# Detecting Security Vulnerabilities in Network Configurations Using LLMs

## Introduction

The rise of digital connectivity has made cybersecurity a critical concern for organizations and individuals. With the exponential growth in network complexity, identifying vulnerabilities in network configurations has become a complicated challenge. Traditional tools and methodologies, while effective to a degree, are limited in their ability to adapt to evolving threats. Recent advancements in Artificial Intelligence (AI), particularly Large Language Models (LLMs), have introduced transformative capabilities. These models, trained on vast datasets, demonstrate exceptional proficiency in understanding, analysing, and mitigating complex cybersecurity issues, including vulnerabilities in network configurations. By leveraging their natural language processing capabilities, LLMs offer an innovative approach to interpreting security policies, identifying weaknesses, and suggesting robust defences [1], [2].

LLMs have shown significant promise in vulnerability detection by analysing huge amounts of data, such as system logs, configuration files, and network traffic patterns. For instance, models like GPT-4 and its counterparts are adept at recognizing intricate patterns that may signal potential security risks. Furthermore, the adaptability of these models through fine-tuning allows them to specialize in detecting specific vulnerabilities in diverse network environments. In addition to detection, LLMs play a critical role in generating actionable recommendations to enhance network security. However, their deployment comes with challenges, including the risks associated with misconfigurations or unintended consequences resulting from automated suggestions [1], [3].

The ability of LLMs to process unstructured and structured data alike provides them with a unique edge in network security. By integrating advanced prompt engineering techniques, researchers and practitioners have refined the ways in which LLMs interact with datasets. Such advancements ensure the accuracy and relevance of the insights generated by these models. Despite their potential, however, reliance on LLMs for security assessments raises important questions about their compliance with established standards and benchmarks. This has driven a surge of interest in exploring their capabilities, limitations, and the ethical considerations of deploying such systems in critical network infrastructures [4], [5].

## LLMs in Cybersecurity

Artificial Intelligence (AI) is transforming the field of cybersecurity by introducing innovative tools and methodologies to counter increasingly complex cyber threats. It offers creative solutions that not only improve current security frameworks but also open the door for proactive measures against potential breaches. By analysing massive amounts of data at high speeds, AI systems can spot anomalies and potential threats that would be challenging for human analysts to detect in real time.

LLMs have demonstrated significant capabilities in cybersecurity by enhancing network security, automating vulnerability detection, and improving incident response. As mentioned by [5], LLMs can analyse vast amounts of security-related data, including network logs, system configurations, and security advisories, to identify potential threats and weaknesses. They support penetration testing by generating security-related insights and simulating potential attack vectors, enabling organizations to proactively strengthen their defences. Furthermore, as noted in [2], LLMs play a role in user authentication by analysing behavioural patterns to detect anomalies in login attempts, enhancing identity verification processes. Additionally, they contribute to malware detection by scanning files, emails, and web traffic for suspicious activity, helping security teams respond to threats in real-time.

LLMs also enhance incident response workflows by analysing security reports, log files, and threat intelligence to uncover vulnerabilities and detect attack patterns [2]. They can be used to generate automated security policies and assist in compliance monitoring, ensuring that organizations adhere to regulatory standards. By leveraging their advanced natural language processing (NLP) capabilities, LLMs improve real-time threat intelligence gathering, allowing cybersecurity professionals to stay ahead of evolving cyber threats.

## Application of LLMs in Cybersecurity

Cybersecurity analytics depend on processing vast amounts of data to identify **patterns and anomalies** that indicate potential threats [10]. LLMs analyse unstructured data from multiple sources like online threat intelligence, helping to detect emerging cyber threats and improve decision-making in security operations.

LLMs contribute to vulnerability management by prioritizing vulnerabilities based on their potential impact, freeing up security personnel to concentrate on the most important problem. AI-powered tools can be used to analyse network traffic data or system logs, detecting anomalies that traditional methods might miss.[2], [7].

Incident response benefits significantly from AI-driven automation, which correlates security events, prioritizes alerts, and optimizes response workflows to mitigate cyber threats effectively [2]. Additionally, LLMs streamline forensic analysis by reconstructing attack scenarios, reducing the time and effort required for cybersecurity investigations [7]. This also allowed the facilitation of automated routine cybersecurity tasks, such as log analysis and vulnerability assessments, freeing up human expert to focus on more complex issues [6].

