Large Language Models for Cybersecurity: New Opportunities

Dinil Mon Divakaran, *Institute for Infocomm Research (I²R), A*STAR*

Sai Teja Peddinti, *Google*

Abstract—Large language models (LLMs) are a class of powerful and versatile models that are beneficial to many industries. With the emergence of LLMs, we take a fresh look at cyber security, specifically exploring and summarizing the potential of LLMs in addressing challenging problems in the security and safety domains.

Index Terms: *LLM, Deep Learning, Security, Vulnerabilities, Safety*

Large Language Models (LLMs) are creating a transformational impact in the space of science and technology, giving rise to a wide variety of new applications for various services across diverse industry verticals. Their capability to comprehend and, in particular, to generate contents, represents a paradigm shift that is reshaping the way we interact with computers, leading to the development of numerous innovative applications. Today, LLMs are able to generate text, images, and videos; there are LLM applications that hold conversations with humans, translate between languages, explain and write code, resolve programming bugs, and so forth.

LLMs generally are based on a transformer architecture that uses self-attention mechanism to efficiently learn long-range dependencies of tokens (words or sub-words) in a sequence of data (e.g., a sentence). This has allowed transformer models to not only improve upon previous sequence models such as RNNs (Recurrent Neural Networks), but also to train large models of billions and even trillions of parameters on datasets of massive sizes. Importantly, the *pretraining* of an LLM is unsupervised, removing the burden of labeling large datasets. Like other generative models, LLMs fundamentally aim to recreate data they are trained on. Using these properties, pretrained LLMs have been used to generalize across many tasks, often by fine-tuning on small amounts of labeled data. GPT-4, Gemini, Llama 2, Mistral, Falcon, OLMo (Open Language Model), etc., are some of the well-known LLMs today, while new ones are being built at a rapid pace. Examples of downstream tasks include language translation, sentiment analysis, domain-specific chatbot conversation, text based image/video generation, assistive medical diagnosis, etc.

Unsurprisingly though, such a compelling technology can be put to dual use. An LLM is fundamentally a probabilistic model, which learns to make predictions based on the massive datasets that it has been trained on; and thus, it is only reasonable that the model may not consistently generate factually accurate, benign, or positive outputs, even if trained to do so. This inherent characteristic can be exploited, e.g., via prompt injection attack (discussed later), by malicious actors for various purposes. We refer the reader to the 'NIST Trustworthy and Responsible AI report (2023)', for a detailed taxonomy of adversarial machine learning (ML) in the context of both conventional ML as well as LLMs.

There are ongoing efforts to mitigate the risks due to LLMs. Companies such as OpenAI (https://openai.com/safety), Google (https://safety.google/cybersecurity-advancements/saif/), Meta (https://ai.meta.com/responsible-ai/), Microsoft (https://www.microsoft.com/en-us/ai/responsible-ai), etc. have frameworks for developing safe and responsible AI systems. In fact, many of the firms also focus on *red teaming* LLMs, to proactively investigate and identify vulnerabilities of LLMs, e.g., to detect adversarial prompts that can generate harmful or malicious responses. In 2023, Microsoft, Anthropic, Google, and OpenAI launched the *Frontier Model Forum* to support best practices to mitigate risks, advance research on AI safety and security, as well as facilitate information sharing among companies and governments. Similarly, companies formed a *C2PA coalition* to create an open technical standard that will aid in the ability to trace the origin of different types of generated media. Lastly, governments across the world are also working on regulatory frameworks for AI, to protect AI users and user privacy (among

others). It is worth noting that, governments are encouraging global collaborative efforts to tackle AI vulnerabilities and security risks (e.g., refer the U.S Executive Order on the 'Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence', and the European Union's 'AI Act'). Despite these efforts, mitigating LLM risks is still an unsolved problem. The emergence of underground LLM market places for malicious services [1], the recent rise in 'Deepfake' impersonation scams, and the evolving jailbreaking and prompt injection attacks attest to the complexity of this evolving landscape.

