"LLM Enhanced Agents for Cybersecurity Applications"
Agents, Multi-Agent Systems and Reinforcement Learning

MSc Artificial Intelligence, 2025

Michael Rice

March 3, 2025

1 Introduction to Agents and Multi-Agent Systems

1.1 A Brief History and Overview of Agents and Multi-Agent Systems

In order to give a comprehensive, yet succinct overview of Agents and Multi-Agent Systems, it is important to first grasp the concept of what an 'Agent' is. Agents, as defined by Luck and D'Inverno, are software-based systems that are 'situated in some environment and are capable of autonomous action in that environment in order to meet their design objectives' [11]. The theory outlining what we have come to know as Agents and Multi-Agent Systems was in development long before the concepts themselves were formalized. Notable contributions to this space include Turing's early ground-breaking introduction to the idea of machine intelligence and learning [16] as well as Genesereth and Ketchpel's much later work as an effort to formalize the theoretical underpinnings of modern Agents and Multi-Agent Systems as 'goal-oriented, autonomous and social entities' [7].

Multi-Agent Systems (MAS), in turn, are systems that are built using Agents as the most basic, fundamental building block. They must define methods and protocols for agents to interact with one another, as well as the environment in which they are to do so. [5]. Agents can have a variety of different reasons for interaction, which fall under the umbrella terms of cooperation, co-ordination and negotiation. Cooperation refers to any processes that allow agents to work together to achieve common or compatible goals [17], while coordination alludes to any agent based efforts to ensure efficient and effective task execution. Finally, negotiation is the concept of agents being able to reach an agreement in the presence of conflicting goals and preferences.

1.2 Modern-Day Agents and Multi-Agent Systems

In recent years, Agents and Multi-Agent Systems have seen a resurgence in popularity, largely due to the advent of Artificial Intelligence, which has seen an improvement autonomous decision-making techniques. Two large factors in this upturn in interest have been the development of Large Language Models (LLMs) as well as the rise of Deep Reinforcement Learning (RL) as viable methods for training agents to perform complex tasks.

As stated by Dong et al.[4], AI agents that deploy LLMs as their core reasoning framework have the potential to emulate 'human-level perception, cognition, and behaviors' [4]. This unprecedented level of multimodal comprehension, generative abilities and reasoning capabilities potentially present a fascinating and potentially revolutionary step towards the eventual long-term goal of artificial general intelligence. As for Deep Reinforcement Learning, Oroojlooy and Hajinezhad's 2023 paper outlines the recent successes of Deep RL in other domains, such as chip design or inventory optimization, as well as how these successes could be translated to the Multi-Agent Reinforcement Learning (MARL) domain, in an attempt to improve the state-of-the-art performance in this area [14].

2 Behaviors of Agent-Based Systems vs Non Agent-Based Systems

The variation in behavior between an agent-based system and a more traditional, non agent-based system can be sizeable. Agents provide a multitude of possibly valuable attributes to an implementation, including but not limited to: autonomy, interactivity, scalability and decentralization.

Firstly, as previously stated, agents provide a degree of autonomy that is not possible in static systems. Autonomous agents or 'Agentic AI' are defined by Acharya et al. as an autonomous and adaptive form of artificial intelligence capable of independent decision-making, reasoning, and goal-directed behavior within dynamic environments [2]. This ability for agent-based systems to control their own environments allows them to behave non-deterministically, which can result in more reactive and dynamic systems.

Secondly, agent-based systems are naturally well-suited for scaling because of their modular architectures. In comparison to traditional systems, which are often monolithic and difficult to alter, agents can be added to or removed from their systems with relative ease. This increased system customization allows for easier and faster design processes. As shown in the implementation of MULTITASK [9], by Kusne and McDonald, the use of agents allows for flexible resizing and reconfiguring of the system, which in this case, was used as a tool to scale their materials discovery tool to demand.