LLMs improve malware detection and classification by analysing code execution patterns, network anomalies, and malicious behaviours [2]. In ransomware mitigation, they help identify attack trends enabling proactive defence measures against evolving threats [9].

AI-driven behavioural analytics enhance user authentication and access management by analysing biometric data, device activity, and login patterns to detect anomalous user behaviour [2]. Machine learning models create behavioural baselines, allowing organizations to identify compromised accounts and insider threats more efficiently [7].

AI also significantly enhances automation in cybersecurity, reducing the manual workload of security professionals. It streamlines repetitive tasks such as monitoring network traffic and analysing alerts, allowing experts to focus on strategic initiatives like threat hunting and incident response [2]. This capability is particularly beneficial for organizations with limited resources, as it enables them to prioritize and mitigate high-risk vulnerabilities more effectively.

## Transformer Architecture

The transformative capabilities of LLMs are rooted in its underlying transformer architecture, a deep learning framework designed for sequential data processing. Since the advent of attention, the transformer has become the gold standard for language models. By leveraging multi-head attention and feed-forward networks, transformers process sequences effectively, ensuring scalability and efficiency in handling vast datasets [12]. This architecture allows LLMs to understand the relationship between words and phrases in a given context, allowing it to generate coherent and contextually accurate responses.

A core feature of the transformer architecture is its attention mechanisms. This mechanism calculates similarity scores between queries (representing the current word), keys (representing all words in the sequence), and values (providing additional context). These scores are then used to weigh the importance of each word, enabling model to understand relationships and context within the input [12]. The Transformer architecture, originally designed with an encoder-decoder structure, enables models to capture dependencies across sequences, forming the basis for various language models. BERT which uses an encoder only structure, T5 encoder-decoder structures and GPT decoder only, They all leverage attention mechanisms to generate coherent and contextually relevant responses [14].

## Prompt Engineering

### Prompt Engineering: Concepts and Significance

Prompt Engineering has emerged as a critical technique in optimizing the performance of Large Language Models (LLMs), enabling them to generate accurate and contextually relevant outputs for a wide range of applications. At its core, it involves the careful formulation and refinement of questions or commands to generate targeted, beneficial responses from AI models. This methodical approach aligns the capabilities of generative AI with human goals and organizational requirements, producing responses that meet specific objectives.

In-Context Learning (ICL) is also an important part of prompt engineering. ICL refers to a process in which a language model is fine-tuned or updated with additional knowledge and information while it is actively deployed and interacting with users or a specific environment. This allows the model to adapt and improve its performance over time based on the context in which it is being used.

An important thing to note is that Prompts should be simple, specific and concise. These qualities ensure that the model provides accurate and relevant responses. Complex queries may introduce ambiguity leading to less precise outputs [5].

### Previous Work Using Prompt Engineering

Prompt engineering has been extensively used in the fields of cybersecurity and software vulnerability detection to optimize Large Language Models (LLMs) like ChatGPT. Research has shown that carefully structured prompts significantly enhance vulnerability detection accuracy and efficiency.

The authors in [15] investigated the impact of various prompt designs on vulnerability detection. The researchers demonstrated that structured prompts incorporating auxiliary information such as API call sequences and data flow graphs improve ChatGPT’s detection capabilities. Additionally, researchers investigating prompt-enhanced vulnerability detection found that incorporating control flow graphs (CFGs) and data flow graphs (DFGs) into prompts significantly enhances LLMs' ability to identify vulnerabilities. These structural elements help models better understand program semantics, leading to more accurate detection of security weaknesses

In another study, J. Cao et al. analysed the effect of various prompting strategies on automated program repair. Their study introduced an improved prompt template and found that providing clearer contextual details about code functionality, intended fixes, and dataset characteristics led to more accurate detection and repair of errors [16].

A key aspect of prompt engineering in cybersecurity applications is its ability to guide LLMs toward security-specific objectives by structuring tasks that align with real-world security requirements. Research has shown that executing LLMs over structured tasks, such as completing encryption modules or securing hardware components, allows them to better analyse security properties in code. By incorporating prompts with well-defined security intents, LLMs can be effectively steered toward identifying vulnerabilities and improving overall security assessment accuracy​ [20]

Another advancement in prompt engineering is prompt chaining, where sequential, interconnected prompts are used to guide LLMs through complex security tasks. Studies show that this technique improves penetration testing and threat analysis by making AI-generated responses more structured and actionable [5].

