VAP-6: A Benchmarking Framework on Vulnerability Assessment and Penetration Testing for Language Models

Bishal Ranjan Das
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
bishalranjandas80@gmail.com

Sonia Jassi
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
sonia.30956@lpu.co.in

Vaibhav Khandelwal
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
khandelwalvaibhav123456@gmail.com

Tarun
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
badgujjar9991@gmail.com

Akansh Agarwal
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
akanshagarwal.alwar@gmail.com

Krittika Priyadarshini
Dept. of Computer Science
Lovely Professional University
Phagwara, Punjab, India
krittika.12111591@lpu.in

2025 International Conference on Electronics, AI and Computing (EAIC) | 979-8-3315-5358-6/25/$31.00 ©2025 IEEE | DOI: 10.1109/EAIC66483.2025.11101426

***Abstract*—The integration of Large Language Models (LLMs) into cybersecurity operations, particularly Vulnerability Assessment and Penetration Testing (VAPT), has shown significant promise. However, there remains a scarcity of comprehensive benchmarks for evaluating LLMs in the VAPT domain, especially for small, open-source models suitable for local deployment. This paper introduces VAP-6, a novel benchmark comprising six distinct datasets designed to evaluate LLM capabilities across crucial VAPT knowledge domains: Common Vulnerabilities and Exposures (CVE) and Common Weakness Enumeration (CWE) identification, Common Vulnerability Scoring System (CVSS) prediction, scenario-based reasoning aligned with Certified Ethical Hacker (CEH) v12 and CompTIA PenTest+ PT0-002 certification exams, VAPT tools proficiency, and CVE-to-Metasploit module mapping. We introduce the VAP-6 methodology, encompassing dataset creation from authoritative sources like CVE and CWE MITRE, Exploit DB and Github with refinement through ChatGPT and manual verification. The benchmark was applied to evaluate selected open-source LLMs with parameters ranging from 2 to 3 billion (Qwen 2.5, Gemma2, Llama 3.2), employing Q4 quantization to ensure local computational efficiency via Ollama. This research establishes a standardized framework for benchmarking and comparing such LLMs, facilitating the development of more robust, private, and computationally efficient AI tools for VAPT professionals.**

***Index Terms*—VAPT, LLM, Benchmark, Cybersecurity, Vulnerability Assessment, Penetration Testing, Small Language Models, Local LLM, Open-Source LLM**

I. Introduction

Vulnerability Assessment and Penetration Testing (VAPT) represents a critical cybersecurity practice, with its effectiveness and lifecycle being significant research areas [1], for identifying and addressing security vulnerabilities within IT infrastructures. Recent advances in Large Language Models (LLMs) demonstrate remarkable potential to transform various cybersecurity domains, as documented in contemporary systematic reviews [2], [3]. However, effectively deploying these models in specialized fields like VAPT requires thorough evaluation methodologies. While numerous general LLM benchmarks exist, including those for language understanding assessment [4], [5], comprehensive multitask capability evaluation [6], and prompt-based testing frameworks [7], with several emerging specifically for cybersecurity applications [8], [9], [10], [11], these typically do not adequately address the specific requirements of VAPT or the particular considerations for smaller, locally deployable models. Our research aims to extend existing benchmarking efforts by developing a framework focused on this specialized domain. The ability for offline model operation is essential for VAPT practitioners who must avoid exposing sensitive system data to third-party LLMs hosted online. Furthermore, less computationally demanding models democratize access and enable greater customization possibilities through fine-tuning. To address this gap, we present VAP-6 (Vulnerability Assessment and Pentesting - 6), a benchmarking framework designed to evaluate the knowledge base and reasoning capabilities of LLMs within the VAPT domain. VAP-6 encompasses 7800 questions distributed across six specialized datasets comprising of CVEMCQs, CWEMCQs, CVSS Prediction, CEHv12 [12] and PenTest+ PT0-002 [13] Styled MCQs, VAPT Tools MCQs and CVE ID to Metasploit Module Mapping MCQs.

This paper elaborates on the design methodology and implementation of VAP-6. We assessed three accessible, open-source LLMs: Qwen 2.5 (3B parameters) [14], Gemma2 (2B parameters) [15], and Llama 3.2 (3B parameters) [16]. These models were selected based on their parameter size (2-3 billion), compatibility with local deployment using Q4 quantization through Ollama [17], and their open-source nature, which facilitates potential fine-tuning, which is a huge advantage often

unavailable with proprietary alternatives like GPT, Gemini, etc. This study contributes a standardized methodology for evaluating LLM effectiveness in VAPT applications, establishing a foundation for developing more dependable and accessible AI-enhanced security tools.

