (throughput, latency, connection counts, traffic patterns)
- **Action Space:** Graduated defensive measures (rate limiting percentages, selective blocking, monitoring rules)
- **Attack Generation:** Deterministic attack patterns including DDoS, port scanning, and lateral movement simulations

3 Baseline Methods

1. **Multi-Agent RL Framework:** CyberBattleSim approach for training joint red-blue agents using PPO and A2C algorithms in simulated cyber environments [1]
2. **Supervised-RL Hybrid Method:** Generic blue agent training that combines reinforcement learning with supervised learning using variational autoencoders [2]
3. **Game Theory-Based RL:** Defense-A3C algorithm that integrates deep reinforcement learning with dynamic game theory for network defense decision-making [3]

4 Feasibility Assessment

This project is feasible within the course timeframe using existing tools and frameworks. Mininet provides a mature platform for network emulation, and standard RL libraries (PyTorch, OpenAI Gym) offer robust implementations. The deterministic attack generation ensures reproducible experiments, while the two-agent comparison keeps the scope manageable. We will leverage existing Mininet tutorials and SDN controller implementations, focusing our development effort on the RL agents and evaluation framework.

Alternative simulation environments such as CyGIL [5] could be considered if Mininet proves insufficient for our experimental requirements, though Mininet's maturity and extensive documentation make it the preferred choice.

5 Timeline

Weeks 1-2: Environment setup (Mininet installation, basic topology creation, attack pattern implementation)
Weeks 3-4: RL agent development and baseline implementation
Weeks 5-6: Training and initial evaluation
Weeks 7-8: Comprehensive evaluation across attack scenarios
Weeks 9-10: Analysis, ablation studies, and report writing

6 Expected Contributions

This research will provide:

- Quantitative analysis of security vs. availability trade-offs in RL-based network defense
- Practical guidance for selecting appropriate RL defense philosophies based on network requirements
- Demonstration of RL adaptability compared to static defense mechanisms
- Reproducible experimental framework for future network security RL research

The study addresses the practical deployment question: "Which RL approach should network administrators choose for their specific environment?" rather than merely demonstrating that RL can work for cybersecurity applications.

References

[1] Thomas Kunz, Christian Fisher, James La Novara-Gsell, Christopher Nguyen, Li Li. A Multiagent CyberBattleSim for RL Cyber Operation Agents. *arXiv preprint arXiv:2304.11052*, 2023. https://arxiv.org/pdf/2304.11052

[2] Muhammad Omer Farooq, Thomas Kunz. Combining Supervised and Reinforcement Learning to Build a Generic Defensive Cyber Agent. *Journal of Cybersecurity and Privacy*, vol. 5, no. 2, pp. 23, 2025. https://www.mdpi.com/2624-800X/5/2/23

[3] Huang Wanwei, Yuan Bo, Wang Sunan, Ding Yi, Li Yuhua. Network Defense Decision-Making Based on Deep Reinforcement Learning and Dynamic Game Theory. *China Communications*, vol. 21, no. 9, pp. 262-275, 2024. https://ieeexplore.ieee.org/abstract/document/10700942

[4] Faris Keti, Shavan Askar. Emulation of Software Defined Networks Using Mininet in Different Simulation Environments. *2015 6th International Conference on Intelligent Systems, Modelling and Simulation*, pp. 205-210, 2015. https://ieeexplore.ieee.org/document/7311238

[5] Jalil Taghia, Maria Naess, Anton Nylund, Linus Gisslén. CyGIL: A Cyber Gym for Training Autonomous Agents over Emulated Network Systems. *arXiv preprint arXiv:2109.03331*, 2021. https://arxiv.org/abs/2109.03331