New opportunity to address cyber security problems

We now turn to the main focus of this article and discuss the new opportunities LLMs present in addressing security and safety challenges that users today face in the digital world. The cyber security domain has already started to see the benefits of utilizing LLMs for addressing some of the important problems in the domain, and we summarize some of these recent advancements. These efforts can be broadly categorized into five themes described below. Refer to Figure 1 for an overview.

LLMs for Vulnerability Detection and Management

Today, there are multiple LLM-based tools that are being built to help with code development. Devin AI, GitHub Copilot, IBM's watsonx, Amazon CodeWhisperer and Codeium are some of the emerging AI code assistants. They perform tasks such as code generation and completion, code repair, code refactoring, and code explanation. Besides lowering the entry barrier for software development, these code assistants help in reducing bugs in software development process. For instance, propagating changes in variable type *automatically*, although appears simple, is a particularly useful feature that helps developers.

The number of CVEs published has doubled over a period of six years reaching close to $29,000$ in 2023 (refer the CVE portal at www.cve.org). The 2024 Open Source Security and Risk Analysis report from Synopsis says that 74% of the codebases that they examined had high-risk vulnerabilities. Software vulnerabilities lead to system failures, and malicious actors target the vulnerabilities to launch cyber attacks. The process of addressing software vulnerabilities consists of two phases: vulnerability discovery and patching the discovered bugs. There are ongoing research works on these two tracks.

Fuzz4all (refer https://fuzz4all.github.io/ for the research paper and code) is a system that uses an LLM to generate prompts that subsequently feeds into another LLM which then creates different fuzzing inputs. This is an example where the 'creativity' of LLMs is useful in generating different inputs to fuzz a system under test. Fuzz4all already detected 98 bugs across 9 systems including GCC and Z3.

To evaluate automated code repair, a benchmark dataset called SWE-bench consisting of 2,294 real-world engineering tasks (GitHub issues) from 12 popular Python repositories was created. While the initial solutions (evaluated in Oct. 2023) could resolve only 3% of the tasks, the best solution a year later takes that number to 43%. We refer the readers to the SWE-bench website for further details (www.swebench.com/). An example solution is AutoCodeRover (https://autocoderover.dev/), which uses LLMs to analyze the GitHub issue, understand the code context, and generate a patch.

We also highlight the report from Google sharing that its Gemini model helped to fix 15% of bugs discovered by their sanitizer tools, resulting in hundreds of bugs patched [2, Section 5]. Also, competitions such as the *AI Cyber Challenge*, a two-year competition announced in late 2023, organized by DARPA in collaboration with others to design and develop AI-based solutions to secure code, have given momentum to this line of research.

The above developments are promising; yet it is important to note that the current evaluations are limited to a small number of benchmark datasets, focusing on a few programming languages. There is still a long way to go, to be able to automatically discover and fix vulnerabilities in critical systems and large codebases.

LLMs for Content Classification and Enforcement

LLMs are being leveraged to augment or automate several general purpose security/safety classifiers, some of which are described below.

Safety Classifiers for Policy Enforcement:
Toxic contents are on the rise on online platforms. Hate speech, harassment, cyber-bullying, etc. adversely affect users of all communities, and in particular underrepresented groups. The complexity of this socio-technological problem is amplified by the multilingual nature of communications, the use of evolving lingo, emojis, styles, and so forth. One of the well-known classifiers for toxic content detection that is used by
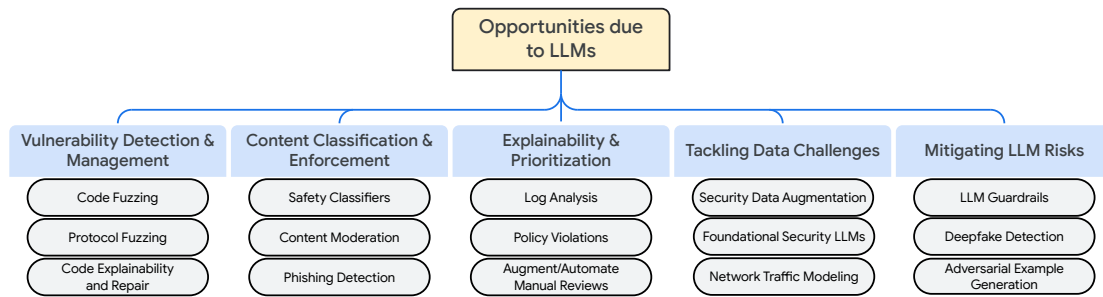
FIGURE 1: LLMs offer versatile solutions to address a wide range of cyber security challenges.

