MM-ATTACKG: A MULTIMODAL APPROACH TO ATTACK GRAPH CONSTRUCTION WITH LARGE LANGUAGE MODELS

Yongheng Zhang College of Electronic Engineering National University of Defense Technology China zhangyongheng@nudt.edu.cn

Yunshan Ma Singapore Management University Singapore Singapore ysma@smu.edu.sg

Yingxiao Guan College of Electronic Engineering National University of Defense Technology China guanyingxiao23@nudt.edu.cn

Yuliang Lu College of Electronic Engineering National University of Defense Technology China luyuliang@nudt.edu.cn Xinyun Zhao College of Electronic Engineering National University of Defense Technology China zhaoxinyun66@nudt.edu.cn

> Haokai Ma National University of Singapore Singapore haokai.ma@u.nus.edu

Guozheng Yang* College of Electronic Engineering National University of Defense Technology China yangguozheng17@nudt.edu.cn

Xiang Wang University of Science and Technology of China China xiangwang@ustc.edu.cn

ABSTRACT

Cyber Threat Intelligence (CTI) parsing aims to extract key threat information from massive data, transform it into actionable intelligence, enhance threat detection and defense efficiency, including attack graph construction, intelligence fusion and indicator extraction. Among these research topics, Attack Graph Construction (AGC) is essential for visualizing and understanding the potential attack paths of threat events from CTI reports. Existing approaches primarily construct the attack graphs purely from the textual data to reveal the logical threat relationships between entities within the attack behavioral sequence. However, they typically overlook the specific threat information inherent in visual modalities, which preserves the key threat details from inherently-multimodal CTI report. Inspired by the remarkable multimodality understanding capabilities of Multimodal Large Language Models (MLLMs), we explore its potential in enhancing multimodal attack graph construction. To be specific, we propose a novel framework, MM-AttacKG, which can effectively extract key information from threat images and integrate it into attack graph construction, thereby enhancing the comprehensiveness and accuracy of attack graphs. It first employs a threat image parsing module to extract critical threat information from images and generate textual descriptions using MLLMs. Subsequently, it builds an iterative question-answering pipeline tailored for image parsing to refine the understanding of threat images. Finally, it achieves content-level integration between attack graphs and image-based answers through MLLMs, completing threat information enhancement. We construct a new multimodal dataset, AG-LLM-mm, and conduct extensive experiments to evaluate the effectiveness of MM-AttacKG. The results demonstrate that MM-AttacKG can accurately identify key information in threat images and significantly improve the quality of multimodal attack graph

construction, effectively addressing the shortcomings of existing methods in utilizing image-based threat information. Code and the corresponding dataset will be released upon acceptance.

Keywords Cyber Threat Intelligence · Attack Graph Construction · Multimodal Large Language Models

1 Introduction

As cyber attacks increase in frequency and complexity, they represent a critical challenge to modern cybersecurity defenses. Attack graph serve as the effective means to combat these escalating threats by graphically depicting the progression of attacks through interconnected nodes that represent individual attack steps, exploited vulnerabilities, and targeted assetss [1–5]. Attack Graph Construction (AGC), the task of generating such graphs systematically, relies on diverse data sources, such as system logs, human-curated knowledge, and Cyber Security Intelligence (CTI) reports. Among these, CTI reports are especially promising. They deliver timely and precise threat intelligence, which helps identify key attack paths and focus on high-risk threats [6–9]. Due to its immense application value, the attack graph construction task has attracted extensive attention from both academia and industry.



Figure 1: Motivation for incorporating images into constructing attack graphs is explained as follows. The images in CTI reports are highly complex and diverse, containing rich threat information. By parsing these images, we can greatly improve our understanding of threat events and thus enhance the quality of attack graphs.

The research on attack graph construction methods has gone through the following phases. In the initial phase, researchers established rules through expert knowledge, extracted Indicators of Compromise (IoCs) based on regular expressions, and constructed attack graph frameworks as perceived [2]. However, this approach was limited by the fixed patterns of regular expressions and the subjective understanding of experts, making it difficult to adapt to the constantly changing tactics and techniques of cyber attackers. To address these challenges, deep learning-based methods have been proposed, such as AttacKG and ThreatKG [1, 3]. Incorporating deep learning has significantly enhanced the construction efficiency of attack graphs. Nevertheless, the complexity of model selection and the stringent requirements on labeled data quality pose challenges for practical implementation. Then, with the tremendous success of Large Language Models (LLMs), more and more researchers have begun to explore the use of LLMs to address the problem of attack graph construction. These pioneering works have explored ways to apply LLMs in attack graph construction tasks. Such as the use of In-Context Learning (ICL) to apply LLMs to attack graph construction sessions like threat data automatic labeling [10], threat entity recognition [11], and threat event extraction [12]. Compared with traditional methods, LLM-based approaches demonstrate advantages in effectiveness, usability, and scalability.

Despite the advantages of the LLM-based method, existing research often neglects the rich multimodal information in CTI reports, such as attack process diagrams, case study illustrations, and system screenshots. We refer to these visual materials containing threat-related content as "threat images". As is illustrated in Figure 1, threat images exhibit complexity and diversity. Threat images are distributed in different locations of the CTI report according to their functional type, providing visual evidence of the corresponding attack scenarios, complementing the textual descriptions, and providing unique insights. Overlooking these images can limit the depth and accuracy of attack graph construction. How to parse threat images to enhance attack graphs remains a pressing research gap.

To bridge this gap, we aim to integrate images into cyber threat analysis to construct multimodal attack graphs. Even though image understanding research is well established, we face three major challenges for this task [13–15] : First, domain-specific knowledge is essential. Analyzing textual cybersecurity data requires domain knowledge to enhance LLMs' understanding of threat contexts [11, 12]. Similarly, existing general LLMs lack inherent mechanisms to effectively parse cyber-specific visual semantics (e.g., network diagrams, intrusion detection alerts), limiting their applicability to threat image analysis (**Challenge 1**). Second, a picture is worth a thousand words. We need to determine

how to precisely and swiftly capture the most critical information from threat images for attack graph construction. Therefore, a new prompt approach is needed to unleash the potential capabilities of MLLMs (**Challenge 2**). Finally, the quality of threat information extraction is critical. Previous studies [16] have shown that improving performance through image feature exploration typically depends on large quantities of labeled training data. However, these data often lack generalizability across different task definitions. Consequently, there is a pressing need for a self-supervised mechanism capable of optimizing the extracted threat image information (**Challenge 3**).

To address the aforementioned challenges, we develop a novel framework for multimodal attack graph construction, named as **MM-AttacKG**. Specifically, to solve the first challenge, we integrate cybersecurity knowledge into threat image parsing. In LLMs usage, prompt learning and knowledge introduction improve the expertise and relevance of threat image information parsing. Then, to address the second challenge, we redefine threat image parsing as an iterative question-answering process, inspired by human brainstorming. Each question targets a specific aspect of attack graph construction. During each iteration, questions are systematically generated to probe critical aspects of the image within the cybersecurity domain, ensuring targeted exploration of its meaningful implications. Finally, to address the third challenge, we set up two answer optimization paradigms. Evaluate the answer content from different dimensions and further optimize the parsed threat information to improve the quality of threat information extraction. To evaluate our approach, we constructed an exploratory dataset by incorporating images from threat intelligence. We call the dataset **AG-LLM-mm**. The final evaluation results show that multimodal attack graphs have richer threat information than text-based attack graphs when supplemented with visual information. The main contributions are as follows:

- To the best of our knowledge, this is the first comprehensive study of exploring visual information for attack graph construction in the era of LLMs.
- We designed a multimodal attack graph construction framework that identifies and integrates threat image information into an LLMs-based attack graph construction process.
- Extensive experiments justify that our framework is able to effectively identify important information embedded in threat images, and the visual information can enhance the completeness of attack graph construction. In addition, these findings point out promising and relevant directions for future research.

2 Related Works

In this section, we review related work in three branches.1) CTI report extraction, 2) LLMs for cyber security. 3) Multimodal for cyber security.

2.1 CTI Report Extraction

Indicators of Compromise (IoCs). Structured IoC sharing remains a cornerstone of open source CTI frameworks, as evidenced by platforms and studies such as [17–20]. These systems catalog attributes including malicious file hashes, process identifiers, and malware metadata [21–23]. However, their reliance on isolated, low-fidelity data limits their utility in reconstructing multi-phase adversarial campaigns, as they fail to capture contextual or behavioral linkages between indicators.

Unstructured Text Intelligence Extraction. The cybersecurity community has developed advanced methods to transform unstructured threat reports into actionable intelligence. Ramnan pioneered automated extraction of vulnerability exploitation patterns and mitigation strategies from unstructured CTIi [24]. Subsequent work by Ghazi established frameworks for correlating extracted threat concepts [25], while Ghaith introduced information-theoretic metrics (entropy and mutual information) for text analysis in security contexts [26]. The EXTRACTOR system [2] generated attack behavior graphs without domain-specific text assumptions, complementing the joint extraction method of network relation triples from [27]. Mao [28] addressed coreference resolution challenges in threat action extraction, whereas [29] mitigated OOV issues through multi-granular feature extraction. Advanced neural architectures like TA-GCN [30] and KnowCTI [31] further improved entity-relationship modeling by integrating domain knowledge and dependency-aware embeddings.

TTP Identification & Operationalization. MITRE Technology, Tactics, Procedures (TTP) matrix [32] has become the de facto classification standard for cyber threat adversarial pattern recognition. Recent advances include: Context-aware TTP extraction via [5] enables real-time defense orchestration, while Ayoade automates TTP classification and mitigation mapping from heterogeneous threat reports [33]. Ge employs semantic impact scoring with conditional probability for TTP prediction [34], whereas Liu's ATHRNN architecture captures hierarchical TTP dependencies through transformer-recurrent hybrid networks [35]. The TCENet framework bridges TTP and operational defense by generating Sigma rules directly from TTP descriptions [36].