II. Related Work

The evaluation of LLMs constitutes a significant research focus. Comprehensive benchmarks such as GLUE [4] and SuperGLUE [5] evaluate fundamental natural language capabilities, while MMLU [6] assesses extensive multitask proficiency. PromptBench [7] provides an integrated framework for LLM assessment, emphasizing prompt engineering techniques.

Within cybersecurity, various benchmarks facilitate understanding LLM capabilities. CTIBench [8] presents a framework for evaluating LLMs in Cyber Threat Intelligence applications. SECURE [9] and CyberSecEval 3 [10] attempt to benchmark LLMs across diverse cybersecurity threats and functions. CyberMetric [11] employs retrieval-augmented generation to assess LLM cybersecurity expertise. VAP-6 extends these initial efforts by concentrating specifically on the VAPT domain and examining the performance of compact, quantized, locally executable models which is an area requiring dedicated attention.

Research on LLM applications in VAPT is advancing rapidly. Systems like PentestGPT [18] and PentestAgent [19] demonstrate how LLMs can facilitate penetration testing automation. Isozaki et al. expanded automated penetration testing further by incorporating benchmarks and analysis for LLM enhancements [20]. Muzsai et al.'s HackSynth combined an LLM agent with an evaluation system for autonomous penetration testing [21]. End-to-end automated web penetration testing has been explored through multi-agent architectures like BreachSeek [22] and frameworks such as AutoPT by Wu et al. [23]. Goyal et al. illustrated how LLMs can enhance manual penetration testing practices [24].

LLMs have also been investigated for specific VAPT functions including vulnerability identification [25], [26], reconnaissance [27], and privilege escalation [28]. LLMs have supported offensive operations such as directory brute-forcing [29]. The potential for LLMs in autonomous exploitation of one-day vulnerabilities [30] and zero-day vulnerabilities [31] has been demonstrated. Begum addresses the growing implementation of AI and ML in penetration testing methodologies [32]. Fang et al. also addresses LLMs applications in web application attacks [33]. Methodological innovations in VAPT, including new algorithms focusing on OWASP Top 10, continue to emerge [34]. Additionally, LLMs are being applied to predictive challenges, such as anticipating cyber attacks in IoT environments [35]. VAP-6 complements these diverse research directions by providing a consistent methodology for evaluating the inherent VAPT knowledge of LLMs, with particular emphasis on locally deployable implementations.

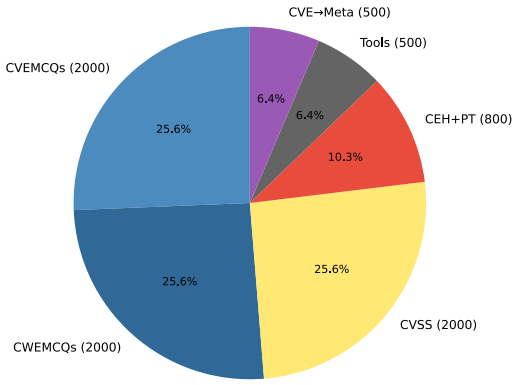


Fig. 1: Distribution of questions in each dataset

III. VAP-6 Benchmark Design & Methodology

The VAP-6 benchmark was developed to be comprehensive, evaluating LLMs across a broad spectrum of VAPT knowledge domains. The complete dataset generation and testing process, from initial data collection through LLM performance assessment, is illustrated in Fig. 2.

A. VAP-6 Datasets

The VAP-6 benchmark comprises six distinct evaluation datasets shown in Fig. 1:

1) **CVEMCQs (2000 questions):** This dataset evaluates knowledge of Common Vulnerabilities and Exposures (CVEs) [36]. Given a specific CVE ID, the LLM must identify the correct description from four options.
2) **CWEMCQs (2000 questions):** This dataset assesses understanding of Common Weakness Enumeration (CWEs) [37]. Similar to CVEMCQs, it requires matching a CWE ID with its proper description.
3) **CVSS Prediction (2000 questions):** This dataset tests the ability to analyze a vulnerability description and determine its appropriate CVSS v3.1 severity classification, vector string composition, and base score.
4) **CEH v12 [12] & CompTIA PenTest+ PT0-002 [13] Styled MCQs (800 questions):** This dataset incorporates scenario-based multiple-choice questions modeled after these industry certifications, evaluating practical reasoning capabilities. (3 CEH-formatted tests containing 125 questions each, and 5 PenTest+-formatted tests containing 85 questions each were created).
5) **VAPT Tools MCQs (500 questions):** This collection focuses on assessing proficiency with popular VAPT tools (Nmap, Burp Suite, Metasploit, sqlmap, Wireshark, and Nessus).
6) **CVE ID to Metasploit Module Mapping MCQs (500 questions):** This dataset verifies the ability to correctly associate given CVEs with relevant Metasploit modules and vice-versa.