Furthermore, agents are inherently interactive, another feature differentiating then from non-agent based systems. Interactions in agent-based systems are defined by a number of key categories. Agent-to-Agent, Agent-to-Environment, Agent-to-User are all types of interaction which can each potentially result in emergent behavior. Inter-agent and agent-to-environment interactions are crucial to system functionality, as communication between all parties allows for such behavior to occur. Emergent behaviors in an agent-based system refer to the 'kind of collective behaviors that are not explicitly programmed but are the result of local interactions among individual components in the system' [19]. This is a key feature of agent-based systems that allows for complex, adaptive and dynamic system behavior.

Finally, another behavior of agent-based systems that is not present in their counterpart systems is decentralization. Decentralization is the concept that organizations, (of agents in this context), are not under control by a singular 'central' entity, but instead are connected by a 'possibly time-varying and sparse communication network, which serves as the channel for agents to exchange information'. [20]. This is another key feature of agent-based systems that again allows for further improvements in system flexibility and reactivity.

3 My Chosen Problem Domain

3.1 Introduction

The problem domain that has been chosen to explore in the context of Agents and MAS is that of Cybersecurity, specifically targeting a project completed as part of an internship in the Global Security department of Hewlett Packard Enterprise for improvement. The project was a proof-of-concept tool that would be used to more effectively hire new Cybersecurity Analysts by practically assessing their technical abilities in a sandbox environment during the interview process. I believe that an agent-based approach, including the utilization of LLMs, could prove beneficial to the overall value of the tool.

In order to define the ways in which agents could potentially improve the original implementation, it is important to first understand the original system. To briefly sum up the operation of the tool, when the candidate sat in front of it, they would be presented with a series of dashboards to monitor the status of networks, servers and other relevant systems. The candidate would then be informed that they should watch this dashboard, and when an alert appeared, they would be required to take action to mitigate the incoming threat.

3.2 How Are Agents Being Used in This Domain

3.2.1 LLM Based Agents

LLM based agents have recently become one of the hottest topics in the fields of Agents, with LLM's increased reasoning capabilities being applied in a variety of different subdomains. Two such interesting and relevant applications that exhibit this are: 'ExpeL: LLMAgents Are Experiential Learners' by Zhao et al. and He et al.'s 'LLM-Based Multi-Agent Systems for Software Engineering: Literature Review, Vision and the Road Ahead'. These two implementations do not tackle the Cybersecurity domain directly, however, they provide an introduction into the potential of LLM based agents, prior to the examination of implementations at the convergence of agents and Cybersecurity.

Zhao et al. focus on the issue of non-parametric agent-based learning, in an attempt to tackle the issue that most prominent LLM's parametric weights have remained proprietary [21]. They approach this problem by introducing the ExpeL agent, which is capable of gathering experiences and extracting knowledge from a range of diverse training tasks [21]. Thus, when the agent is called upon at inference, it can recall and apply the knowledge it possesses from previous experiences to improve its decision-making abilities.

Further, the work of He et al. in their 2025 IEEE publication [8] focuses on the integration of LLMs and MAS in an attempt to address some of the most complex challenges in software engineering. The authors posit that the benefits of combining such techniques are many, including advancing autonomous problem-solving abilities, improving fault tolerance and introducing enhanced scalability to such systems, all in the pursuit a system capable of handling a variety of software engineering tasks such as Requirement Engineering, Code Generation, Quality Assurance and Maintenance [8].

3.2.2 Agents in Cybersecurity

In the domain of Cybersecurity, agents, especially those combined with LLMs, have the capability to revolutionize the industry. Over the past few years, an increasing number of papers have been published exploring this area, some of which will be discussed below.

A first relevant and interesting application of LLM empowered agents in the domain of offensive cybersecurity is detailed by Ning et al. in their publication 'CheatAgent: Attacking LLM-Empowered Recommender Systems via LLM Agent' [12]. They reason that LLMs provide 'unprecedented opportunities to serve as attack agents' [12] due to their thus unseen, human-like decision-making skills. The presented attack strategy details the utilization of RL agents to conduct poisoning attacks on the target recommender system. This is due to the fact that in the experimental environment, the attackers had no access to the models or parameters behind the target system, only the inputs they gave to the system, along with the corresponding outputs. Further, the authors detail a self-reflection policy optimization method which further enhanced the attack's effectiveness [12]. Performance of CheatAgent was tested on three real-world datasets, with the results showing that an effective attack strategy was designed, while also simultaneously demonstrating the vulnerabilities of such LLM based recommender systems [12].