These findings highlight the necessity of precise, structured prompt designs in cybersecurity applications, ensuring that LLMs effectively contribute to threat detection, security compliance, and vulnerability management.

## Network Security Standards and Benchmarks

### Foundations of Network Security Standards

Network security standards provide robust frameworks to safeguard digital infrastructures against cyber threats, offering essential guidelines for secure network configurations. Among the most prominent standards are NIST SP 800-115, OWASP and CIS benchmarks.

The National Institute of Standards and Technology (NIST) provides guidance for secure information security procedures through its Special Publication 800-115. This document outlines a phased methodology (planning, execution, and post-execution) for conducting thorough information security assessments. It also emphasizes the importance of continuous improvement, ensuring organizations adapt to evolving security challenges while implementing NIST guidelines [3].

The Open Web Application Security Project (OWASP) has established itself as the industry application security standard since 2003. Its OWASP Top 10 serves as a launchpad for addressing the most critical risks, reflecting community data on evolving security threats [3].

Similar to NIST and OWASP, The Centre of Internet Security (CIS) benchmarks offer invaluable guidelines for securing diverse operating systems. These benchmarks encompass a wide array of security recommendations, from essential system configurations to advanced security measures, ensuring systems are resilient against a range of cyber threats [13]. The core principle underlying CIS benchmarks is adherence to best practices, informed by extensive research and expertise. These practices encompass a broad spectrum of security measures, from access control to network configurations, designed to create robust defence layers within IT environments [13].

### Challenges in Adhering to Standards

Adhering to network security standards poses challenges for organizations, including resource limitations and maintaining compliance in dynamic IT environments. While frameworks like NIST SP 800-115, OWASP and CIS benchmarks provide robust guidelines, consistent implementation can be complex and resource intensive.

Manual implementation remains a significant hurdle. Manual configuration is complex, error-prone, and time consuming, making it difficult to scale as IT environments grow. Additionally, systems that are individually secure may collectively represent vulnerabilities if not properly integrated. Financial and human constraints further hinder compliance with resource limitations often lead to incomplete or delayed adherence to standards.

The dynamic nature of cyber threats also necessitates frequent updates to security frameworks. The rapid evolution of cyber threats requires organizations to constantly update their security frameworks, which can be resource-intensive and time-consuming. For example, OWASP’s regularly updated Top 10 list requires organizations to adapt their practices to address emerging risks [13].

## Dataset Generation

Dataset generation is crucial for training machine learning models in cybersecurity, yet obtaining high-quality labelled data is difficult due to privacy concerns, proprietary restrictions, and evolving threats. Large Language Models (LLMs) offer an efficient solution by generating synthetic datasets that mimic real-world data, reducing dependency on manually curated datasets while ensuring scalability

In cybersecurity, dataset generation is crucial for developing machine learning models for vulnerability detection, malware classification, and network intrusion detection. This approach addresses data scarcity by producing synthetic attack scenarios, filling gaps left by limited real-world datasets. Researchers enhance dataset quality through task decomposition, refining data at different scales , and multi-step generation, structuring complex datasets into smaller chunks and generate a portion at a time [19].

LLMs sometimes hallucinate which can produce significant noise into generated results [19]. Overfitting is another risk when the model is limited to the training data, limiting its practical applicability [14]. Security risks also arise, as adversaries may exploit vulnerabilities in the models design or training data to manipulate its behaviour [2].

To enhance dataset reliability, Zero-shot and chain-of-thought prompting could be used to produce a final “approve/reject” recommendation [18]. Human intervention is also a straightforward strategy where comparing annotations from humans and LLMs guided by the same rules may be able to confirm the codes validity but can lead to considerable labelling costs and can be unrealistic in practical deployment [19].

LLMs contributes well to zero-day exploit detection, enabling models to recognize novel threats [14]. They also simulate cyber-attacks supporting intrusion detection and identifying vulnerabilities in IT infrastructures [2].