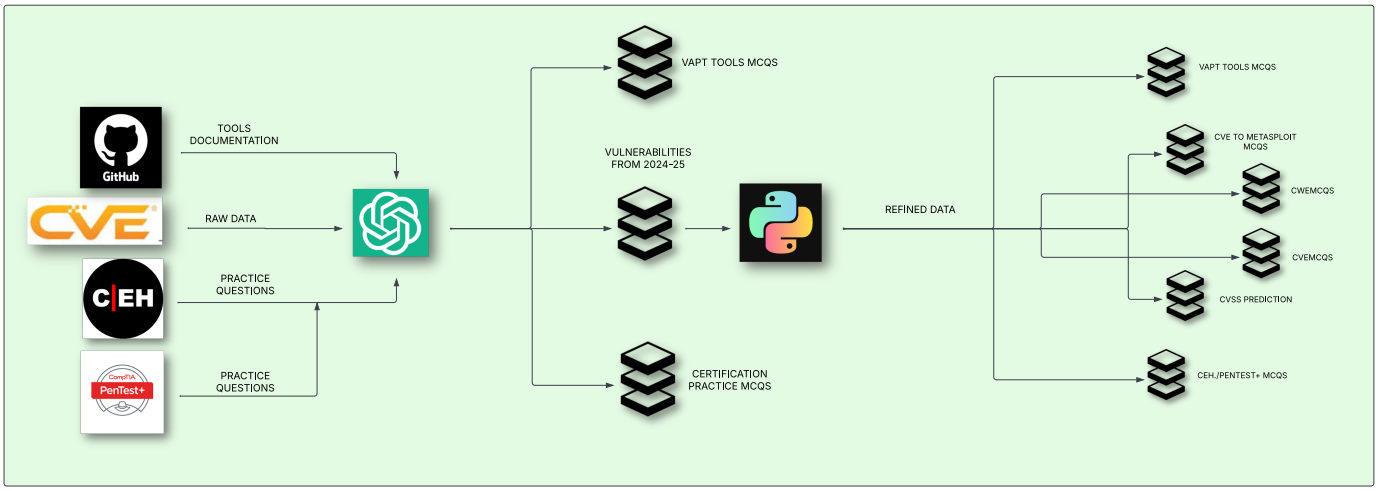


Fig. 2: VAP-6 dataset creation and refinement workflow.

B. Dataset Construction

Source material for CVEMCQs, CWEMCQs, and CVSS prediction datasets was gathered from authoritative public repositories including CVE MITRE for CVEs [36], the official CWE website [37], and Exploit DB [38]. This raw data, along with content for other datasets, underwent processing and reformatting into appropriate MCQ or prediction formats using ChatGPT (GPT-4o/GPT-4o mini) [39] to ensure consistency in question structure and clarity of options. A critical subsequent phase involved comprehensive manual validation and verification of each question and response developed to confirm accuracy and relevance within the VAPT domain context.

C. List of Picked LLMs and Experimental Setup

The selected LLMs, their parameter configurations, quantization methods, and sources are presented in Table I. These models, with parameter counts ranging from 2 to 3 billion, were specifically chosen for their open-source availability and capacity to function locally on moderate computing hardware (16GB RAM, 4GB Nvidia vRAM). All models were implemented with Q4 quantization (specifically Q4_K_M where necessary, representing an optimal quantization level) through the Ollama framework [17]. This configuration prioritizes practical utility for users valuing data confidentiality, requiring offline accessibility, or possessing limited computational resources, and who might wish to fine-tune models for specialized VAPT applications. A base prompt was created and modified specifically for all the models. *"You are a cybersecurity and VAPT expert. Follow the given instructions and answer the questions accordingly. No explanations required."*

TABLE I: Profile of Evaluated LLMs.

Model Name	Base Parameters	Quantization	Developer/Source
Qwen 2.5	3B	Q4_K_M	Alibaba Cloud [14]
Gemma2	2B	Q4_K_M	Google [15]
Llama 3.2	3B	Q4_K_M	Meta AI [16]

D. Evaluation Metrics

Performance on each VAP-6 dataset was assessed using specific metrics. For MCQ-based datasets, accuracy serves as the primary metric (Eq. 1). For the CVSS Prediction dataset, evaluation incorporates accuracy of severity classifications, CVSS vector accuracy measured through exact matches (Eq. 1) complemented by Levenshtein distance calculations for near matches, and Root Mean Squared Error (RMSE) for base score predictions (Eq. 2).

$$\text{Accuracy } (\%) = \frac{C_{\text{Ans}}}{T_{\text{Ans}}} \times 100 \tag{1}$$

where C_{Ans} represents the correct answers of the models and T_{Ans} represents the total number of questions/answers.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - A_i)^2} \tag{2}$$

Where n represents the collection of vulnerability descriptions, P_i denotes the LLM's predicted CVSS base score for item i, and A_i indicates the actual (ground truth) CVSS base score for item i.