developers and publishers is Google Jigsaw's Perspective API (https://perspectiveapi.com/). The collaborative team has been publishing tools and data, besides improving the model capabilities. There are also a number of ML models proposed in the literature to address this issue.

Despite the active research in toxic content detection, the scarcity of large-scale, high-quality data impedes research. However, LLMs pretrained on massive data offer a promising direction. As noted previously, LLMs have the capability to solve downstream tasks with a small number of labeled samples, or even without fine-tuning. Indeed, He *et al.* show that, with prompt learning–giving a few examples at an LLM's prompt, pretrained LLMs are able to achieve better performance than models trained specifically for toxic content detection [3]. That said, the problem is far from being solved. We have to develop solutions that extend beyond text analysis to detect toxicity in various media formats, including images, audios, videos, and obfuscated messages. Continued research in the field of LLMs, aimed at enhancing their capability to perform on tasks across diverse content formats, holds the potential to offer new solutions for combating toxic content in online platforms.

Another area where LLMs are useful is content moderation. Content safety policies often evolve too frequently to catch-up with the different types of threats emerging online. LLM's *zero-shot* capabilities are immensely valuable in quick enforcement of these evolving policies, or for reducing labeling costs when creating annotated datasets for training down-stream ML models. Kumar *et al.* [4] show that LLMs (such as GPT-3.5) are effective at rule-based moderation for many Reddit communities, achieving performance close to human moderators for some communities. This early result motivates exploring LLM use for content moderation in other settings.

Phishing Detection:

Phishing is one of the most common cyber attacks in recent times. Attackers craft and send phishing emails to victims, often including text, image (e.g., brand logo) and a URL to a phishing website. Phishing emails can be targeted to specific individuals (say, a person in the Finance department of a company), and links to phishing websites are also distributed via social media, chats, SMSes, etc. This also presents multiple options for solution development. For example, specific phishing detection solutions are integrated with email and SMS gateways. Also, threat intelligence services get URLs from various sources and analyze them using standalone services. A popular service is VirusTotal, which utilizes more than 70 URL-analyzing engines from cyber security vendors and provides aggregate results to users. Despite these protections in place, many (carefully crafted) phishing emails are evading these scanners and reaching users' mail boxes.

Phishing emails. Over the years, phishing email solutions have evolved from relying solely on rules and signatures to the use of ML models to automatically learn patterns of phishing emails. Recently, we also see the use of LLMs for addressing this threat. Koide *et al.* [5] created *ChatSpamDetector*, that utilizes LLMs to detect phishing emails and obtain detailed reasoning for the phishing determination. This system is shown to outperform existing baseline detection systems, does not require continuous updates to the detection models and block lists like in existing spam filters, and the generated rationales assist users in making informed decisions when handling suspicious emails.

Phishing webpages. A well-known approach to detecting phishing webpages, called *reference-based approach*, is to compare the logos on a given webpage to a known *reference* set of logos of popular brands (e.g., Paypal, Amazon, etc.). The basic idea in reference-based approach is that, if a webpage contains a well-known brand's logo (e.g., Paypal's) but has a different domain name, then it is a phishing page.

The state-of-the-art solution, Phishpedia [6], trains an object-detection model to detect the logos on screenshots of webpages and a Siamese model to identify the brand of a detected logo.