2.2 Multimodal for cyber security

Multimodal learning enhances threat understanding by fusing heterogeneous data sources.

Multimodal data fusion and threat detection. In the area of cyber threat analysis and modeling, Nirnimesh proposed a multimodal graph-based approach for modeling and analysing cyber attacks [37]. This work constructs a comprehensive analysis framework that includes victims, adversaries, autonomous systems, and cyber events by representing the stages, actors, and outcomes of cyber attacks as multimodal graphs. Multimodal data fusion techniques were widely used for critical infrastructure protection and threat detection. For example, Nikolaos proposed an attack detection framework based on multimodal data fusion and adaptive deep learning for the vulnerability of critical water infrastructure to cyber attacks [38]. The framework improves the detection of attacks by integrating multimodal data. In addition, Li proposed an LLM-based detection method for phishing attacks, which combines image and text information of web pages to overcome the limitations of traditional single-modal-based methods [39].

Multimodal threat information recognition. Automated extraction of cyber threat intelligence is another important application area of multimodal learning. Zhang proposed the EX-Action framework, which extracts threat actions from CTI reports through natural language processing techniques and combines them with multimodal learning algorithms to identify threat behaviors [40]. Similarly, Xiao proposed an advanced persistent threat participant attribution method based on multimodal and multilevel feature fusion (APT-MMF), which solves the problem of ignoring heterogeneous information in existing methods [41].

Security of Multimodal Large Language Models. To address the security of multimodal learning models, researchers have proposed various defense mechanisms. Liu proposed a 'machine forgetting' based defence mechanism for backdoor attacks in multimodal comparative learning to reduce the impact of malicious behaviors on the inference process of the model [42]. Shan reveals the vulnerability of text-to-image model generation to poisoning attacks under large-scale training data and proposes a 'night shadow' attack method to precisely control the model output through a small number of poisoned samples [43]. JailGuard focuses on the vulnerability of large and multimodal language models to jailbreak and hijacking attacks and proposes a generic detection framework for identifying multiple attacks across modalities [44].

2.3 LLMs for Cybersecurity Threat

The application of large language models in cybersecurity has emerged as a multi-faceted research frontier.

Systematic methodological frameworks form the theoretical basis for LLM security implementations. Kucharavy established essential taxonomies for generative language modeling, delineating capability boundaries and implementation principles [45]. Building on this foundation, Wrsch proposed a semantic pattern recognition framework for cybersecurity knowledge entity extraction [46], while Pan achieved breakthroughs in log anomaly detection through retrieval-augmented architectures with vector database integration [47]. In complex pattern recognition, Ferrag demonstrated FalconLLM's effectiveness in identifying multi-stage attack vectors through contextual reasoning [48]. Charan quantitatively evaluated LLMs capabilities in implementing MITRE ATT&CK techniques, establishing benchmark comparisons between ChatGPT and Bard for tactical code generation [49]. For security operations, Rigaki deployed pretrained models as autonomous agents in network environments, optimizing sequential decision-making processes [50]. The chained prompt engineering framework proposed by Moskal structured threat response workflows through plan-act-report cycles [51]. Addressing LLM domain adaptation challenges, Kereopa constructs an evaluation system combining expert case analysis with quantitative metrics, while current research generally faces the challenge of enhancing models' cybersecurity-specific cognition [52].

3 Preliminary

3.1 Problem Formulation

Current research on multimodal attack graph construction does not have a rigorous paradigm. Specific application scenarios influence the characterization and structure of attack graphs. To formally define the problem, we adopt the attack graph definition in text-based attack graph work as a basis.

CTI Report. A CTI Report is an evidence-based, structured analysis that encompasses the context, mechanisms, indicators of compromise, potential impacts, and actionable recommendations regarding existing or emerging cyber threats. It serves as a critical output of the cyber threat intelligence lifecycle, providing organizations with actionable insights to proactively manage and mitigate cyber risks.



Figure 2: The atomic event data structure of the attack graph contains three parts: **Tactics**, **Techniques & Procedures**, where procedures are presented in the structure of the threat behavior graph.

Attack Graph. An attack graph is a graphical representation method used for modeling and analyzing cyber attack pathways. It illustrates the relationships of the elements in a threat event and the steps taken by an attacker to launch an attack, thereby assisting security analysts in understanding potential threats within a network environment and assessing the security posture of a system.

Text-based Attack Graph Construction. Text-based attack graph obtains threat information from the textual modal content of CTI reports. Given a collection of historical threat intelligence text events $E = \{e_1, e_2, \ldots, e_k\}$. This formulation defines extracting each threat behavior as a quadruple (s, a, o, t), which is also called an atomic event, where s, a, o and t correspond to the subject, threat action, object, and timestamp in the current threat event. At each timestamp t, all the quadruples form an event graph, denoted as $AG_t = \{(s, a, o, t)\}^N$, where N is the number of atomic events at the threat event. It can be seen that the scheme uses a sequence of threat actions to characterize the evolutionary process of the attack graph. Furthermore, as is shown in Figure 2, some works [1] focus on the content support of TTP (Technology, Tactics, Procedures) labels for attack graphs to characterize the cyber threat technologies to which threat atomic events are mapped. Specially, each atomic event is extended from a quadruple (s, a, o, t) to a quintuple (s, a, o, t, p), where $s \in \mathcal{E}, a \in A, o \in \mathcal{E}$ and $p \in P$, represent the subject, threat action, object, and TTP label. Correspondingly, the atomic event at each timestamp will be extended as $AG_t = \{(s, a, o, t, p)\}^N$. It is worth mentioning that for the presence of non-verbal relations R in threat events, R is the set of non-verbal relations that are not ignored, but rather are used as attack graph supplementary links that hang over the relevant entities, but not as attack steps descriptions.

Multimodal Attack Graph Construction by Image-enhanced. Based on the text-based attack graph, the images associated with structured events in CTI reports are introduced to construct the multimodal attack graph, where the images are represented by $V = \{v_1, v_2, \dots, v_m\}_{m=1}^M$, where M is the number of images. The image-enhanced attack graph construction process is divided into two phases: multimodal attack information extraction and attack graph integration.

- **Image Attack Information Extraction**. This phase is based on the content summarization of the text-based attack graph, and parses the attack information contained in the images of the CTI report from different aspects, providing additional context and detail for the construction of the attack graph.
- **Multimodal Attack Graph Integration**. This phase leverages the attack information extracted from the images in the CTI report as the basis, and supplements the text-based attack graph from three dimensions: entities, relation, and techniques. By integrating threat information from multiple modalities, it forms an image-enhanced attack graph.

Finally, the multimodal attack graph construction task can be characterized as follows: given a collection of historical threat intelligence text events $E = \{e_1, e_2, \ldots, e_k\}$ and associate images $V = \{v_0, v_1, \ldots, v_m\}$, identify quintuples (s, a, o, t, p) from text and images form that can portray the evolutionary flow of threat events to form a multimodal attack graph.



Figure 3: The overall framework of MM-AttacKG consists of five modules: (1) **Text-based Construction**, here we use AttacKG+ as the constructor to construct a text-based attack graph by parsing CTI text. (2) **Brainstorming**, the module targets key aspects of image parsing through question generation; (3) **Extraction**, the module that combines the question set with the image content to extract key information from CTI images; (4) **Verification**, the module optimizes the quality of threat image understanding through a two-stage process of question filtering and answer refinement; and (5) **Integration**, the module refines the attack graph by adding modal information from CTI image.

3.2 Text-based Attack Graph Construction

AttacKG+¹ is a novel attack graph construction framework based on large language models, designed to transform textual cyber threat intelligence reports into structured attack graphs. The framework employs a modular design to build multilayered attack graphs, comprising four modules: Rewriter, Parser, Identifier, and Summarizer. Each module leverages the instruction prompting and in-context learning capabilities of LLMs to sequentially perform tasks such as report rewriting, behavior graph extraction, technique label matching, and state summarization. Furthermore, AttacKG+ introduces an upgraded attack knowledge schema that represents the attack process as a temporally unfolding complex event. Each temporal step encompasses three layers: behavior graph, TTP labels, and state summary, providing a more comprehensive characterization of the attack process.

By utilizing the powerful language understanding and zero-shot learning capabilities of LLMs, AttacKG+ overcomes the limitations of traditional methods in terms of generalization ability and adaptability to new attack scenarios. Given the superior processing performance of AttacKG+ in text-modal threat intelligence, we use it as a textual attack graph constructor for multimodal attack graph construction work as shown in Figure 3. Meanwhile, we obtain the summary information of the current report (Global-Context) and the image context information (Image-Aware-Context) as the parsing support of the threat image through the summary function module in it.

4 Approach

The proposed framework is outlined in Figure 3. Our approach MM-AttacKG consists of five phases, which smoothly implements the flow from threat image parsing to attack graph multimodal gain. Section 4.1 outlines the brainstorming procedure, which is intended to clarify the key aspects of threat image parsing. Section 4.2 defines the threat information

¹https://github.com/multilayer-go/AttacKG-plus

extraction process, which is specifically designed to accurately extract threat-related information from threat images. Section 4.3 presents the question filtering mechanism, which can effectively screen out threat information aspects that meet the criteria for constructing attack graphs. Section 4.4 proposes two optimization paradigms to enhance the quality of answers in threat image extraction. Section 4.5 introduces the attack graph integration scheme, which can integrate the parsed threat image information into text-based attack graphs.

Table 1: Annotated descriptions of each assessment aspect involved in the answer quality assessment process.