IV. Results and Analysis

This section presents the performance outcomes of the LLMs evaluated on the VAP-6 benchmark datasets. Accuracy metrics for each model across the five MCQ-based sets are detailed in Table II. Performance results for the CVSS prediction task are provided in Table III.

A comparative overview of LLM accuracy across the VAP-6 MCQ sets is visualized in Fig. 3. This grouped bar chart illustrates the percentage of correct responses for each model across the six MCQ sub-benchmarks. For the CEH/PenTest+ style MCQs (combined as a single dataset), 70 percent and 85 percent are typical certification passing score thresholds for human candidates.

Fig. 4 displays the error correlation (from a total of 5800 questions) across all models in a heatmap for the combined

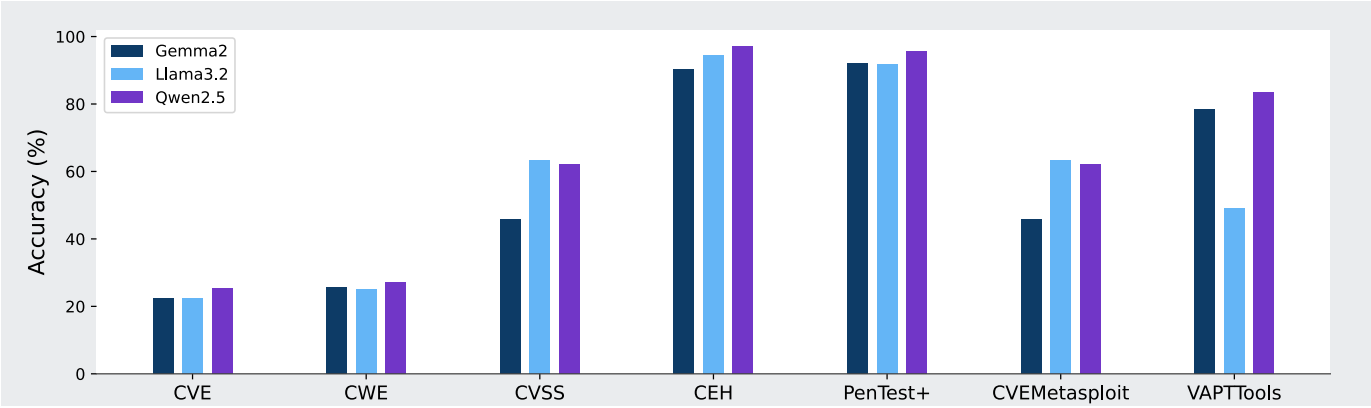


Fig. 3: Comparative accuracy of three language models on the VAP-6 MCQ dataset categories.

TABLE II: LLM Accuracy (%) on VAP-6 MCQ Datasets.

Dataset	Qwen 2.5	Gemma2	Llama 3.2
CVE MCQs	25.45	22.25	22.50
CWE MCQs	27.20	25.65	25.10
CEH v12 MCQs	97.60	88.80	96.00
PenTest+ MCQs	96.47	92.94	96.47
VAPT Tools MCQs	83.57	78.40	49.10
CVE-Metasploit	62.00	45.80	63.20

TABLE III: LLM Performance on VAP-6 CVSS Prediction.

CVSS Metric	Qwen 2.5	Gemma2	Llama 3.2
Severity (Acc %)	2.40	51.25	68.30
Vector Exact Match (Acc %)	1.75	0.35	0.70
Vector Avg. Levenshtein	5.73	5.98	8.81
Base Score (RMSE)	3.4675	1.2327	1.456

VAP-6 MCQ datasets, offering a comprehensive view of overall MCQ performance patterns.