We now have an opportunity to rethink how we address the challenging problem of phishing. Lee *et al.* takes a clean-slate approach to phishing detection [7], where a pipeline of two multimodal LLMs is used—the first one to identify the brand of a given webpage, and a subsequent LLM to check if there is a match between the identified brand and the domain in the webpage's URL. If there is *misalignment* between the identified brand and the domain name in the URL, it is considered phishing. The research shows promise in detecting phishing webpages, and importantly, also providing an explanation for the decision. Further investigations are required to understand if an LLM-integrated phishing detection system can detect attacks targeting regional brands that might not be well-represented by an LLM. Yet another challenge is to detect phishing pages in uncommon and local languages.

LLMs for Explainability and Prioritization

LLMs, with their natural language interface and the ability to work with data in multiple modalities (text, images, videos, code, etc.), can help with understanding diverse data. Newer LLMs, such as Google's Gemini Pro 1.5 and Anthropic's Claude 3 Haiku, boast extremely large context windows of more than 100,000 tokens, enabling them to digest and summarize large amounts of data. These capabilities have opened up new avenues of utilizing LLMs for data explainability, summarization, and for automating or augmenting human reviews.

Explainability:
Enterprises deploy security solutions from one or more vendors to protect their endpoints. To gain high visibility, modern security solution providers gather detailed data from processes, network connections, applications, file/registry accesses, etc., thus resulting in humongous logs. SentinelOne Singularity, CrowdStrike Falcon and Trend Micro Apex are examples of commercial EDR (endpoint detection and response) solutions. Besides the logging capability, EDR solutions also come with a set of rules to detect malicious patterns of known malware. Similar problem also exists in the cloud and distributed computing systems. For example, the promising microservice architecture that helps to scale up resources as required for an application,

also comes with threats due to insecure packages, misconfigured authentications, etc. The large attack surface exposed due to the distributed nature of the architecture makes it all the more relevant to log information and analyze them in real-time for timely detection of anomalies and attacks.

As traditional approach of writing rules to match malicious patterns neither scales nor achieves high detection accuracy, and so security researchers are developing ML models that train on huge amounts of process/audit logs to detect suspicious behaviors. However, this creates another challenge—the detected patterns from the endpoints need to be investigated by security analysts to take the appropriate mitigation steps. Besides, ML models also raise false positives; and a high number of patterns that need to be investigated leads to *alert fatigue*, which in turn results in missing out high-risks threats and attacks. Cyber defenders' burn-out is a known chronic problem [2, Section 3]. LLMs are currently being used to explain the detected patterns, to make it easier for an analyst to decide quickly. For example, *HuntGPT* [8] is a specialized intrusion detection dashboard that uses LLMs to discern patterns in network traffic and deliver detected threats in an understandable format. Powered by GPT-3.5-turbo, the system achieved more than 80% success rate at the CISM (Certified Information Security Manager) Practice exams, showing promise in guiding security decisions. Other examples from a recently published Google report [2] include the following. i) The Google Detection & Response teams have leveraged Gemini LLM for natural language querying and automatic summarization of alerts data, and have seen a 51% time savings and higher quality results in incident analysis. ii) Google Cloud's SecLM, a security-specific LLM, facilitates analysts to conversationally search and interact with security events, provides explanations for complex attack graphs, and even recommends mitigations. Similarly, VirusTotal Code Insight explains what a potentially malicious Powershell code is doing, and solutions such as CrowdStrike's Charlotte AI, Google Cloud's DuetAI, and Microsoft's Security Copilot also aim to empower security analysts in their threat hunting process. Such assistive solutions can help even non-expert security analysts to detect, investigate, and respond to cyberthreats with confidence.

Performing content moderation across online platforms has very similar challenges, where human reviewers have to investigate a multitude of (ML or user) flagged posts for policy violations. Kumar *et al.* [4] show that the reasoning capabilities of LLMs are immensely useful in providing explanations and in identifying the specific rules being violated by the policy violating

posts, making LLMs a valuable aid for humans performing content moderation.