Aspects	Instructions
Accuracy	 Score 1: The description is entirely incorrect to the question and contains severely misleading information. Score 2: The description is partially correct but includes significant errors or irrelevant content. Score 3: The description is mostly accurate but contains minor errors or ambiguous phrasing. Score 4: The description is accurate and correct but lacks direct image references. Score 5: The description is fully accurate, unambiguous, and directly supported by the image content.
Consistency	 Score 1: The description completely contradicts or is unrelated to the image information. Score 2: The description includes limited relevant details, but most content deviates from the image information. Score 3: The description partially aligns with the image information but contains irrelevant or redundant content. Score 4: The description closely adheres to the image information, with only minimal unrelated content. Score 5: The description is entirely based on the image information, with no extraneous or redundant elements.
Completeness	 Score 1: The description fails to address any critical aspects of the question, omitting all essential information. Score 2: The description addresses only a subset of the question, omitting most key details. Score 3: The description broadly covers the question's requirements but lacks minor details. Score 4: The description comprehensively addresses the question, with only negligible omissions of minor details. Score 5: The description fully addresses all requirements of the question, providing thorough details.
Relevance	 Score 1: The description is entirely unrelated to cybersecurity and provides no value for threat analysis. Score 2: The description has marginal relevance, requiring substantial inference to connect to threat analysis. Score 3: The description partially relates to cybersecurity but lacks explicit ties to practical applications. Score 4: The description directly aligns with cybersecurity and offers moderate analytical value for threat analysis. Score 5: The description focuses heavily on cybersecurity and provides actionable insights into threat analysis.

4.1 Brainstorming

In multimodal threat intelligence analysis, accurately interpreting domain-specific threat images is essential for constructing attack graphs. These images visually depict cybersecurity scenarios and have different roles and contextual links in attack graphs. Brainstorming phase mimics human cognition by analyzing image types and key features, generating critical questions to pinpoint image parsing priorities, thus underpinning follow-up work.

Specifically, in this phase, we first seed the LLMs with an initial set of questions, named leading questions (Table 5). The leading questions contain a set of general questions that security practitioners are interested in for different types of threat images, which are grouped according to different image types. Secondly, the LLMs are guided by the prompts to generate a set of general questions for the current threat image based on the content of the image and the corresponding leading questions. Then, in order to better construct task associations with the attack graphs extracted from the textual modalities, the prompts guide the LLMs to generate a set of task-specific questions for the image based on the structure of the unimodal attack graph and the content of the current threat image. The general and task-specific question seeds are explained below.

- General Question. This type of question focuses on mining information about the attributes that the images themselves have. Such as image subject, image type, and image source. The purpose is to prompt LLMs to perform a summary of the content of the image itself.
- **Task-specific Question**. This type of question focuses on the task-oriented nature of threat images in constructing attack graphs. Such as "*What are the temporal features exhibited by the attack flow graph?*" The purpose of this is to motivate LLMs to mine threat image information from the attack graph construction task.

where the prompts used to generate the question pool are displayed in Table 6 (Question Generation). Following the brainstorming phase, the general questions and task-based questions together form the question pool, which forms parsing guidance for the current threat intelligence images.

4.2 Extraction

The extraction phase builds on the brainstorming output. It uses a framework to help LLMs give accurate answers to parsing questions. This approach facilitates the construction of attack graphs by introducing threat image context information as parsing support to create high quality answers.

Specifically, after generating the question pool, we will prompt the LLMs to answer the questions based on the question descriptions and threat image contents. In this phase, we focus on mining the content of threat images by answering the corresponding questions. In the actual threat image parsing process, we found that when LLM parses threat images without CTI contexts, the results are ambiguous and lack logical reliability. For example, the answer only describes abstract concepts but cannot identify specific threat objects. And such threat information is not suitable as a basis for the construction of multimodal attack graphs. To address this issue, we enhanced the summarization module of AttacKG+ to extract two types of parsing-support information via image tag localization: image-aware context and global context. Section 5.5 (Ablation Study) evaluates their importance for threat information extraction.

- **Image-Aware-Context**. In this setting, the Image-Aware-Context is derived from the summarization of the image context paragraph, such as image type and analysis approach, providing dynamic contextual information for threat images.
- **Global-Context**. In this setting, the Global-Context originates from the content abstract of the CTI report in which the threat image is located, encompassing the outline of the CTI report's subject matter and offering a macrosemantic framework for threat image parsing.

We control the output from three aspects: content limitation, topic relevance, and expression format through rule setting. The template of question answering prompts is shown in Table 6 (Question Answering).

4.3 Verification-Question Filtering

Question filtering aims to be more targeted and effective in parsing threat images, so as to exclude irrelevant questions. This is because the quality of a question largely determines how helpful its answer is for attack graph construction. The question filtering module consists of two phases: direct correlation question capture and answer-oriented question capture.

- Direct correlation question capture. The idea of direct correlation question capture is to prompt LLMs to judge the quality of questions by taking the attack graph summary and domain rules as the basis. This method is suitable for cases where the direct formulation of the question has domain characteristics. For example, "What is the functional role of the malicious script in this image." In this case, "malicious script" has strong cyber security domain context.
- Answer-oriented question capture. The idea of answer-oriented question capture is to first generate the answer to the corresponding question, and then judge the quality of the question based on the content of the answer and the hints of the LLMs. This method is suitable for cases where the direct formulation of the question does not have domain characteristics or the domain characteristics are weak. For example, "What trend does the graph in this image reflect?" The description of the question does not have direct domain characteristics, but is based on the image identification and embedded text description of the chart. It is possible that threat information will be obtained that will assist in the construction of the attack graph.

After filtering the set of questions generated in the threat image parsing phase, we labeled the above two types of questions and fused them into the question candidate set for constructing multimodal attack graphs.

4.4 Verification-Answer Refinement

After completing question pool (Q) construction and question filtering, we input the questions along with the current threat images into the LLMs to obtain the corresponding set of question answers called answer pool (A). Specifically, these answers are our analysis of threat intelligence images from both general and task perspectives. However, we have to pay attention to two issues: First, it is necessary to determine that the questions currently generated are relevant to the attack graph construction rather than the minutiae. Second, we need to ensure that the answer to the question at hand is superior.

To address the above issues, we set up a self-learning module for LLMs to help optimize the quality of answer generation by iterative question-answering. Here, we propose two basic prompt optimization paradigms, question-led (Q-Led) and answer iteration (A-Iteration), drawing on the idea of textual answer optimization. Specifically, after completing the answer relevance analysis, we set up an answer assessment module to evaluate the quality of generated answers. The quality criteria for the answers are divided into four levels, including: failing, satisfactory, good, excellent. As is



Figure 4: The threat image parsing answer optimization paradigm. **Question-led**, generate a target refinement parsing guide to guide the next round of question parsing of threat images. **Answer Iteration**, generate an initial answer, and refine the current answer in the next round based on the optimization comments.

shown in Table 1, the assessment of answer quality is realized by guiding the LLMs to evaluate from four dimensions: accuracy, consistency, completeness, and relevance. Accuracy represents whether the answer accurately answers the corresponding question. Consistency represents whether the answer maintains content relevance to the threat image information. Completeness represents whether the answer adequately answers the needs of the corresponding question. Relevance represents whether the answer fits the cybersecurity domain in terms of presentation.

For answers below the good level, we perform answer refinement. The stopping condition for the number of iteration rounds of answer refinement is that (1) the current answer reaches "excellent/good" or (2) the set iteration threshold is reached. The optimization approach adopted in the answer refinement process is as follows:

$$IA^{current} = LLM \left(IA^{last} + Image \right)$$
(1)

As shown in equation 1, IA^{current} denotes the threat image extraction answer in the current round, IA^{last} denotes the threat extraction answer in the previous round, and Image denotes the current threat image file to be parsed. The idea of answer iteration is to evaluate the answers by co-inputting the generated answers and the belonging images into LLMs (without history). Then LLMs will output quality ratings and optimized comments for the current answer. Then the current answer is refined in the next iteration based on the optimization comments.

$$IA^{current} = LLM (Suggestions^{last} + Image)$$
⁽²⁾

In contrast, as shown in equation 2. Suggestion^{current} represents the optimization tips given by LLM in the previous round based on the answers and image content. The idea of question guidance is to evaluate the answers by co-entering the generated answers and the belonging images into LLMs (without history). Then LLMs will output quality ratings and the targeted refinement parsing guide for the current image. It is worth noting that the suggestion differs from the answer comment. The suggestion is not oriented toward the previous round's answers but rather provides LLMs with an outline for a more comprehensive understanding of the image. In the next round, the question, the image along the guide will be fed into LLMs together to get the new round of answers.

The processing of the above two answer refinement paradigms can effectively improve the quality of responses against cyber threat intelligence images. The template of answer optimization prompts is shown in Table 6 (Answer Optimization).

4.5 Integration

According to the problem formulation in section 3.1, this section presents a framework for constructing multimodal attack graphs based on LLMs. The construction of text-based attack graphs by AttacKG+ mainly involves (1) threat behavior extraction and (2) TTP labeling. These two sub-tasks identify the behavior level and technique level threat information from the cyber threat intelligence respectively and portray the evolution flow of the whole threat event. The details are as follows: for cyber threat intelligence, the method takes as input cyber threat intelligence in textual

modality. First, an attack graph ontology model is constructed via the STIX 2.0 standard [53], and then threat atomic event (s, a, o, t) is extracted based on the ontology model, where t is derived from the relative timing of that atomic event in the intelligence. Then, the dictionary of technology labels is constructed by the TTP matrix of MITRE ATT&CK [32], identifying the technology labels corresponding to atomic events, and expanded to form the quintuple (s, a, o, t, p). Next, based on the timestamp t, the attack graph $AG_t = \{(s, a, o, t, p)\}^N$ is formed.

It is worth considering that the image parsing module will generate multiple questions and their corresponding answers for the given image. However, answers originating from the same threat image do not necessarily correspond to the same atomic event in the attack graph. This is because the information in the images can complement different aspects of the threat event. For example, the content in the attack flow diagram involves information about the flow of the entire current threat event. To solve this problem, we construct each problem as a corresponding "threat enhancement reference". Then, the structural enhancement of the attack graph is achieved by using the threat enhancement units as reference information.