Fig. 5 provides a detailed visual breakdown of performance on the CVSS prediction task. Panel (a) shows the misclassifications in severity across all the three models. Panel (b) shows CVSS vector mismatches across all the categories for all the models. Panel (c) depicts the CVSS score predictions for each model

V. Discussion

The results documented in Tables II and III, alongside the visualizations in Figures 3, 4, and 5, provide valuable insights into the VAPT domain capabilities of the evaluated compact LLMs. The CVSS prediction results reveal unexpected patterns, with particularly notable findings in severity classification accuracy. Llama 3.2 substantially outperforms its counterparts with a 68.30% accuracy rate, followed by Gemma2 at 51.25%, while Qwen 2.5 demonstrates a surprisingly low 2.40% accuracy despite its strong performance in other categories. This disparity suggests that Llama 3.2 may have encountered more CVSS-related content during pre-training or possesses architectural advantages for this specific task. Vector string generation proved challenging for all models, with extremely low exact match rates (Qwen 2.5: 1.75%, Gemma2: 0.35%, Llama 3.2: 0.70%). The Levenshtein distance measurements (where lower values indicate closer matches) confirm that all models struggled to produce accurate CVSS vector strings, with Qwen 2.5 showing the least deviation (5.83), followed by Gemma2 (6.00), and Llama 3.2 exhibiting the greatest average distance (8.81). For base score prediction, Gemma2 achieved the lowest RMSE (1.2327), followed by Llama 3.2 (1.456), with Qwen 2.5 demonstrating a substantially higher error rate (3.4675). This indicates that despite Qwen 2.5's strong performance across most MCQ datasets, it faced particular challenges with numeric scoring aspects of CVSS assessment.

The models demonstrate a clear dichotomy between knowledge retrieval and reasoning tasks. While all three LLMs struggle with factual knowledge retrieval (evident in CVE and CWE identification tasks), they excel in scenario-based reasoning (reflected in CEH and PenTest+ MCQs). This suggests that these quantized models retain strong reasoning frameworks but have limited capacity for storing extensive vulnerability databases in their compressed form. A particularly noteworthy finding is that despite parameter counts of only 2-3 billion and aggressive Q4 quantization, all models achieved

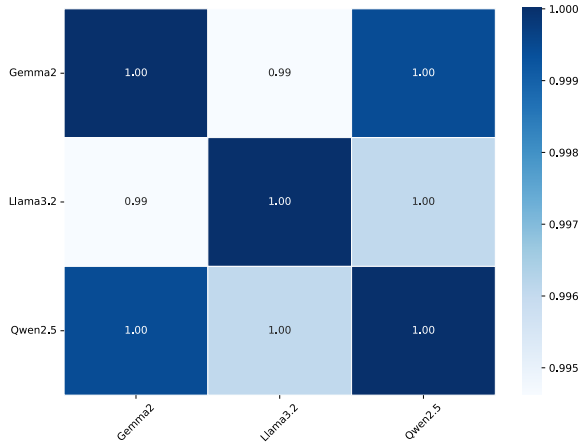
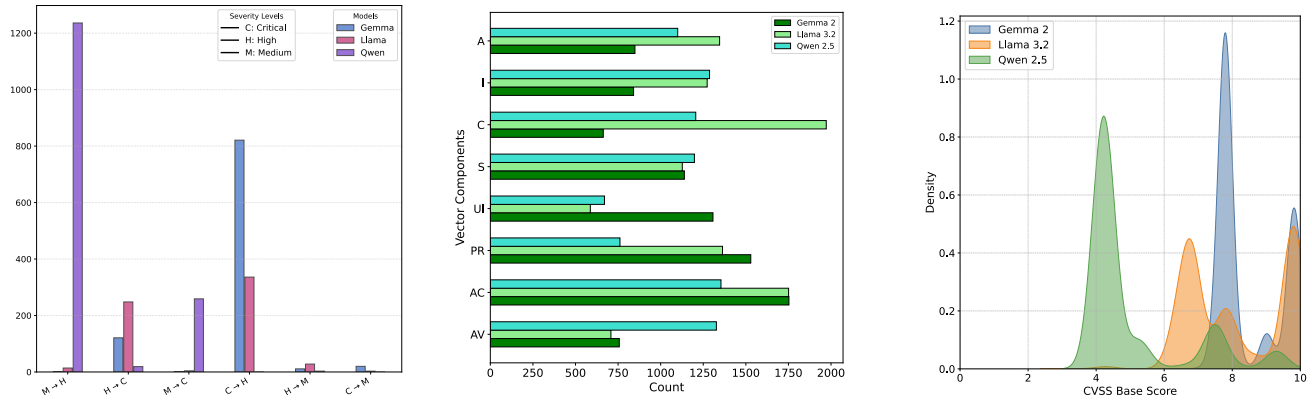


Fig. 4: Error Correlation between each model for the MCQ datasets.