Prioritization:
LLMs are also very useful in automating or augmenting manual reviews, and help reduce a reviewer's fatigue when sifting through detected security incidents or flagged online content. They help evaluate the veracity of identified incidents or policy violations, automating decisions in clear cases and triaging/escalating high risk, complex, or borderlines cases to help focus engineering/expert resources efficiently. For instance, Qiao *et al.* [9] employed LLMs to scale up content moderation in Google Ads. They were able to reduce the number of manual reviews by more than 3 orders of magnitude while achieving a 2x recall compared to a baseline non-LLM implementation.

Automated decision making of LLMs also helps reduce exposure of human reviewers to harmful content, thereby enhancing their mental well-being. Puentes *et al.* [10] propose a Large Language Model (LLM) that analyzes and classifies the information received in reports on sextortion, sexting, grooming, and sexual cyberbullying. Their system even efficiently forwards the reports to competent authorities, and reduces the exposure of analysts to harmful contents.

Despite LLM's strengths in content summarization, explanability, and automation, they are known to be prone to hallucinations—where they generate responses that are factually incorrect, nonsensical, or disconnected (from inputs). Research focusing on 'grounding' the LLMs to the provided data can alleviate these concerns.

LLMs for Tackling Data Challenges

Building highly accurate ML models for security and safety use cases requires large labeled datasets. In the domain on cyber security, there are two challenges in obtaining quality datasets for training models.

- Labeling Cost: As in many domains, labeling is a costly task requiring human effort. To develop ML models for solving security problems (such as detection of network attacks, malware detection via static and dynamic analyses, etc.) requires large labeled datasets. While the research community publishes data once in a while, they are limited in size, may contain artifacts (e.g., malicious datasets for network attacks and endpoint logs for malware analysis are often generated via emulation in a controlled environment), or may be obsolete.
- Data Privacy and Retention: Another challenge

in obtaining real-world dataset is the risk of leaking sensitive or confidential information. Consider email data (required for phishing detection), social media data (required for content moderation), network traffic, etc., where there is risk of privacy leak. On the other hand, to provide privacy guarantees, companies often employ retention timelines when storing user data, that indicate how long the data can be stored and used. Often these retention requirements also get applied to the manually annotated training data, when it is derived from user data. For instance, consider the case of a toxicity model trained on social media data. Based on the policy that *a user's data would be deleted from the social media website within a week after they delete their account*, the toxicity model would start forgetting patterns seen across deleted users' data. For model performance benchmarking over time and to avoid forgetting patterns observed in old data, permanent access to annotated training data is necessary.

Given the above challenges, LLMs are being explored for data augmentation needs. Data augmentation techniques help with diversifying training examples without the need for additional data collection or labeling. For instance, Lee *et al.* [11] have proposed *LLM2LLM*, an iterative data augmentation strategy to enhance a small-seed dataset, and have demonstrated that this reduces dependence on labor-intensive data curation while simultaneously achieving improvements over regular fine-tuning in low-data regime tasks. Others are leveraging LLMs for augmenting training datasets in new languages (to enhance cross-lingual performance of base models), or are exploring synthetic data generation approaches for completely skipping training data annotation. To avoid any privacy leaks, LLMs are also being fine-tuned on sensitive datasets in a 'differentially private' way. While these data augmentation techniques show promise, they do not completely solve the challenges. For instance, Akkus et al. [12] conduct an empirical analysis and show that fine-tuning on LLM generated data does not completely avoid privacy leakages.

Traffic Modeling for Network Security:
However, data augmentation alone isn't sufficient. Consider network traffic analysis for detection of various threats, anomalies and attacks. Years of research works have led to the development of numerous statistical and ML models for specific network security tasks, such as detection of bots, C&C channels used for communication between attacker and compromised hosts,

low-rate DDoS attacks, password-spraying attempts, generic anomalies, etc. Each of these tasks require large amounts of labeled data with minimum noise for training accurate ML models. For example, to train a model for detecting bot traffic to an e-commerce website, the dataset has to have hundreds of thousands of labeled network requests that are made by both bots and legitimate users. Yet, it is arguable whether such a dataset helps in building models that can generalize well, given data can come from different operating systems, browsers, locations, etc. Therefore, to generalize, and even to sustain model by retraining, such an e-commerce entity would have to label network traffic regularly.