First, we use LLMs to merge image topics, questions, and their corresponding answers to form threat enhancement reference extracted from the current threat image. For example, 'This is the temporal description (Question) of the protocol attack flowchart (Image Theme) as follows:(Answer)'. In this case, the protocol attack flowchart is the topic of the current threat image, the temporal description is the corresponding particular question, and the specific content is the answer obtained through three stages of brainstorming, extraction, and verification. Then, we use the threat enhancement information as a basis to correlate and enhance the current attack graph. We regulate three kinds of functional enhancements of threat image attack graphs:

- Node Extension. Adding new nodes or checking the node descriptions of existing attack graphs to supplement data or attack resources through threat image information.
- **Relation Update**. If the image reveals a new attack action (such as *deliver* or *execute*), add or replace the relation and description.
- **Technique Addition**. Match the added technique (such as *T1204.002-User Execution*) according to the MITRE ATT&CK framework.

Through the targeted parsing and fusion of threat images by LLMs, the attack graphs combined with threat images can be greatly improved in the three levels of data integrity, structural flow and technical richness.

5 Evaluation

We conduct extensive experiments to answer the following research questions:

- **RQ1:** How does MM-AttacKG perform against existing threat information extractors? (see Section 5.2)
- **RQ2:** How effective is MM-AttacKG in enhancing attack graph with threat image information? (see Section 5.3)
- RQ3: Whether each key module in MM-AttacKG effective? (see Section 5.4)
- RQ4: How does each technical module function within the MM-AttacKG framework? (see Section 5.5)

5.1 Experiments Setting

To ensure the rationality and reproducibility of the evaluation phase for MM-AttacKG, we describe the experiments setting, including the dataset, baseline methods, and implementation details.

5.1.1 Dataset

To evaluate MM-AttacKG, we constructed the dataset by collecting cyber threat intelligence reports from Cisco Talos Intelligence Group [54], Microsoft Security Intelligence Center [55]. We utilize AMinerU-PDFScanner ² to extract each CTI report into two parts: textual content and threat images. For the textual content, we extract it into attack graphs using the construction process provided in AttaKG+ [12], and manually verify the extraction results. For the threat images, we follow the following process: first, due to the layout, the original CTI reports or web pages often contain a large number of irrelevant, low-quality threat images. Therefore, we preliminarily filter the threat images and remove those with the following problems. (a) containing irrelevant information such as logos, advertisements, etc. (b) presence of occlusion or strong watermarks; (c) weakly informative visual samples; (d) poor image quality or lack of clarity; and (e) crippled images or incomplete graphics. Then, to achieve this, we set up image rule filtering rules, write the prompting scheme, and use LLMs to analyze the content and board style of images in order to eliminate

²https://github.com/liuhuapiaoyuan/MinerU-PDFScanner

irrelevant images. Meanwhile, we arranged researchers with cybersecurity background to proofread again. Finally, we integrate textual content and threat images. Then name the dataset as Attack Graph-LLM-multimodal, short for AG-LLM-mm. Three postgraduate students from our team acted as participants in the manual assessment of the CTI reports. Participants exchanged views after completing the assessments individually, which was secondarily discussed in order to obtain a comprehensive assessment.

Table 2: Effectiveness of MM-AttacKG for multimodal threat information extraction. For each column, the bold number indicates the best performance, and the underlined number corresponds to the second-best performance. Human Anotation-Text represents the entity, relation and technique extraction of threat intelligence text under the manual annotation process.

Mathad		Entity			Relation			Technique	
Wethou	Precision	Recall	F-1	Precision	Recall	F-1	Precision	Recall	F-1
			Te	xt-based Metho	d		•		
Extractor	0.6568	0.5387	0.5919	0.2158	0.1026	0.1391	-	-	-
AttacKG	0.5580	0.2612	0.3559	-	-	-	0.2060	0.3399	0.2565
AttacKG+	0.7701	0.5294	0.6274	0.7693	0.6806	0.7222	0.4502	0.4481	0.4491
Human Anotation-Text	1.0000	0.4559	0.6263	1.0000	0.6820	0.8109	1.0000	0.6547	0.7913
			Imag	e-enhanced Met	thod				
ICL	0.6901	0.7326	<u>0.7107</u>	0.7106	0.8261	<u>0.7640</u>	0.4948	0.5383	0.5156
CoT	0.6805	0.7432	0.7105	0.6949	0.8383	0.7599	0.5063	<u>0.5508</u>	0.5277
MM-AttacKG	0.7224	0.8280	0.7716	0.7460	0.8973	0.8147	0.5256	0.6232	0.5703

5.1.2 Baseline Methods

To assess the effectiveness of MM-AttacKG for threat information extraction and image-enhanced attack graph integration, we compare it with four text-based threat information extractors (e.g., AttacKG [1], EXTROCTOR [2], AttacKG+ [12]). Meanwhile, we migrate two LLM-based methods (ICL [56] and CoT [57]) to the multimodal threat information extraction and attack graph aggregation task. More details are in Appendix Table 7 and Table 8.

5.1.3 Implementation Details

As the field of cybersecurity requires transparent information processing and prevents information leakage, commercial API services may be unstable and face service restrictions. We have chosen to use the open-source multimodal large language model from the Qwen [2] series (e.g., Qwen2.5-VL-72B, Qwen2.5-VL-32B, Qwen2.5-VL-7B) for relevant experiments. We hosted the large language models on an AliCloud Elastic Compute Service (ECS)³. instance and developed all implementation modules within a Python 3.10.14 environment. To ensure the reproducibility of the research, we fixed the temperature parameter of the proprietary LLMs used to 0.7 and set the seed parameter to a constant value. Meanwhile, to keep the experiments manageable, for different LLMs prompt methods, we provide them with exactly the same context and specify output limits in the same format. ethod, we turn on the relevant optimization setting to ensure the construction quality of the attack graph. To prevent invalid responses, we limit the maximum output length to 512 tokens when parsing threat images.

5.2 Performance Comparison for Threat Extraction (RQ1)

This experiment aims to quantify the effectiveness of images in enhancing CTI understanding by counting the threat information that can be mined from CTI reports. Specifically, we analyze the performance improvement of threat information extraction with the introduction of visual information.

Based on the problem formulation in section 3.1, a cyber threat event consists of multiple threat atomic events in temporal order. These atomic events are represented in the threat behavior graph as interconnected quintets (s, a, o, t, p), where s and o are coordinated for entity extraction, and actions a and entity-entity relations are coordinated for relation extraction. p represents the TTP technique label involved in the current behavior. By parsing the threat image information, more information can be gained to introduce to the attack graph. The gained information includes three aspects: (1) Entity. Subjects and objects in cyber threat behavior. (2) Relation. Descriptions of correlations between subjects and objects in cyber threat behavior. (3) Technique. Mining technique labels from threat images based on the TTP matrix. Table 2 summarizes the effectiveness of MM-AttacKG for multimodal threat information extraction. Our findings are:

³https://www.aliyun.com/

Yongheng Zhang et al.



Figure 5: Performance of MM-AttacKG's threat gain in different prompting schemes and LLM versions.

(1) Advantages of multimodal integration. The image-enhanced method demonstrates superior performance across entity extraction, relation extraction, and technical identification tasks compared to Text-based methods. This indicates that incorporating visual information significantly improves threat intelligence mining efficacy and enhances perception of complex threat scenarios.

(2) Image-enhancement methods comparison. Among multimodal approaches, the MM-AttacKG model achieves superior F-1 scores across all three subtasks relative to ICL and CoT. This suggests that MM-AttacKG's architecture provides stronger multimodal integration capabilities and semantic comprehension, offering more reliable foundations for cybersecurity threat intelligence analysis.

(3) Subtask difficulty disparities. Technical identification exhibits markedly lower F-1 scores (e.g., 0.4491 for AttacKG+ in technical identification vs. 0.6274 and 0.7222 in entity/relation extraction). This highlights the heightened complexity of technical identification tasks and underscores the need for algorithmic improvements to enhance semantic parsing capabilities in this domain.

(4) Limitations of manual annotation. Though manual text annotation by humans yields high precision, it is constrained by the absence of image information, resulting in low recall rates. Additionally, this process is labor-intensive. The dependence on domain experts further limits scalability and real-time applicability. These factors underscore the critical need for automated methods in large-scale data processing and continuous monitoring within cybersecurity contexts.

5.3 Performance of Attack Graph Integration (RQ2)

To address RQ2, we introduced In-Context Prompt Learning (ICL) and Chain of Thought (CoT) as comparative methods for constructing multimodal attack graphs. ICL enables LLMs to learn directly from input examples or instructions to complete tasks. CoT allows learners to solve problems through step-by-step reasoning, mimicking the human thought process for complex problem-solving. MM-AttacKG enhances the performance of attack graph construction by integrating prompts from three phases: brainstorming, extraction, and verification.

For the dataset in the experimental setup, we implemented the above prompt schemes in three versions of Qwen-VL. We evaluated the performance of the attack graphs in terms of entity, relationship, and the gain from utilizing image modality information. Namely, the incremental information of the updated multimodal attack graph over the text-based attack graph at the entity, relation, and technology levels. The evaluation results are shown in Figure 5. The experimental results indicate that: (1) Across all versions of large language models, MM-AttacKG outperforms the other two prompt schemes in terms of information gain in the entity, relationship, and technique dimensions. (2) We observed the most significant performance fluctuations in ICL across different LLM versions, suggesting that merely providing operational examples results in poor robustness of LLMs for CTI image parsing and attack graph construction. Although the CoT prompt scheme demonstrated strong robustness in constructing mult-modal attack graphs across various LLM versions, its chain-like thinking process lacks an active guidance mechanism for performance enhancement, resulting in limited information gain. In summary, MM-AttacKG is superior as a multimodal attack graph construction scheme.