(a) Severity Misclassifications (b) CVSS Vector Predictions (c) Distribution of CVSS Score Predictions

Fig. 5: LLM performance on CVSS prediction

performance levels well above typical certification passing thresholds (70-85%) on the CEH and PenTest+ style questions. Qwen 2.5 and Llama 3.2 exceeded 96% accuracy on these tasks, demonstrating that even small, quantized models can possess sophisticated reasoning capabilities for VAPT scenarios. The CVSS prediction results reveal additional nuances in model capabilities. The substantial performance gap in severity classification (Llama 3.2: 68.30% vs. Qwen 2.5: 2.40%) suggests that Llama 3.2's architecture or training approach may confer advantages for categorical classification tasks within the cybersecurity domain. Conversely, the uniformly low performance on vector string generation (all models below 2% exact match accuracy) highlights a significant limitation in generating structured, standardized output formats—a critical skill for real-world VAPT automation.

A significant contribution of this research is evaluating the practical utility of these Q4-quantized, locally executable LLMs (with 2–3 billion parameters) as assistive tools for VAPT professionals, especially considering their data privacy advantages and reduced computational requirements. The performance on CEH/Pentest+ styled questions, when contextualized against human certification passing thresholds (Fig. 3), provides a tangible reference point for assessing their knowledge level relative to established professional competency standards.

A. Limitations

A fundamental limitation of VAP-6 is its focus on knowledge assessment rather than evaluating LLMs in dynamic, interactive penetration testing scenarios requiring planning, execution, and environmental adaptation capabilities. While the datasets underwent refinement through ChatGPT processing and manual verification, they may retain inherent biases from source materials or the refinement methodology. Additionally, this evaluation concentrates on a specific selection of small, quantized LLMs and may not generalize to larger model architectures. Future research could explore expanding VAP-6 to incorporate more sophisticated reasoning assessments or simulated interactive testing environments, and evaluate a more diverse range of model implementations.

VI. Conclusion

This research introduces VAP-6, a comprehensive benchmark containing 7800 questions across six specialized datasets, designed to evaluate the capabilities of small (2–3 billion parameters), Q4-quantized, locally deployable LLMs in the Vulnerability Assessment and Penetration Testing domain. We detail its development methodology, dataset creation process utilizing authoritative sources such as CVE, CWE, and Exploit-DB with ChatGPT-enhanced processing and rigorous manual validation, and a systematic evaluation framework for models deployed for running locally.

VAP-6 contributes to the evolving landscape of cybersecurity benchmarks by providing a dedicated methodology for assessing LLMs on specific VAPT knowledge requirements, enabling the development and implementation of privacy-preserving and computationally efficient AI solutions for cybersecurity professionals. The findings generated through this benchmark offer valuable insights into the capabilities and limitations of selected small-scale LLMs, informing their practical application as supportive tools in VAPT operations and guiding future research into specialized LLM development for cybersecurity applications.

References

[1] K. S. Veenababu, "Life Cycle Assessment of Vulnerability and Penetration Testing on Systems and Proactive Action Taken to Resolve Possible Attacks on Networks," *International Journal of Management, Technology And Engineering*, vol. 13, no. 01, pp. 122–132, Oct 2023, [Online]. Available at: https://ssrn.com/abstract=4622392.

[2] H. Xu, S. Wang, N. Li, K. Wang, Y. Zhao, K. Chen, T. Yu, Y. Liu, and H. Wang, "Large Language Models for Cyber Security: A Systematic Literature Review," *arXiv preprint arXiv:2405.04760*, May 2024, [Online]. Available at: https://arxiv.org/abs/2405.04760.

[3] F. N. Motlagh, M. Hajizadeh, M. Majd, P. Najafi, F. Cheng, and C. Meinel, "Large Language Models in Cybersecurity: State-of-the-Art," *arXiv preprint arXiv:2402.00891*, Feb 2024, [Online]. Available at: https://arxiv.org/abs/2402.00891.

[4] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding," *arXiv preprint arXiv:1804.07461*, Apr 2018, [Online]. Appeared in Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Available at: https://arxiv.org/abs/1804.07461.

[5] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., 2019, pp. 3261–3275, [Online]. Available at: https://proceedings.neurips.cc/paper/2019/hash/4496bf24afe7fab6f046bf4923da8de6-Abstract.html.

[6] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, "Measuring Massive Multitask Language Understanding," *arXiv preprint arXiv:2009.03300*, Sep 2020, [Online]. Appeared in Proceedings of the International Conference on Learning Representations (ICLR) 2021. Available at: https://arxiv.org/abs/2009.03300.

[7] K. Zhu, Q. Zhao, H. Chen, J. Wang, and X. Xie, "PromptBench: A Unified Library for Evaluation of Large Language Models," *Journal of Machine Learning Research*, vol. 25, no. 254, pp. 1–22, 2024, [Online]. Available at: https://www.jmlr.org/papers/v25/24-0023.html.