The advancements in LLM development present a new opportunity to train domain-specific foundational models in an unsupervised way. In the network traffic analysis example, a network-specific foundation model that learns network 'conversations' (e.g., requests and responses) can be trained using openly available real-world network traffic datasets. CAIDA and MAWI, for example, continuously publish network traces for research purposes; while being massive in size, they are mostly unlabeled. But these unlabeled datasets can be utilized for training a foundation model in an unsupervised way. Such a powerful model can then be fine-tuned for multiple downstream tasks, such as botnet detection. Although fine-tuning is a supervised approach, it typically requires only small amounts of labeled data, thereby decreasing labeling costs significantly. The network research community is witnessing active discussions in this direction, of training an LLM that learns network communication language (see ACM HotNets 2022 and 2023 proceedings). Research efforts are required to come up with good representations of network traffic for training an effective foundation model, that can then be utilized for multiple downstream security tasks.

LLMs for Mitigating LLM Risks

With their generative capability, LLMs have lowered the entry barrier for cyber criminals. Phishing emails, tailored to specific roles or individuals, can be generated using LLM applications such as ChatGPT easily. Researchers at CyberArk outlined how to generate polymorphic malware; the malware runs with ChatGPT API generating new payloads and malicious modules as and when required to evade detection. Security researchers have already discovered generative AI tools in the dark web marketplaces that help attackers with their cyber criminal activities [1]; examples include FraudGPT and WormGPT. And attackers are exploiting the capability of LLMs to generate highly realistic and convincing images, videos, and audio to create *Deepfakes*. Deepfakes are already being used for unethical and malicious purposes such as spreading misinformation, generating fake news, and scamming or defaming individuals. Microsoft lists a number of threat actors that have adopted generative AI tools to launch recent attacks (refer https://aka.ms/emerging-AI-threats).

While the above attacks are not novel *per se*, their proliferation is enabled by LLMs, specifically due to a new attack vector of LLMs, namely *prompt injection*. In this attack, an attacker exploits the ability to query LLM models through well-defined APIs and interfaces to either *extract* sensitive information (such as application product keys), or enable scope for other threats such as remote code injection. The attack surface increases when an LLM is extended with data sources to provide more up-to-date information via retrieval augmented generation (RAG), thereby blurring the line between instruction and data. An example is of an attacker sending an email with malicious instructions that are automatically fed to an LLM application meant for detecting spam or phishing emails, but then inadvertently follows the attacker's instructions. Prompt injection attack is recognized as the top LLM related attack by OWASP; and they are of particular concern when new applications interface with an LLM for automated responses.

To negate the above mentioned LLM risks and vulnerabilities, there is also research studying and deploying a multitude of security risk mitigation strategies, including defining and applying strict policies for moderating the input and filtering the output. One approach is to have safeguard checks and controls, also termed as *guardrails*, in place. For example, safety filters in text-to-image models, such as DALL-E 2 and Midjourney, prevent generating not-safe-for-work (NSFW) content. *LlamaGuard* from Meta is an LLM trained to classify an LLM prompt or a response as safe. There are also independently developed guardrail solutions focusing on a specific data type and task, such as unsafe image detectors (e.g., https://github.com/LAION-AI/CLIP-based-NSFW-Detector). Countering the challenge of exploiting AI-generated contents (Deepfakes) for fraudulent purposes is an active area of research within the AI domain, and one of the interesting research directions is to add watermarks to contents generated by LLMs. Developing such guardrail solutions is a challenging and ongoing effort, as they have to catch up to different models, applications and evolving policies. In addition, research has shown that these guardrails are also vulnerable. For instance,

the guardrails in DALL-E 2 and Midjourney around generating NSFW images can be easily bypassed through prompt injection attacks [13]. Therefore, while these guardrails do raise the bar in preventing LLM misuse, developing effective protections remains an open problem.