5.4 Ablation Study (RQ3)

In our research, to evaluate the influence of support context and various components on the performance of MM-AttacKG, we carried out two types of ablation studies through the controlled removal of support context and key modules. To confirm the facilitating role of supporting information in threat image information extraction, we removed specific configurations, namely Image-Aware-Context&Global-Context, Image-Aware-Context, and Global-Context. In order to verify the importance of key components for threat image information extraction, we removed the settings of

Table 3: Ablation study. For each column, the bold number indicates the best performance, and the underlined number corresponds to the second-best performance. **Support Context for Image Threat Information Extraction**: I stands for Image-Aware-Context, G stands for Global-Context. **Image Threat Information Extraction Module**: B stands for Brainstorming, V stands for Verification.

Mathad		Entity			Relation			Technique	
Method	Precision	Recall	F-1	Precision	Recall	F-1	Precision	Recall	F-1
				Our Method					
MM-AttacKG	0.7224	0.8280	0.7716	0.7460	0.8973	0.8147	0.5256	0.6232	0.5703
		Support	Context for In	nage Threat Info	ormation Extra	uction			
w/o I&G	0.6413	0.7759	0.7022	0.7298	0.8583	0.7888	0.4543	0.5795	0.5093
w/o G	0.6449	0.7913	0.7106	0.7116	0.8581	0.7780	0.4876	0.6037	0.5395
w/o I	0.6768	0.7740	0.7221	0.7027	0.8539	0.7710	0.4697	0.5826	0.5201
		In	nage Threat In	formation Extra	uction Module				
w/o B&V	0.7710	0.6000	0.6749	0.7887	0.7947	0.7917	0.5005	0.5829	0.5386
w/o V	0.7086	0.8247	0.7623	0.7386	<u>0.8963</u>	<u>0.8099</u>	0.4859	0.6225	0.5458
w/o B	0.7607	0.6028	0.6726	0.7862	0.7936	0.7899	0.4820	0.5795	0.5263

Brainstorming&Verification, Brainstorming, and Verification. The outcomes of these ablation experiments are presented in Table 3.

(1) Multimodal Fusion Advantage. MM-AttacKG, through integrating Image-Aware-Context and Global-Context with component collaboration (Brainstorming, Extraction, Verification), significantly outperforms text-only variants in entity recognition and relation extraction. This highlights the effectiveness of multimodal fusion in enhancing threat information identification, making it a key driver of model performance.

(2) Synergy of Textual Support. Image-Aware-Context and Global-Context support each excel in different tasks. Image-Aware-Context support boosts relation extraction recall with dynamic context, while Global-Context support improves entity recognition precision with macro semantic frameworks. It is their synergy that powerfully boosts the model's performance in a multitude of tasks.

(3) Module Collaboration Superiority. Comparing variants without verification module and those without both brainstorming and verification module shows that removing only verification improves entity recognition but harms threat information extraction. The full MM-AttacKG model achieves better balanced and slightly superior performance in all tasks, proving its component configuration effectively meets diverse task needs.

(4) Overall Performance and Design Rationality. MM-AttacKG delivers consistently strong performance across entity recognition, relation extraction, and threat information extraction. This validates its effectiveness and robustness in multi-task scenarios.

5.5 In-depth and Analysis of Key Modules (RQ4)

Given that our method is an attack graph construction pipeline composed of multiple modules, to address RQ4, we divide this question into three aspects: the diversity of questions generated by the brainstorming module, the necessity of the question filtering module, and the superiority of the answer refinement module. For the brainstorming module, we focus on whether the set of questions generated based on leading questions can sufficiently cover more main points of mining CTI images and whether there is enough distinction among the questions. For the question filtering module, we focus on the necessity of question filtering and how much unnecessary attention information in CTI images we can reduce through this method. For the superiority of the answer refinement module, we focus on the changes in the distribution of answers of different quality levels with the increase of iterative rounds. Then, analyzing the differences between the two optimization paradigms.

5.5.1 The Diversity of Questions Generated By Brainstorming

During the brainstorming phase, two types of questions, namely general questions and task-specific questions were generated. On average, 21 questions were generated per CTI image, a quantity comparable to that in manual judgment scenarios. Specific question examples are provided in Table 5. To assess question diversity, we adopted the monotonicity indicator from work [58]. Here, monotonicity is evaluated by measuring the similarity among questions targeting the same threat image. Higher monotonicity indicates more similar questions and thus poorer diversity. We analyzed the monotonicity of question sets for a random sample of 10 CTI images from the dataset, with results shown in Figure 6. We found that (1) the number of questions varies with the CTI image, reflecting that the information richness of the CTI image affects the number of questions generated.Richer information leads to more threat information being mined.



Figure 6: Brainstorming questions generate evaluation situations. The abscissa represents the monotonicity of the question set, and the ordinate represents the scale of the current image question set.



Figure 7: The answer quality distribution changes with the number of refinement rounds. (1) Positive: Answers scored as excellent or good are marked as positive. (2) Negative: Answers scored as failing or satisfactory are marked as negative.

(2) The monotonicity of question sets remains stable and close to [1], indicating low similarity among the generated questions, thus demonstrating high diversity.

Scheme	Direct correlation	Answer-oriented	Non-Related
Proportion	0.4699	0.4185	0.1116
Size	602	536	143

5.5.2 The Necessity of Question Filtering Module

The question filtering module aims to capture the set of important questions from two aspects: direct correlation and answer-oriented, thereby enhancing the overall quality of the questions generated in the brainstorming phase through filtering. During the construction of AG-LLM-mm, 1,281 questions were initially generated. However, some of these questions had unclear wording or low relevance. After question filtering, the overall question distribution is shown in Table 4. It can be observed that: (1) the proportion of direct correlation questions is 0.4699, and the proportion of answer-oriented questions is 0.4185. This indicates that solely using question direct expression strategies to filter relevant questions will miss many valuable aspects of CTI images. (2) Non-Related questions account for 143, which shows that the initial set of questions generated in the brainstorming phase cannot be guaranteed to fully meet the requirements of attack graph construction. Therefore, further filtering of questions is necessary.

5.5.3 The Superiority of Answer Refinement Module

The answer refinement module refines the parsing results of CTI images through multiple rounds based on optimization paradigms to enhance the quality of answers. We propose two optimization paradigms: question-led and answer iteration. question-led focuses on using the current answer as a case study to suggest optimizations for the next round of answers, while answer iteration emphasizes supplementing and optimizing the current answer. To evaluate the effectiveness of the answer refinement module in improving answer quality, we designed an experiment as shown in Figure 7, setting the number of optimization rounds from 1 to 4 and tracking the changes in answer quality with each round. Here we set the answer to be marked as positive when the score is excellent or good, indicating that it meets the quality requirements, and negative (failing/satisfactory) when it does not. The results show that: (1) As the number of optimization rounds increases, the overall quality rating of the answers consistently improves, demonstrating the effectiveness of the answer refinement module in enhancing answer quality. (2) The question-led optimization paradigm converges more easily, indicating that suggesting new answers based on recommendations may be more effective in addressing the questions than refining the current answers. (3) As the number of optimization rounds increases, the answer refinement effect gradually decreases, which means gradually approaching the optimization limit.

5.6 Case Study

This section delves into the efficacy of MM-AttacKG in real-world security tasks through case studies. In cybersecurity, professionals and researchers often conduct in-depth analyses of existing threats based on CTI reports. This process focuses on two key aspects: one is to accurately extract core entities linked to the attack process from intricate event info and remove irrelevant redundancy; the other is to bridge the inherent knowledge gap between CTI reports and actual attacks, thus precisely determining tactical stages and identifying attack techniques. This analysis heavily depends on the comprehensive evaluation and information extraction of textual and image information.

MM-AttacKG, with its unique advantages, leverages structured knowledge for in-depth event analysis, effectively meeting the above two goals. As shown in Figure 8, based on MM-AttacKG's advanced concept, firstly, it interprets threat texts to structurally describe the attack process or event, accurately building a text-based attack graph. Then, it parses and extracts threat image info, seamlessly integrating it with the text-based attack graph to innovatively form a multimodal attack graph. We've created an intuitive and efficient visualization interface for MM-AttacKG using pyvis⁴.

Take the Stuxnet worm attack as an example. MM-AttacKG first accurately extracts structured knowledge of the threat event scenario from the text description. It can clearly identify the 6 key tactical stages and 14 specific techniques in the attack, like using *T1091-Replication Through Removable Media* for initial access, *T1574-Hijack Execution Flow* for execution, and *T1055-Process Injection* for defense evasion. Next, based on the techniques and entities related to the threat event, MM-AttacKG accurately infers the complex system environment of the intrusion activity. Specifically, the attack graph details how the Stuxnet worm gains initial access via malicious .LNK files and vulnerabilities, achieves precise system control and persistence through the main module, and escalates privileges using a zero-day vulnerability. It can scan network shares for lateral movement and employs various sophisticated techniques to evade detection. Finally, it communicated with the C&C server through the encrypted channel to receive updates and instructions, and adopted threat behaviors to realize serious interference with the normal operation of industrial equipment.

Notably, after integrating threat image information, MM-AttacKG captures three new techniques (*T1003-OS Credential Dumping*, *T1107-Function hooking*, *T1546-Event Triggered Execution*) in the report, derived from deep information extraction of corresponding threat images. Meanwhile, MM-AttacKG offers a more detailed entity set, uncovers deeper and more hidden threat relationships, and accurately supplements and improves the attack process. This enriches and optimizes the attack graph's description of threat events, successfully integrating text and image into an advanced multimodal attack graph. In conclusion, MM-AttacKG provides a solid and reliable foundation of key information to reconstruct threat events with its superior performance, showing great potential for application and value in cybersecurity analysis.

