Introducing Threat Detectors in the Meta Attack Language

Pontus Johnson, Mathias Ekstedt, Nikolaos Kakouros
KTH Royal Institute of Technology
Stockholm, Sweden
pontusj@kth.se

***Abstract*—In the realm of cybersecurity, the ability to simulate attacks accurately constitutes an opportunity for the development of effective intrusion detection systems (IDS). The Meta Attack Language (MAL) has emerged as a tool for modeling complex attack scenarios, offering a structured approach to representing the multifaceted nature of cyber threats. In this paper, we explore the possibilities to generate realistic training data for machine learning (ML)-based IDS from MAL simulations. The paper introduces an innovative approach to enrich MAL simulations with event logging simulation capabilities, enabling the generation of detailed sequences of computer network events. This advancement allows for the creation of extensive datasets that closely mimic real-world network activities, facilitating the training of ML models to predict the state of each attack step in an attack graph with heightened accuracy. By integrating event detectors with specific attack steps and establishing information relations within the MAL framework, we pave the way for a more nuanced and effective simulation of cyberattacks, enhancing the potential for early detection and mitigation of threats in computer networks.**

1. Introduction

The increasing sophistication of cyber threats necessitates the development of advanced intrusion detection systems capable of identifying and mitigating attacks in their nascent stages. Traditional approaches to generating training data for such systems often fall short in terms of scalability and realism, limiting the effectiveness of ML-based IDS. Together with [1], this paper addresses these limitations by proposing the use of MAL-based attack simulations [2] [3] to generate unbounded training data that reflects the complexities of real-world network environments. The data comprises sequences of logged events, each associated with a label indicating the status of specific attack steps within the MAL attack graph. This novel approach not only enhances the realism and applicability of the training data but also introduces the concept of event logging within MAL simulations, a significant leap forward in the simulation of cyberattacks.

2. Background and Related Work

2.1. MAL

The Meta Attack Language (MAL) is a domain-specific language designed to model cyberattack scenarios systematically. It enables the creation of attack graphs from a specification of a system's architecture and configuration that represent the relationships and dependencies among various attack vectors. MAL's structured approach facilitates the detailed modeling of cyber threats, making it a valuable tool for researchers and cybersecurity professionals.

2.2. Intrusion Detection

Intrusion detection systems are critical components of network security infrastructures, designed to identify unauthorized access or anomalous behavior within computer systems and networks. The evolution of IDS from signature-based to anomaly-based systems, and the increasing incorporation of ML techniques, highlights the ongoing need for high-quality, realistic training data to enhance detection accuracy.

3. Approach

In addressing the challenge of generating realistic training data for machine learning-based intrusion detection systems, this paper proposes a novel extension to the Meta Attack Language (MAL). By integrating the concept of "detectors" into MAL, we aim to bridge the gap between simulated cyberattacks and the generation of corresponding event logs that mirror real-world intrusion detection scenarios.

Detectors are conceptualized as entities bound to specific attack steps within a MAL model. These entities are designed to trigger upon the compromise of their associated attack steps, simulating the detection of an attack by generating logs. The simulation of detection and logging introduces a probabilistic element, reflecting the real-world accuracy and limitations of network monitoring tools. Furthermore, each detector not only triggers based on the attack step but is also capable of establishing information relations with other objects in the instance model. These relations define the

scope and detail of the data logged, encompassing a range of information from user identities to device characteristics, depending on the nature of the MAL language logic with system assets and attack steps as well as the configuration of the detector.

This approach allows for a dynamic and granular simulation of cyberattack detection, closely mirroring the complexities and uncertainties inherent in real-world network environments. The flexibility of the proposed system enables the simulation of a wide range of logging mechanisms. Through this method, we facilitate the generation of a rich dataset comprising detailed sequences of logged events, each associated with a specific context within the simulated cyberattack scenario. This dataset serves as a valuable resource for training and evaluating ML-based intrusion detection systems, offering insights into the detection and mitigation of cyber threats.

4. Example

To exemplify the practical application of our proposed approach, consider the simulation of an attacker attempting to list the virtual machines within a cloud environment project. This common reconnaissance activity is typically logged by cloud service providers as part of their monitoring and security protocols. Within our enhanced MAL framework, this scenario is represented by an attack step which we will call `list_vms`, associated with a *Project* asset. The `list_vms` attack step is directly linked to a detector designed to simulate the logging of this event.

The detector associated with `list_vms` is configured to capture and log specific information relating to the attack, such as the account used to perform the action and the IP address from which the request originated. This is achieved through information relations established between the detector and other assets in the MAL instance model, reflecting the complex web of interactions and dependencies typical in cyberattack scenarios.

In our example, the `list_vms` attack step may be accessible through multiple *accounts* and from various *IP addresses*, that would be assets in the model. It is only upon the simulated compromise of the `list_vms` step that the specific *account* and *IP address* involved in the attack are determined. This determination is facilitated by instance model navigation over assets and their relations, enabling the identification of the compromised account and the originating IP address at the moment the attack step is triggered.

This dynamic mechanism allows for the generation of log entries that accurately reflect the specifics of the simulated attack, including the relationships between different entities within the model. By capturing this detailed information, the generated logs provide a rich dataset for training ML-based intrusion detection systems, offering a realistic representation of how such systems might encounter and respond to cyber threats in real-world scenarios.

5. Conclusion

The integration of event logging into MAL-based attack simulations represents a significant advancement in the field of cybersecurity research and intrusion detection system development. By introducing detectors capable of generating detailed event logs in response to specific attack steps, this approach enhances the realism and applicability of simulated intrusion detection scenarios. The generated datasets, rich in detailed observations of simulated cyberattacks, offer unprecedented opportunities for training and evaluating machine learning-based intrusion detection systems. Through this innovative extension to MAL, we contribute to the ongoing efforts to improve the accuracy and effectiveness of cyber threat detection and response mechanisms, ultimately enhancing the security posture of computer networks and systems.

References

[1] Pontus Johnson and Mathias Ekstedt. Towards a graph neural network-based approach for estimating hidden states in cyber attack simulations. *arXiv preprint arXiv:2312.05666*, 2023.

[2] Pontus Johnson, Robert Lagerström, and Mathias Ekstedt. A meta language for threat modeling and attack simulations. In *Proceedings of the 13th International Conference on Availability, Reliability and Security*, ARES 2018, New York, NY, USA, 2018.

[3] Wojciech Wideł, Simon Hacks, Mathias Ekstedt, Pontus Johnson, and Robert Lagerström. The meta attack language-a formal description. *Computers & Security*, 130:103284, 2023.