ML-based defense solutions are susceptible to evasion attacks. A well-studied approach to counter such evasions is *adversarial training*, where training with adversarial examples can enhance the robustness of defense models against evasion attacks. With their generative capabilities, LLMs are being leveraged to automate the generation of adversarial examples with little human effort [14]. These adversarial examples can then be incorporated into training to build potentially robust ML models to defend against threats and attacks. However, it is to be noted that this state-of-the-art defense technique also has limitations: cost of generating adversarial examples is not cheap, and prior research has shown that adversarial training is susceptible to 'blind-spot' attacks, where input examples far away from the embedded training data are still vulnerable to attacks [15].

Key takeaways

There is an inherent asymmetry between the attackers and defenders in the cyberspace, popularly referred to as "Defender's Dilemma", which states that it is sufficient for an attacker to succeed once but a defender must be successful in protecting at all times [2]. Machine Learning and Artificial Intelligence (AI), and specifically Large Language Models, have the potential to tilt the scales of cyberspace to give the defenders an advantage over the attackers. The emergence of LLMs presents an opportunity to reimagine how we approach and solve cyber security challenges, enabling the development of innovative solutions by leveraging the capabilities of these powerful models. There are early works indicating that LLMs are helpful in this regard – in defending against software vulnerabilities, phishing attacks, network threats, moderating toxic content on social networks, etc. A recent MIT study has shown that inexperienced workers stand to gain the most from generative AI solutions, such as LLMs, while skilled workers gain incremental benefits [2, Section 5]. In other words, generative AI solutions are democratizing security expertise for everyone and are being termed as the "great equalizer". Organizations without much security expertise are leveraging AI assistive solutions for improving their security postures. Similarly, experiments are being carried out to evaluate the effectiveness of LLMs in succeeding at security

practitioner exams (e.g., CISM), CTF (Capture The Flag) challenges with and without human-in-the-loop, etc. The findings suggest these evolving models can narrow the divide between attackers and defenders.

On the other hand, LLMs also introduce significant security and privacy challenges, potentially expanding the attack surface in organizations where LLMs or LLM-integrated applications are deployed. Factors such as the novelty, scale, efficiency, and effectiveness of potential attacks, coupled with the unprecedented growth of new LLM-powered applications, add to the concerns. However, cyber security stands out as a domain where the concept and practice of red teaming has long been established. Now, red teaming is also being performed on LLM models and applications, during the different phases of LLM training, fine-tuning and operation. This evolution encourages a new synergy between ML and security researchers, architects, and engineers. It is also worth noting that the LLM security domain is witnessing multifaceted activities spanning industry, academia and government bodies, including the development of AI safety frameworks, the formation of alliances, the drafting of regulations, and the definition of processes. This comprehensive approach holds promise for mitigating LLM security risks and pave way for responsible development in this exciting field.

REFERENCES

1. Z. Lin, J. Cui, X. Liao, and X. Wang, "Malla: Demystifying real-world large language model integrated malicious services," in *USENIX Security*, 2024.

2. Google, "Secure, Empower, Advance: How AI Can Reverse the Defender's Dilemma." https://services.google.com/fh/files/misc/how-ai-can-reverse-defenders-dilemma.pdf, 2024.

3. X. He, S. Zannettou, Y. Shen, and Y. Zhang, "You Only Prompt Once: On the Capabilities of Prompt Learning on Large Language Models to Tackle Toxic Content," in *Proc. IEEE Symposium on Security and Privacy (SP)*, 2024.

4. D. Kumar, Y. AbuHashem, and Z. Durumeric, "Watch Your Language: Investigating Content Moderation with Large Language Models," in *Proceedings of the Eighteenth International AAAI Conference on Web and Social Media, ICWSM*, pp. 865–878, AAAI Press, 2024.