Another fascinating work at the intersection of agents and Cybersecurity is proposed in Carrasco et al.'s 'CYBERSHIELD: A Competitive Simulation Environment for Training AI in Cybersecurity' [3]. This work focuses on the simultaneous training of both offensive and defensive Cybersecurity agents by pitting them against one another in a strategic battle where they attempt to outsmart one another. This environment is optimized for the exploration of novel attack and defense strategies, which are evaluated and perpetuated through the system, depending on their success, using RL algorithms. The defining feature of this publication is its simultaneous training of both offensive and defensive agents, a novel approach to autonomous agentic Cybersecurity.

In an effort to present a well-rounded discussion, a number of experiments using LLM agents solely as defensive tools shall also be discussed. This is necessary in the cybersecurity domain largely due to the inadequacy of current defenses in dealing with the volume and complexity of modern threats. One of these pieces of research, undertaken by Oesch et al. of the Oakridge National Laboratory is titled 'Towards a High Fidelity Training Environment for Autonomous Cyber Defense Agents' [13]. The paper outlines the development of a novel training environment for autonomous cyber defense agents, known as 'CyberWheel' [13]. The motivation behind CyberWheel was to enable flexible training of Autonomous Cyber Defense Agents through easier alteration of reward function and observation space. CyberWheel, as an example, helped with the training of a blue (defensive) agent, which learned to deploy a decoy server when under attack by a red (offensive) agent [13] allowing the defensive structure behind the blue agent to detect the red agent, as well as leaving the system's actual servers unharmed.

Another engaging publication in the agent-based defense subspace of Cybersecurity, published by Loevenich et al., is titled 'Training Autonomous Cyber Defense Agents: Challenges And Opportunities in Military Networks' [10]. This paper draws on techniques such as Deep Reinforcement Learning (DRL), LLMs and a rule-based system to training hybrid AI models for Autonomous Cyber Defense (ACD) agents in military networks. The authors' combination of these technologies resulted in the creation of high performing agents, capable of complex decision-making and adaptive learning in highly unpredictable and limited 'tactical edge environments' [10]. This paper also discusses the integration and evaluation of the resulting trained agents using a simulated environment, inspired by NATO Protected Core Networking Environment [10]. Three types of agents are used to perform the experiment: red agents, as disruptors of the network environment, green agents, to perform operations with rewards and penalties generated by the maintenance of network abilities and blue agents, to defend a designated environment zone. The conclusions from this experiment were that the authors plan to improve the system by leveraging recent LLM improvements by fine-tuning numerous LLMs, each for a specific task, to further enhance the decision-making capabilities of the agents.

3.3 Potential System Enhancements

Altering the previously mentioned system to incorporate agents and their associated benefits could significantly improve overall system performance. The original tool could only operate via the explicit control of an operator or through a pre-written, deterministic script/schedule. These scripts would contain commands to influence the sandbox environment, as well as delays, to control the flow of the examination. This approach provided plenty of customizable options to the hiring manager when designing and executing scripts during interviews, however a more dynamic and realistic environment via the use of agents could improve the usability and effectiveness of the tool. Below, a number of potential agent-based enhancements will be suggested as improvements.

The integration of a multi-agent system (MAS) would enable a hands-off approach to deploying this tool, where specialized agents independently manage different aspects of the environment. For instance, a set of offensive agents, each being LLMs fine-tuned using Deep RL (with frozen policies to ensure earlier tested candidates don't experience easier tests than those tested later in the hiring process) are each trained to exploit a specific vulnerability. These would be used as the 'Red Team' portion of the system. The red team could potentially operate alongside a team of 'Scoring Agents', which would be responsible for evaluating candidate responses.

In order to ensure a cohesive and structured examination, a coordination mechanism, such as token passing could be used. At the start of the experiment, a randomly selected offensive agent would receive the token, initiating a coordinated attack of it's choosing. Once complete, the token would transfer to the scoring agents, who would produce a multi-faceted assessment of the candidate's solution. This process would repeat for a predefined number of cycles, or until the candidate failed a task.