[8] M. T. Alam, D. Bhusal, L. Nguyen, and N. Rastogi, "CTIBench: A Benchmark for Evaluating LLMs in Cyber Threat Intelligence," *arXiv preprint arXiv:2406.07599*, Jun 2024, [Online]. Available at: https://arxiv.org/abs/2406.07599.

[9] D. Bhusal, M. T. Alam, L. Nguyen, A. Mahara, Z. Lightcap, R. Frazier, R. Fieblinger, G. L. Torales, B. A. Blakely, and N. Rastogi, "SECURE: Benchmarking Large Language Models for Cybersecurity," *arXiv preprint arXiv:2405.20441*, May 2024, [Online]. Available at: https://arxiv.org/abs/2405.20441.

[10] S. Wan, C. Nikolaidis, D. Song, D. Molnar, J. Crnkovich, J. Grace, M. Bhatt, V. Bala, M. Bargury, M. Clifford, and A. Stubblefield, "CyberSecEval 3: Advancing the Evaluation of Cybersecurity Risks and Capabilities in Large Language Models," *arXiv preprint arXiv:2408.01605*, Aug 2024, [Online]. Available at: https://arxiv.org/abs/2408.01605.

[11] N. Tihanyi, M. A. Ferrag, R. Jain, T. Bisztray, and M. Debbah, "CyberMetric: A Benchmark Dataset Based on Retrieval-Augmented Generation for Evaluating LLMs in Cybersecurity Knowledge," in *2024 IEEE International Conference on Cyber Security and Resilience (CSR)*. IEEE, Sep 2024, pp. 296–302, [Online]. Available at: https://doi.org/10.1109/CSR61664.2024.10679494.

[12] EC-Council, "Certified Ethical Hacker (CEH) v12," 2025, [Online]. Available at: https://www.eccouncil.org/train-certify/certified-ethical-hacker-ceh/.

[13] CompTIA, "CompTIA PenTest+ Certification (Exam PT0-002)," 2025, [Online]. Available at: https://www.comptia.org/certifications/pentest . Exam code PT0-002.

[14] Qwen Team and Alibaba Cloud, "Qwen2.5 Model on Ollama," Oct 2024, model available on Ollama. Original model by Alibaba Cloud. [Online]. Available at: https://ollama.com/library/qwen2.5.

[15] Gemma Team and Google DeepMind, "Gemma 2 Model on Ollama," Jun 2024, model available on Ollama. Original model by Google DeepMind. [Online]. Available at: https://ollama.com/library/gemma2.

[16] Meta AI, "Llama 3.2 Model on Ollama," Apr 2024, model available on Ollama. Original model by Meta AI. [Online]. Available at: https://ollama.com/library/llama3.

[17] Ollama Team, "Ollama: Run Large Language Models Locally," 2024, [Online]. Available at: https://ollama.com/.

[18] G. Deng, Y. Liu, V. Mayoral-Vilches, P. Liu, Y. Li, Y. Xu, T. Zhang, Y. Liu, M. Pinzger, and S. Rass, "PentestGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing," in *33rd USENIX Security Symposium (USENIX Security 24)*. USENIX Association, 2024, pp. 847–864, [Online]. Available at: https://www.usenix.org/conference/usenixsecurity24/presentation/deng.

[19] X. Shen, L. Wang, Z. Li, Y. Chen, W. Zhao, D. Sun, J. Wang, and W. Ruan, "PentestAgent: Incorporating LLM Agents to Automated Penetration Testing," *arXiv preprint arXiv:2411.05185*, Nov 2024, [Online]. Available at: https://arxiv.org/abs/2411.05185.

[20] I. Isozaki, M. Shrestha, R. Console, and E. Kim, "Towards Automated Penetration Testing: Introducing LLM Benchmark, Analysis, and Improvements," *arXiv preprint arXiv:2410.17141*, Oct 2024, [Online]. Available at: https://arxiv.org/abs/2410.17141.

[21] L. Muzsai, D. Imolai, and A. Lukács, "HackSynth: LLM Agent and Evaluation Framework for Autonomous Penetration Testing," *arXiv preprint arXiv:2412.01778*, Dec 2024, [Online]. Available at: https://arxiv.org/abs/2412.01778.

[22] I. Alshehri, A. Alshehri, A. Almalki, M. Bamardouf, and A. Akbar, "BreachSeek: A Multi-Agent Automated Penetration Tester," *arXiv preprint arXiv:2409.03789*, Sep 2024, [Online]. Available at: https://arxiv.org/abs/2409.03789.