6 Conclusion and Future Work

6.1 Conclusion

In this work, we introduces CTI images into attack graph construction for the first time by analyzing the role of image information in the cyber threat intelligence analysis process.

Leveraging the superior multimodal information understanding capabilities of LLMs, we propose an automated LLMbased framework (MM-AttacKG) for constructing multimodal attack graphs. Given the performance advantages of MLLMs and the parsing requirements of CTI images, we design a multistage prompt scheme that integrates brainstorming, extraction, verification and integration. As a byproduct, we construct the multimodal threat intelligence dataset AG-LLM-mm. Finally, through detailed experiments, we demonstrate that incorporating CTI images enhances the overall performance of attack graphs and that LLMs hold great potential for multimodal attack graph construction.

⁴https://github.com/WestHealth/pyvis



Figure 8: Example of multimodal attack graph Constrction (MM-AttacKG).

6.2 Future Work

In this work, we present a novel multimodal attack graph construction method and develop an automated image modality information extraction pipeline using LLMs. Although MM-AttacKG is novel and effective, there are still limitations and areas for further research in CTI parsing.

Further Integration of Multimodal Information. Current analysis processes for multimodal threat information separate and individually understand different modalities, but the parsing requirements of different modalities may be interdependent. Therefore, constructing an end-to-end workflow to simultaneously analyze different modalities of threat information and perform cross-verification and support can lead to a deeper understanding of CTI.

Joint Analysis of Multi-Source Threat Reports. There may be event correlations between multiple threat actors, such as different application scenarios and targets of the same malware. By tracking the version iterations of the malware, one can infer the ongoing confrontation between defense strategies and attackers, as well as potential future victims. Analyzing multi-source threat intelligence to build joint analysis strategies can provide deeper insights into the evolution of threats.

Training Domain-Specific Cybersecurity LLMs. General-purpose large models still have gaps in handling complex cybersecurity problems. Therefore, exploring how to pre-train LLMs with domain-specific data and perform targeted optimizations is valuable and can reduce processing costs.

Authorship contribution statement

Yongheng Zhang: Writing original draft, Software, Methodology, Data curation, Conceptualization. Xinyun Zhao: Writing review & editing, Supervision, Formal analysis, Conceptualization. Yunshan Ma, Haokai Ma, Yingxiao Guan: Writing review & editing, Visualization, Investigation, Data curation. Guozheng Yang, Yuliang Lu, Xiang Wang: Writing review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The code and the corresponding dataset will be released upon acceptance.

Funding

The project was supported by Open Fund of Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation.

References

- [1] Zhenyuan Li, Jun Zeng, Yan Chen, and Zhenkai Liang. *AttacKG: Constructing Technique Knowledge Graph from Cyber Threat Intelligence Reports*, pages 589–609. 09 2022.
- [2] Kiavash Satvat, Rigel Gjomemo, and V. Venkatakrishnan. Extractor: Extracting attack behavior from threat reports, 04 2021.
- [3] Peng Gao, Xiaoyuan Liu, Edward Choi, Sibo Ma, Xinyu Yang, Zhengjie Ji, Zilin Zhang, and Dawn Song. Threatkg: A threat knowledge graph for automated open-source cyber threat intelligence gathering and management, 12 2022.
- [4] Muharrem Aksu, Kemal Bicakci, Mustafa Hadi Dilek, Murat Ozbayoglu, and Emin Tatlı. Automated generation of attack graphs using nvd. pages 135–142, 03 2018.
- [5] Ghaith Husari, Ehab Al-Shaer, Mohiuddin Ahmed, Bill Chu, and Xi Niu. Ttpdrill: Automatic and accurate extraction of threat actions from unstructured text of cti sources. pages 103–115, 12 2017.

- [6] Ryan Lee, Montek Boparai, and Tim Duong. Detection of breast cancer lesions using apt weighted mri: a systematic review. *Journal of Translational Medicine*, 23, 01 2025.
- [7] Emma Oye, Edwin Frank, and Jane Owen. Improving apt detection with ensemble learning. 12 2024.
- [8] Karthikeyan Loganathan, Shanna Banda, Lynn Peterson, Carter Tiernan, and Nila Veerabathina. Reframing the role of academic professional track (apt) faculty. 03 2025.
- [9] Mostafizur Rahman Masum. Advanced persistent threat (apt), 10 2024.
- [10] Hamza Abdi, Steven R Bagley, Steven Furnell, and Jamie Twycross. Automatically labeling cyber threat intelligence reports using natural language processing. In *Proceedings of the ACM Symposium on Document Engineering 2023*, pages 1–4, 2023.
- [11] Sheng-Shan Chen, Ren-Hung Hwang, Chin-Yu Sun, Ying-Dar Lin, and Tun-Wen Pai. Enhancing cyber threat intelligence with named entity recognition using bert-crf. In *GLOBECOM 2023-2023 IEEE Global Communications Conference*, pages 7532–7537. IEEE, 2023.
- [12] Yongheng Zhang, Tingwen Du, Yunshan Ma, Xiang Wang, Yi Xie, Guozheng Yang, Yuliang Lu, and Ee-Chien Chang. Attackg+: Boosting attack graph construction with large language models. *Computers & Security*, 150:104220, 2025.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.
- [15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.
- [16] Haoxuan Li, Zhengmao Yang, Yunshan Ma, Yi Bin, Yang Yang, and Tat-Seng Chua. Mm-forecast: A multimodal approach to temporal event forecasting with large language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 2776–2785, 2024.
- [17] Threatcrowd, 2019.
- [18] OpenCTI, 2022.
- [19] AlienVault OTX, 2022.
- [20] Xiaojing Liao, Kan Yuan, XiaoFeng Wang, Zhou Li, Luyi Xing, and Raheem Beyah. Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In *Proceedings of the 2016 ACM* SIGSAC conference on computer and communications security, pages 755–766, 2016.
- [21] Abuse.ch, 2022.
- [22] IBM X-Force Exchange, 2022.
- [23] PhishTank, 2022.
- [24] Roshni R Ramnani, Karthik Shivaram, and Shubhashis Sengupta. Semi-automated information extraction from unstructured threat advisories. In *Proceedings of the 10th Innovations in Software Engineering Conference*, pages 181–187, 2017.
- [25] Yumna Ghazi, Zahid Anwar, Rafia Mumtaz, Shahzad Saleem, and Ali Tahir. A supervised machine learning based approach for automatically extracting high-level threat intelligence from unstructured sources. In 2018 International Conference on Frontiers of Information Technology (FIT), pages 129–134. IEEE, 2018.
- [26] Ghaith Husari, Xi Niu, Bill Chu, and Ehab Al-Shaer. Using entropy and mutual information to extract threat actions from cyber threat intelligence. In 2018 IEEE international conference on intelligence and security informatics (ISI), pages 1–6. IEEE, 2018.
- [27] Kashan Ahmed, Syed Khaldoon Khurshid, and Sadaf Hina. Cyberentrel: Joint extraction of cyber entities and relations using deep learning. *Computers & Security*, 136:103579, 2024.
- [28] Dengtian Mao, Ruike Zhao, Rui He, Pengyi He, Fanghui Ning, and Lingqi Zeng. Threat action extraction based on coreference resolution. In *International Conference on Database Systems for Advanced Applications*, pages 207–221. Springer, 2023.
- [29] Bo Cui, Jinling Li, and Wenhan Hou. Atdg: An automatic cyber threat intelligence extraction model of dpcnn and bigru combined with attention mechanism. In *International Conference on Web Information Systems Engineering*, pages 189–204. Springer, 2023.