While the objectives differ, the coordination principles outlined by Flammini et al. in 'Preventing Deadlocks for Multi-Agent Pickup and Delivery in Dynamic Environments' [6] are relevant to this approach. Their study addresses the challenge of ensuring agents are both environmentally aware and capable of influencing their surroundings, using token passing to prevent deadlocks in dynamic environments. Similarly, in the context of this Cybersecurity tool, token passing would ensure controlled execution and structured interaction between offensive and scoring agents.

The implementation of a team of 'Scoring Agents' as mentioned above could serve a variety of purposes for the overall system, such as dynamic difficulty adjustment, candidate feedback etc. Initially, dynamic difficulty adjustment would ensure that the candidate is consistently challenged, providing a more accurate assessment of their skills and abilities. Should the candidate score well on preliminary tasks, this would indicate potential for an uptake in difficulty. Candidates could be scored on their task performance by measuring response time and accuracy as well as task-specific metrics. Additionally, the scoring agents could generate detailed natural language reports on the candidate's performance, highlighting strengths and areas for improvement, which would be invaluable to hiring managers.

3.4 Future Impact of Agent-Based Approaches

As a conclusion to this report, it is evident that agents and agent-based systems will continue to make significant contributions both to the larger field of AI, as well as to more niche domains like Cybersecurity. Whether or not these developments come at the hands of LLMs, is it abundantly clear that such agents will most often require some form of human-like decision-making in order to make inroads into their application domains.

In the overall context of cybersecurity, the most substantial impact of agent-based systems is predicted to be Autonomous Threat Detection and Response. As previously mentioned, modern systems and professionals are ill-equipped to defend against the volume and sophistication of modern threats, however, with the strides made in agentic techniques, as well as the advancement of LLMs, a system that can autonomously detect and respond to threats is certainly within theoretical reach. Research into this sector is already well underway, as previously detailed, with more targeted research producing literature such as Knack and Burke's 'Autonomous Cyber Defense: Authorizing bounds for autonomous agents' [1] as well as Wei et al.'s 'Offline Reinforcement Learning for Autonomous Cyber Defense Agents' [18], which both discuss, to varying degrees, novel methods and implementations to design and create autonomous defensive agents. Wei et al. in particular, propose a fascinating approach to the problem area, detailing the use of offline reinforcement learning to train agents to defend against Advanced Persistent Threats (APTs) [18]. However, effective and generalizable autonomy in this sector remains goal, as the authors detail a need to periodically retrain agents to account for changing behaviors in the adversaries [18].

However, such advancement will not come without challenge, as outlined by Roshanaei et al. [15], who list three main 'Collaboration and Integration' challenges in Multi-Agent Cybersecurity: Data Sharing, Threat Intelligence Syncing and Response Harmonization [15]. Firstly, they propose the establishment of standardized protocols and legal frameworks [15] as a solution to the Data Sharing challenge. However, achieving widespread adoption across major entities could prove majorly challenging. Secondly, the literature identifies unified threat intelligence platforms as the most effective way to synchronize threat intelligence, but their development and maintenance would also require substantial resources and Collaboration, making implementation realistically difficult. Lastly, cross-agency training is suggested as the optimal approach for harmonizing threat response, though it also demands significant resources.

References

[1] Autonomous Cyber Defence Phase II. https://cetas.turing.ac.uk/publications/autonomous-cyber-defence-autonomous-agents.

[2] Deepak Bhaskar Acharya, Karthigeyan Kuppan, and B. Divya. Agentic AI: Autonomous Intelligence for Complex Goals—A Comprehensive Survey. *IEEE Access*, 13:18912–18936, 2025.

[3] José Álvaro Fernández Carrasco, Iñigo Amonarriz Pagola, Raúl Orduna Urrutia, and Rodrigo Román. CYBERSHIELD: A Competitive Simulation Environment for Training AI in Cybersecurity. In *2024 11th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, pages 11–18, September 2024.

[4] Xiaofei Dong, Xueqiang Zhang, Weixin Bu, Dan Zhang, and Feng Cao. A Survey of LLM-based Agents: Theories, Technologies, Applications and Suggestions. In *2024 3rd International Conference on Artificial Intelligence, Internet of Things and Cloud Computing Technology (AIoTC)*, pages 407–413, September 2024.