[23] B. Wu, G. Chen, K. Chen, X. Shang, J. Han, Y. He, W. Zhang, and N. Yu, "AutoPT: How Far Are We from the End2End Automated Web Penetration Testing?" *arXiv preprint arXiv:2411.01236*, Nov 2024, [Online]. Available at: https://arxiv.org/abs/2411.01236.

[24] D. Goyal, S. Subramanian, and A. Peela, "Hacking, the Lazy Way: LLM Augmented Pentesting," *arXiv preprint arXiv:2409.09493*, Sep 2024, [Online]. Available at: https://arxiv.org/abs/2409.09493.

[25] G. Eberhardt and Á. Milákovich, "VulnGPT: Enhancing Source Code Vulnerability Detection Using AutoGPT and Adaptive Supervision Strategies," in *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE, Apr 2024, pp. 450–454, [Online]. Available at: https://doi.org/10.1109/DCOSS-IoT61029.2024.00072.

[26] A. Cheshkov, P. Zadorozhny, and R. Levichev, "Evaluation of ChatGPT Model for Vulnerability Detection," *arXiv preprint arXiv:2304.07232*, Apr 2023, [Online]. Available at: https://arxiv.org/abs/2304.07232.

[27] S. Temara, "Maximizing Penetration Testing Success with Effective Reconnaissance Techniques Using ChatGPT," *arXiv preprint arXiv:2307.06391*, Jul 2023, [Online]. Available at: https://arxiv.org/abs/2307.06391.

[28] A. Happe, A. Kaplan, and J. Cito, "LLMs as Hackers: Autonomous Linux Privilege Escalation Attacks," *arXiv preprint arXiv:2310.11409*, Oct 2023, [Online]. Available at: https://arxiv.org/abs/2310.11409.

[29] A. Castagnaro, M. Conti, and L. Pajola, "Offensive AI: Enhancing Directory Brute-forcing Attack with the Use of Language Models," in *Proceedings of the 2024 Workshop on Artificial Intelligence and Security (AISec '24)*. ACM, Nov 2024, pp. 184–195, [Online]. Available at: https://doi.org/10.1145/3689932.3694770.

[30] R. Fang, R. Bindu, A. Gupta, and D. Kang, "LLM Agents Can Autonomously Exploit One-Day Vulnerabilities," *arXiv preprint arXiv:2404.08144*, Apr 2024, [Online]. Available at: https://arxiv.org/abs/2404.08144.

[31] Y. Zhu, A. Kellermann, A. Gupta, P. Li, R. Fang, R. Bindu, and D. Kang, "Teams of LLM Agents Can Exploit Zero-Day Vulnerabilities," *arXiv preprint arXiv:2406.01637*, Jun 2024, [Online]. Available at: https://arxiv.org/abs/2406.01637.

[32] S. R. Begum, G. Nalinipriya, and C. P. Reddy, "Integrating Machine Learning and AI in Penetration Testing: Enhancing Threat Detection and Vulnerability Assessment," *International Journal of Advanced Engineering Technologies and Innovations*, vol. 1, no. 1, pp. 762–782, Aug 2024, [Online]. Available at: https://ijaeti.com/index.php/Journal/article/view/665.

[33] R. Fang, R. Bindu, A. Gupta, Q. Zhan, and D. Kang, "LLM Agents Can Autonomously Hack Websites," *arXiv preprint arXiv:2402.06664*, Feb 2024, [Online]. Available at: https://arxiv.org/abs/2402.06664.

[34] R. Ventura, D. J. Franco, and O. K. Akram, "A Novel VAPT Algorithm: Enhancing Web Application Security Trough OWASP Top 10 Optimization," *arXiv preprint arXiv:2311.10450*, Nov 2023, [Online]. Available at: https://arxiv.org/abs/2311.10450.

[35] A. Diaf, A. A. Korba, N. E. Karabadji, and Y. Ghamri-Doudane, "Beyond Detection: Leveraging Large Language Models for Cyber Attack Prediction in IoT Networks," in *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE, Apr 2024, pp. 117–123, [Online]. Available at: https://doi.org/10.1109/DCOSS-IoT61029.2024.00026.

[36] The CVE Program, "Common Vulnerabilities and Exposures (CVE)," 2025, [Online]. Available at: https://www.cve.org/.

[37] The MITRE Corporation, "Common Weakness Enumeration (CWE)," 2025, [Online]. Available at: https://cwe.mitre.org/.

[38] Offensive Security, "Exploit DB," 2025, [Online]. Available at: https://www.exploit-db.com/.

[39] OpenAI, "ChatGPT," 2025, [Online]. Available at: https://openai.com/chatgpt/.