- [30] Wenli Shang, Bowen Wang, Pengcheng Zhu, Lei Ding, and Shuang Wang. A span-based multivariate informationaware embedding network for joint relational triplet extraction of threat intelligence. *Knowledge-Based Systems*, 295:111829, 2024.
- [31] Gaosheng Wang, Peipei Liu, Jintao Huang, Haoyu Bin, Xi Wang, and Hongsong Zhu. Knowcti: Knowledge-based cyber threat intelligence entity and relation extraction. *Computers & Security*, 141:103824, 2024.
- [32] Mitre.
- [33] Gbadebo Ayoade, Swarup Chandra, Latifur Khan, Kevin Hamlen, and Bhavani Thuraisingham. Automated threat report classification over multi-source data. In 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC), pages 236–245. IEEE, 2018.
- [34] Wenhan Ge and Junfeng Wang. Seqmask: Behavior extraction over cyber threat intelligence via multi-instance learning. *The Computer Journal*, 67(1):253–273, 2024.
- [35] Chenjing Liu, Junfeng Wang, and Xiangru Chen. Threat intelligence att&ck extraction based on the attention transformer hierarchical recurrent neural network. *Applied Soft Computing*, 122:108826, 2022.
- [36] Yizhe You, Jun Jiang, Zhengwei Jiang, Peian Yang, Baoxu Liu, Huamin Feng, Xuren Wang, and Ning Li. Tim: threat context-enhanced ttp intelligence mining on unstructured threat data. *Cybersecurity*, 5(1):3, 2022.
- [37] Nirnimesh Ghose, Loukas Lazos, Jerzy Rozenblit, and Ronald Breiger. Multimodal graph analysis of cyber attacks. In 2019 Spring simulation conference (SpringSim), pages 1–12. IEEE, 2019.
- [38] Nikolaos Bakalos, Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Avi Ostfeld, Elad Salomons, Juan Caubet, Victor Jimenez, and Pau Li. Protecting water infrastructure from cyber and physical threats: Using multimodal data fusion and adaptive deep learning to monitor critical systems. *IEEE Signal Processing Magazine*, 36(2):36–48, 2019.
- [39] Yuexin Li, Chengyu Huang, Shumin Deng, Mei Lin Lock, Tri Cao, Nay Oo, Hoon Wei Lim, and Bryan Hooi. {KnowPhish}: Large language models meet multimodal knowledge graphs for enhancing {Reference-Based} phishing detection. In 33rd USENIX Security Symposium (USENIX Security 24), pages 793–810, 2024.
- [40] Huixia Zhang, Guowei Shen, Chun Guo, Yunhe Cui, and Chaohui Jiang. Ex-action: Automatically extracting threat actions from cyber threat intelligence report based on multimodal learning. *Security and Communication Networks*, 2021(1):5586335, 2021.
- [41] Nan Xiao, Bo Lang, Ting Wang, and Yikai Chen. Apt-mmf: An advanced persistent threat actor attribution method based on multimodal and multilevel feature fusion. *Computers & security*, 144:103960, 2024.
- [42] Kuanrong Liu, Siyuan Liang, Jiawei Liang, Pengwen Dai, and Xiaochun Cao. Efficient backdoor defense in multimodal contrastive learning: A token-level unlearning method for mitigating threats. *arXiv preprint arXiv:2409.19526*, 2024.
- [43] Shawn Shan, Wenxin Ding, Josephine Passananti, Stanley Wu, Haitao Zheng, and Ben Y Zhao. Nightshade: Prompt-specific poisoning attacks on text-to-image generative models. In 2024 IEEE Symposium on Security and Privacy (SP), pages 807–825. IEEE, 2024.
- [44] Xiaoyu Zhang, Cen Zhang, Tianlin Li, Yihao Huang, Xiaojun Jia, Ming Hu, Jie Zhang, Yang Liu, Shiqing Ma, and Chao Shen. Jailguard: A universal detection framework for llm prompt-based attacks. arXiv preprint arXiv:2312.10766, 2023.
- [45] Andrei Kucharavy, Zachary Schillaci, Loïc Maréchal, Maxime Würsch, Ljiljana Dolamic, Remi Sabonnadiere, Dimitri Percia David, Alain Mermoud, and Vincent Lenders. Fundamentals of generative large language models and perspectives in cyber-defense. arXiv preprint arXiv:2303.12132, 2023.
- [46] Maxime Würsch, Andrei Kucharavy, Dimitri Percia David, and Alain Mermoud. Llms perform poorly at concept extraction in cyber-security research literature. *arXiv preprint arXiv:2312.07110*, 2023.
- [47] Jonathan Pan, Wong Swee Liang, and Yuan Yidi. Raglog: Log anomaly detection using retrieval augmented generation. In 2024 IEEE World Forum on Public Safety Technology (WFPST), pages 169–174. IEEE, 2024.
- [48] Mohamed Amine Ferrag, Mthandazo Ndhlovu, Norbert Tihanyi, Lucas C Cordeiro, Merouane Debbah, and Thierry Lestable. Revolutionizing cyber threat detection with large language models. arXiv preprint arXiv:2306.14263, pages 195–202, 2023.
- [49] PV Charan, Hrushikesh Chunduri, P Mohan Anand, and Sandeep K Shukla. From text to mitre techniques: Exploring the malicious use of large language models for generating cyber attack payloads. *arXiv preprint arXiv:2305.15336*, 2023.
- [50] Maria Rigaki, Ondřej Lukáš, Carlos A Catania, and Sebastian Garcia. Out of the cage: How stochastic parrots win in cyber security environments. *arXiv preprint arXiv:2308.12086*, 2023.

- [51] Stephen Moskal, Sam Laney, Erik Hemberg, and Una-May O'Reilly. Llms killed the script kiddie: How agents supported by large language models change the landscape of network threat testing. *arXiv preprint arXiv:2310.06936*, 2023.
- [52] Ben Kereopa-Yorke. Building resilient smes: Harnessing large language models for cyber security in australia. *Journal of AI, Robotics & Workplace Automation*, 3(1):15–27, 2024.

- [54] Comprehensive Threat Intelligence.
- [55] Microsoft.
- [56] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.
- [57] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- [58] Xinyun Zhao, Yongheng Zhang, Qingying Zhai, Jinrui Zhang, and Lanlan Qi. A multi-attribute decision-making approach for critical node identification in complex networks. *Entropy*, 26(12):1075, 2024.

A

Here we show sample tasks for each stage of the MM-AttacKG runtime: Figure 9 shows the question-answer process for the example image. Figure 10 shows the differences in the answers for the example image with different textual support. Figure 11 shIterationExampleows the process of optimization of the answers. Figure 12 shows CombineAnswerExamplethe process of transforming the answers into reference combinatorial process

B

Here we show the LLMs prompt templates involved in the MM-AttacKG runtime: Table 5 shows the related samples of leading Questions. Table 6 shows the template of the task instruction prompt in different phases, which contain the generation of questions, the answer answering, the evaluation of the answers, the optimization of the answers.

^[53] STIX.



Figure 9: QA Example



Figure 10: Answer Example



Figure 11: Iteration Example

Offset	Name	Func. Count	Bound?	OriginalFirstThun	TimeDateStamp	Forwarder	NameRVA	FirstThunk
7E518	api-ms-win-cor	1	FALSE	7E0E0	0	0	7DA3C	7D480
7E52C	WS2_32.dll	8	FALSE	7E0F0	0	0	7DA30	7D490
7E540	api-ms-win-cor	2	FALSE	7E138	0	0	7DA0C	7D4D8
7E554	api-ms-win-cor	4	FALSE	7E150	0	0	7D9E4	7D4F0
7E568	api-ms-win-eve	3	FALSE	7E178	0	0	7D9B8	7D518
7E57C	api-ms-win-cor	3	FALSE	7E198	0	0	7D994	7D538
7E590	api-ms-win-cor	1	FALSE	7E1B8	0	0	7D970	7D558
7E5A4	api-ms-win-cor	2	FALSE	7E1C8	0	0	7D944	7D568
7E5B8	api-ms-win-cor	2	FALSE	7E1E0	0	0	7D918	7D580
7E5CC	api-ms-win-cor	1	FALSE	7E1F8	0	0	7D8E8	7D598
7E5E0	api-ms-win-cor	3	FALSE	7E208	0	0	7D8C4	7D5A8
7E5F4	ntdll.dll	5	FALSE	7E228	0	0	7D8B8	7D5C8
7E608	api-ms-win-cor	1	FALSE	7E258	0	0	7D894	7D5F8
7E61C	api-ms-win-cor	2	FALSE	7E268	0	0	7D86C	7D608
7E630	wmiwk.dll	1	FALSE	82258	0	0	82268	82258
wmiwk.dll	[1 entry]							
Call via	Name	Ordinal	Original Thunk	Thunk	Forwarder	Hint		
82258	StreamHandler	-	82272	82272	-	1		

//_ Answer

The main content of the image is a table detailing the Import Address Table (IAT) entries, including the appended entry for wmiwk.dll that forces the loading of a malicious DLL when the WMI service is initiated.

- The possible uses of the image are to illustrate how attackers modify a wbemcomn.dll file to load a malicious StealthVector DLL named wmiwk.dll during the ScrambleCross infection chain.

- The possible attack techniques involve modifying the Import Address Table (IAT) to force the loading of a malicious DLL named wmiwk.dll when the WMI service is initiated.

- The highlighted fields include "7E630" in the Offset column, "wmiwk.dll" in the Name column, and "82258" in both the OriginalFirstThunk and FirstThunk columns for the entry related to wmiwk.dll, as well as "82258" and "StreamHandler" in the Call via and Name columns of the second table.

- The possible informations relevant to malicious activity in the image are the presence of wmiwk.dll with a single entry for StreamHandler at offset 7E630, indicating a potential hijacking of WMI functionality.

The significance of the wmivxk.dll entry in the image is that it represents a modified Import Address Table (IAT) entry with an OriginalThunk value of 82258, which forces the loading of the malicious wmiwk.dll DLL when the WMI service is initiated, facilitating DLL injection as part of the ScrambleCross infection chain.

The details depicted by the Offset values in the image represent the starting addresses of entries in the Import Address Table (IAT) for each imported function, such as the Offset value '7E630' for 'wmiwk.dll', which is relevant to cybersecurity as these values are used in the context of DLL injection and malware execution.

The role of the StreamHandler function (Original Thunk at 82272) is part of the malicious wmiwk.dll, which is loaded through an appended IAT entry as indicated in the image, and this loading mechanism is a key component of the ScrambleCross infection chain used for executing malicious shellcode via a legitimate WMI service

- The implications of the FALSE Bound status in the image indicate that the DLLs, including wmiwk.dll, are not properly secured, which can be exploited to load malicious code during WMI service initialization as part of the ScrambleCross infection chain.

The significance of the MoonBounce malware in the image is that it demonstrates a technique used to load a malicious DLL named wmiwk.dll

into the WMI service, facilitating the execution of malicious shellcode through a legitimate system service. - The presence of UEFI firmware modifications implies the loading of a malicious `wmiwk.dll` through an appended IAT entry in the modified `wbemcomn.dll` file, as shown in the image, indicating a potential ScrambleCross infection chain.

APT41's tactics include modifying the Import Address Table (IAT) to force the loading of a malicious DLL named wmiwk.dll when the WMI service is initiated.

The defensive evasion technique depicted is the appending of an IAT entry to force the loading of a malicious StealthVector DLL named wmiwk.dll when the WMI service is initiated

- The image illustrates the persistence mechanisms employed by APT41 by showing an appended IAT entry that forces the loading of a malicious StealthVector DLL named wmiwk.dll when the WMI service is initiated.