[5] Ali Dorri, Salil S. Kanhere, and Raja Jurdak. Multi-Agent Systems: A Survey. *IEEE Access*, 6:28573–28593, 2018.

[6] Benedetta Flammini. Preventing Deadlocks for Multi-Agent Pickup and Delivery in Dynamic Environments. *New Zealand*, 2024.

[7] Michael R. Genesereth and Steven P. Ketchpel. Software agents. *Commun. ACM*, 37(7):48–ff., July 1994.

[8] Junda He, Christoph Treude, and David Lo. LLM-Based Multi-Agent Systems for Software Engineering: Literature Review, Vision and the Road Ahead. *ACM Trans. Softw. Eng. Methodol.*, January 2025.

[9] A. Gilad Kusne and Austin McDannald. Scalable multi-agent lab framework for lab optimization. *Matter*, 6(6):1880–1893, June 2023.

[10] Johannes F. Loevenich, Erik Adler, Adrien Bécue, Alexander Velazquez, Konrad Wrona, Vasil Boshnakov, Jerry Falkcrona, Nils Nordbotten, Olwen L. Worthington, Juha Röning, Roberto Rigolin, and F. Lopes. Training Autonomous Cyber Defense Agents: Challenges & Opportunities in Military Networks. In *MILCOM 2024 - 2024 IEEE Military Communications Conference (MILCOM)*, pages 158–163, October 2024.

[11] Michael Luck and Mark d'Inverno. A Conceptual Framework for Agent Definition and Development. *The Computer Journal*, 44(1):1–20, January 2001.

[12] Liang-bo Ning, Shijie Wang, Wenqi Fan, Qing Li, Xin Xu, Hao Chen, and Feiran Huang. CheatAgent: Attacking LLM-Empowered Recommender Systems via LLM Agent. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2284–2295, Barcelona Spain, August 2024. ACM.

[13] Sean Oesch, Amul Chaulagain, Brian Weber, Matthew Dixson, Amir Sadovnik, Benjamin Roberson, Cory Watson, and Phillipe Austria. Towards a High Fidelity Training Environment for Autonomous Cyber Defense Agents. In *Proceedings of the 17th Cyber Security Experimentation and Test Workshop*, pages 91–99, Philadelphia PA USA, August 2024. ACM.

[14] Afshin Oroojlooy and Davood Hajinezhad. A review of cooperative multi-agent deep reinforcement learning. *Applied Intelligence*, 53(11):13677–13722, June 2023.

[15] Maryam Roshanaei, Mahir R. Khan, and Natalie N. Sylvester. Enhancing Cybersecurity through AI and ML: Strategies, Challenges, and Future Directions. *Journal of Information Security*, 15(3):320–339, May 2024.

[16] A. M. TURING. I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX(236):433–460, October 1950.

[17] Junyang Wang, Haiyang Xu, Haitao Jia, Xi Zhang, Ming Yan, Weizhou Shen, Ji Zhang, Fei Huang, and Jitao Sang. Mobile-Agent-v2: Mobile Device Operation Assistant with Effective Navigation via Multi-Agent Collaboration. *Advances in Neural Information Processing Systems*, 37:2686–2710, December 2024.

[18] Alexander Wei, David Bierbrauer, Emily Nack, John Pavlik, and Nathaniel Bastian. Offline Reinforcement Learning for Autonomous Cyber Defense Agents. In *Proceedings of the Winter Simulation Conference*, WSC '24, pages 1978–1989, Orlando, Florida, USA, February 2025. IEEE Press.

[19] Shuo Yang, Dilini Samarasinghe, Anupama Arukgoda, Shadi Abpeikar, Erandi Lakshika, and Michael Barlow. Automatic Recognition of Collective Emergent Behaviors Using Behavioral Metrics. *IEEE Access*, 11:89077–89092, 2023.

[20] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully Decentralized Multi-Agent Reinforcement Learning with Networked Agents. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5872–5881. PMLR, July 2018.

[21] Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM Agents Are Experiential Learners. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19632–19642, March 2024.