5. T. Koide, N. Fukushi, H. Nakano, and D. Chiba, "ChatSpamDetector: Leveraging Large Language Models for Effective Phishing Email Detection," *CoRR*, vol. abs/2402.18093, 2024.

6. Y. Lin, R. Liu, D. M. Divakaran, J. Y. Ng, Q. Z. Chan, Y. Lu, Y. Si, F. Zhang, and J. S. Dong, "Phishpedia: A Hybrid Deep Learning Based Approach to Visually Identify Phishing Webpages," in *30th USENIX Security Symposium*, 2021.

7. J. Lee, P. Lim, B. Hooi, and D. M. Divakaran, "Multimodal Large Language Models for Phishing Webpage Detection and Identification," in *Symposium on Electronic Crime Research (eCrime 2024)*, 2024.

8. T. Ali and P. Kostakos, "HuntGPT: Integrating Machine Learning-Based Anomaly Detection and Explainable AI with Large Language Models (LLMs)," *CoRR*, vol. abs/2309.16021, 2023.

9. W. Qiao, T. Dogra, O. Stretcu, Y.-H. Lyu, T. Fang, D. Kwon, C.-T. Lu, E. Luo, Y. Wang, C.-C. Chia, A. Fuxman, F. Wang, R. Krishna, and M. Tek, "Scaling Up LLM Reviews for Google Ads Content Moderation," in *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, Mar. 2024.

10. J. Puentes, A. Castillo, W. Osejo, Y. Calderón, V. Quintero, L. Saldarriaga, D. Agudelo, and P. Arbeláez, "Guarding the guardians: Automated analysis of online child sexual abuse," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3728–3732, 2023.

11. N. Lee, T. Wattanawong, S. Kim, K. Mangalam, S. Shen, G. Anumanchipali, M. W. Mahoney, K. Keutzer, and A. Gholami, "LLM2LLM: Boosting LLMs with Novel Iterative Data Enhancement," *arXiv preprint arXiv:2403.15042*, 2024.

12. A. Akkus, M. Li, J. Chu, M. Backes, Y. Zhang, and S. Sav, "Generated data with fake privacy: Hidden dangers of fine-tuning large language models on generated data," *arXiv preprint arXiv:2409.11423*, 2024.

13. Z. Ba, J. Zhong, J. Lei, P. Cheng, Q. Wang, Z. Qin, Z. Wang, and K. Ren, "Surrogateprompt: Bypassing the safety filter of text-to-image models via substitution," 2023.

14. P. Guo, F. Liu, X. Lin, Q. Zhao, and Q. Zhang, "L-AutoDA: Leveraging Large Language Models for Automated Decision-based Adversarial Attacks," *arXiv preprint arXiv:2401.15335*, 2024.

15. H. Zhang, H. Chen, Z. Song, D. Boning, I. S. Dhillon, and C.-J. Hsieh, "The limitations of adversarial training and the blind-spot attack," *arXiv preprint arXiv:1901.04684*, 2019.

Dinil Mon Divakaran is a senior principal scientist at the Institute for Infocomm Research (I^2R), A*STAR, Singapore, 138632, Singapore, and an adjunct assistant professor at the School of Computing, National University of Singapore, 117417, Republic of Singapore. His research interests include AI for security, network security and privacy, phishing attacks, and endpoint protection. Divakaran received a Ph.D. from ENS Lyon, France, in the joint lab of INRIA and Bell Labs. He is a Senior Member of IEEE. Contact him at dinil_divakaran@i2r.a-star.edu.sg.

Sai Teja Peddinti is a staff research scientist at Google, Sunnyvale, CA 94089 USA. His research interests include applying ML/AI and data analysis techniques to build novel privacy and security solutions. Peddinti received a Ph.D. in computer science from the School of Engineering, New York University. He is a Member of IEEE. Contact him at psaiteja@google.com.