Answer-Rewrite

The image depicts a cyber threat scenario involving the modification of the Import Address Table (IAT) to force the loading of a malicious DLL named wmiwk.dll during the initialization of the WMI service, which is part of the ScrambleCross infection chain used by APT41. This malicious activity is facilitated by altering the wbemcomn.dll file, resulting in the loading of wmiwk.dll, also associated with MoonBounce malware, through an appended IAT entry. The table highlights key entries, including of Winiwk.ui, also associated with Moonbounce maware, through an appended IAT entry. The table highlights key entries, including offset 7E630 for wmiwk.dll and the OriginalFirstThunk and FirstThunk value of 82258, indicating potential functionalities like StreamHandler, which is part of the malicious payload executed via WMI. The implications of this technique include possible DLL injection and execution of malicious shellcode, demonstrating a defensive evasion approach and persistence mechanism. The FALSE Bound status suggests a lack of proper security measures, further facilitating the exploit. Additionally, UEFI firmware modifications imply the presence of the malicious wmiwk.dll entry, underscoring the sophisticated tactics used by attackers to integrate malicious code into legitimate system services.

Figure 12: Combine Answer Example

Table 5: Leading Que	estions.
----------------------	----------

Attack Flow or Intelligence Structure	 What is the main content of the image? What are the possible uses of the image? What are the possible attack techniques involved in the image? What is the sequential relationship of the processes in the image? What are the possible targets of attack in the image?
Malware Code	 What is the main content of code in the image? What are the possible uses of the image? What are the possible attack techniques involved in the image? What is the possible function of the Code in the image? What are the possible variables in the Code in the image?
Application Tool Screenshot	What is the main content of the image?What are the possible uses of the image?What are the possible attack techniques involved in the image?What is the key highlighted information in the picture?
Data Table	 What is the main content of the image? What are the possible uses of the image? What are the possible attack techniques involved in the image? What are the fields highlighted in the image? What are the possible informations relevant to malicious activity in the image?
Charts and Data Visualization	 What is the main content of the image? What are the possible uses of the image? What are the trends reflected in the image? What are the conclusions that can be made based on the image?
File Paths and Names	 What is the main content of the image? What are the possible uses of the image? What are the possible attack techniques involved in the image? What paths are included in the image?
Descriptive Image and Content Explanation	What is the main content of the image?What are the possible uses of the image?What are the possible attack techniques involved in the image?

Question Generation	 You are a cybersecurity threat intelligence analyst. Please generate relevant questions derived from the content of an image based on the following rules: 1. Review the list of existing questions provided and generate new questions that explore different perspectives, details, or contexts within the image. 2. Generate new questions that explore different perspectives or details of the image based on the information from the knowledge graph. 3. These new questions should help further analyze the image from different perspectives. 4. These new questions should be related to cybersecurity or assist in the analysis of cyber threat intelligence. 5. Refer to the format of the given questions, which follows the pattern: "What is/are the XXX of/in the image?", "Where XXX should be replaced with a specific aspect of the image?".
Question Answering	 You are a cybersecurity threat intelligence analyst. Please answer the questions based on the following rules: 1. Your answer must strictly adhere to the content visible in the image when mentioning any entities, objects, and their relationships. 2. Your answer must include a topic phrase that is specific to the question. 3. Your answer should be a single, concise sentence. 4. Only provide the direct answer to the question. Do not provide explanations or reasons for uncertainty.
Answer Evaluation	 You are a cybersecurity threat intelligence analyst. Please rate the description based on the following rules: 1. Evaluate the description using the following four criteria: Accuracy: Accuracy represents whether the description accurately answers the question. Consistency: Consistency represents whether the description maintains content relevance to the image information. Completeness: Completeness represents whether the description adequately addresses the needs of the question. Relevance: Relevance represents whether the description is relevant to the cybersecurity field or useful for cyber threat analysis. Apply the following rating scale based on the overall quality: "excellent": The description meets three of the criteria with only minor flaws or imperfections. "good": The description meets two of the criteria, but contains more noticeable flaws. "failing": The description meets only one of the criteria or none at all, with significant flaws that make the response unable to provide useful or relevant information. If there are statements in the description such as unknown, no details, not mentioned, etc., mark it as "failing". Your answer should be a single word: either "excellent", "good", "satisfactory", or "failing".
Answer Optimization	 You are a cybersecurity threat intelligence analyst. Please provide your answers to the following questions again, using the image as a reference, based on the following rules: 1. Re-answer the questions to ensure the answers meet the following four criteria: - Accuracy: Accuracy represents the answer accurately answers the question. - Consistency: Consistency represents the answer maintains content relevance to the image information. - Completeness: Completeness represents the answer adequately addresses the needs of the question. - Relevance: Relevance represents the answer is relevant to the cybersecurity field or useful for cyber threat analysis. 2. Improve existing unqualified answers (Paradigm 1) or re-answer questions based on suggestions provided (Paradigm 2). 3. Ensure the revised answer differs from the previous unqualified answer (Paradigm 1), or strictly follows the suggestions given (Paradigm 2).

Table 6: Prompts of image function identification module.

Table 7: Prompts of ICL.

Extraction of Entities and Relation	You are a cyber-security information-extraction specialist. Given an image, its context, and a CTI summary, identify entities and their relationships, then output them as triplets: Subject (Type); Relation; Object (Type). [Input] 1. An image; 2. Image In-context; 3. CTI summary; 4. Entity Types & Descriptions; 5. Relation Table. [Example] Input: Image: attack path from public Internet → SSH→ Public-facing Server1 In-context: This led our responders to identify the occurrence of CTI summary: CrowdStrike's analysis of the StellarParticle campaign Output: Public Internet(infrastructure); communicate-with; Public-facing Server1(infrastructure)
Extraction of Techniques	You are a cyber-threat intelligence specialist. Given an image, its context, and a CTI summary, identify the one MITRE ATT&CK tactic that best matches the image. [Input] 1. CTI image; 2. Image context; 3. CTI summary; 4. MITRE ATT&CK tactics list. [Example] Input: Image: attack path from public Internet → SSH→ Public-facing Server1 In-context: This led our responders to identify the occurrence of CTI summary: CrowdStrike's analysis of the StellarParticle campaign Output: Lateral Movement
Integrated Attack Graph	You are a cyber-threat intelligence specialist. Given a CTI image, and a Knowledge Graph of existing triplets, perform one of three operations based on the image: 1. New Node Addition: add a new entity to extend an existing triplet. 2. New Relationship Addition: link entities from different triplets. 3. Technique Addition: tag an existing triplet with a new MITRE ATT&CK technique. [Input] 1. CTI image; 2.Knowledge Graph (list of triplets); 3. Entity Types & Descriptions; 4.MITRE ATT&CK techniques list. [Rules] Rule 1: New Node Addition: if the image shows an entity related to a KG triplet, output a JSON list [] with objects containing: description: reason for adding the node new_node: {id, type, properties:{description}} relationship: {Subject, SubjectType, Relation, Object, ObjectType} Rule 2: New Relationship Addition: if the image connects entities from different triplets, output objects with: description: reason for the new relation relationship: {Subject, SubjectType, Relation, Object, ObjectType} Rule 3: Technique Addition: if the image indicates a new MITRE technique for a triplet, output objects with: [Example] Input: Image: attacker executes Stuxnet injection, installs a malicious module Knowledge Graph: triplets for Stuxnet → install → dropper; attackers → develop → capabilities Entity Types: includes campaign MITRE list: includes T1055-Process Injection Output: ["type":"new_node_addition",}, {"type":"new_relationship_addition",}, {"type":"new_relationship_addition",}, {"type":"new_relationship_addition",}, {"type":"rechnique_addition",}, }

Table 8: Prompts of CoT.

Extraction of Entities and Relation	 You are a cybersecurity information-extraction specialist. Given a CTI image (plus its context and summary for reference), identify visible entities and their relationships, then output them as triplets: Subject(type); relation; Object(type). [Thinking Process] 1. Spot entities in the image. 2. Map each to the shortest matching provided type. 3. Identify clear, active-voice relationships between them. 4. Form triplets.
Extraction of Techniques	 You are a cyber-threat intelligence expert. Given a CTI image (with optional context/summary) and a list of MITRE ATT&CK tactics, identify the single tactic that best matches the image. [Thinking Process] 1. Inspect the image for attack indicators (flowcharts, logs, interfaces). 2. Identify candidate tactics from the provided list. 3. Select the one tactic that most directly reflects what you see. 4. Ensure it fits the attack phase implied by the image.
Integrated Attack Graph	You are a cyber-threat intelligence expert. Given a CTI image, a Knowledge Graph of triplets, Entity Types & Descriptions, and MITRE ATT&CK techniques, extend the graph by discovering: 1. New Node Addition: adding an image-derived entity to an existing triplet. 2. New Relationship Addition: linking entities from different triplets. 3. Technique Addition: tagging a triplet with a new MITRE technique. [Thinking Process] 1. Identify triplets strongly correlated with the image. 2. Extract and type entities visible in the image. 3. For each strong-match triplet: check if an image entity adds a new connection to its subject or object (new node addition). 4. Check if an entity in this triplet should link to an entity in another triplet (new relationship addition). 5. Determine if the image implies a new MITRE technique for the most relevant triplet (technique addition). 6. Assemble JSON outputs. 7. If none apply, output No Match. [Rules] Rule 1: New Node Addition: if the image shows an entity related to a KG triplet, output a JSON list [] with objects containing: description: reason for adding the node new_node: {id, type, properties:{description}} relationship: {Subject, SubjectType, Relation, Object, ObjectType} Rule 2: New Relationship Addition: if the image indicates a new MITRE technique for a triplets, output objects with: description: reason for the new relation relationship: {Subject, SubjectType, Relation, Object, ObjectType} Rule 3: Technique Addition: if the image indicates a new MITRE technique for a triplet, output objects with: description: reason for the new technique target_relationship; {Subject, Relation, Object, ObjectType} Rule 3: Technique Addition: if the image indicates a new MITRE technique for a triplet, output objects with: description: reason for the new technique target_relationship; {Subject, Relation, Object} new_techniques: ["technique_id - technique_name"] Rule 4: Use active-voice, concise verb phrases for relations. Rule 5: Only generate JSON if there is a strong mat