### Dataset Tailoring for Network Configurations

The reliability of Large Language Models (LLMs) in detecting security vulnerabilities within network configurations depends largely on the quality and relevance of the datasets used for training and evaluation. Network security datasets should accurately reflect real-world scenarios, capturing the intricacies of router configurations, topology constraints, and access control policies. This section explores methods for tailoring datasets that improve the ability of LLMs to synthesize, analyse, and validate network configurations while minimizing security risks.

LLM-generated network configurations often contain syntactic, semantic, and topological errors, which can lead to critical security vulnerabilities. Mondal et al. introduce the concept of Verified Prompt Programming (VPP), in which an LLM-generated configuration is assessed using automated verifiers such as Batfish and topology analysers to ensure correctness​ [8]. The VPP approach allows iterative refinement of datasets by identifying and correcting configuration errors, thereby reducing reliance on human intervention. This methodology ensures that datasets not only contain valid network configurations but also include common misconfigurations that allow LLMs to learn how to identify and remediate security flaws.

A major issue with existing network configuration datasets is that they often lack coverage of real-world vulnerabilities, making it difficult to assess how LLMs perform when detecting network-based attacks. Abdullin et al. highlight the potential of LLM-driven synthetic dialogue dataset generation, which can be applied to network security research by creating datasets that simulate real-time threat scenarios [17]. These datasets can introduce adversarial configurations, network misconfigurations, and access control flaws, which are crucial for training LLMs to recognize vulnerabilities in network configurations.

A key limitation in network security datasets is their bias toward specific network architectures, which can reduce an LLM’s ability to generalize across diverse environments. To overcome this, Netgen employs synthetic dataset generation by manipulating different network topologies, such as the NSFNet and Spanish 5-node networks, allowing the dataset to include a broader range of configurations​ [11]. Additionally, the Net2Plan tool is used to simulate various network planning scenarios, ensuring that datasets reflect realistic traffic patterns, device constraints, and failure conditions.

The process of dataset curation must balance realism with security considerations. According to research on network security and LLMs, datasets containing incorrect or insecure configurations can inadvertently train models to generate faulty or vulnerable outputs. To mitigate this, a combination of synthetic dataset augmentation and real-world dataset validation is necessary. Using ML-assisted dataset labelling, researchers have achieved an 85% feasibility rate in classifying optimal routing configurations, improving the accuracy of models trained for security assessments​ [11]

# Metrics

Evaluating the effectiveness of Large Language Models (LLMs) in detecting security vulnerabilities in network configurations requires well-defined performance metrics. Drawing from prior studies, key assessment areas include accuracy, efficiency, security robustness, and interpretability and usability to ensure LLM-based security solutions are reliable, scalable, and practical for real-world cybersecurity applications.

Accuracy and effectiveness play a critical role in measuring how well LLM-generated security assessments align with expected results. One essential metric used in prior evaluations is Perfect Predictions (PP), which classifies an LLM-generated security assessment as correct only if it exactly matches the expected output. This rigorous evaluation standard ensures that predictions are not just approximately correct but entirely accurate, minimizing the risk of false positives or misleading security assessments. This approach has been used in prior research to ensure that vulnerability detection models are held to high standards of precision and correctness.

In addition to accuracy, efficiency and performance determine the practical feasibility of deploying LLM-based security detection tools in real-world environments. Total Execution Time serves as a primary performance metric, measuring how long an LLM takes to analyse a network configuration and produce a vulnerability assessment. Faster response times indicate better scalability and responsiveness, especially for large-scale enterprise networks where real-time security assessments are necessary. Studies evaluating LLM-generated security policies have included execution time as a benchmark to assess computational efficiency and the overall feasibility of integrating such models into cybersecurity workflows [14].

Finally, interpretability and usability are essential to ensuring that LLM-based security detection systems produce outputs that cybersecurity professionals can easily understand and apply. One key method for evaluating usability is Expert Review, where cybersecurity professionals assess LLM-generated policies for effectiveness, clarity, and relevance. Through this process, experts can identify potential gaps, ambiguities, or inconsistencies in the generated security recommendations and suggest refinements to improve their accuracy and compliance with security regulations. This approach ensures that automated vulnerability detection systems remain practical and actionable for security teams, making LLM-driven security solutions more reliable for enterprise cybersecurity use [9].

By integrating these evaluation metrics—Perfect Predictions for accuracy, Execution Time for performance, Formal Property Verification for robustness, and Expert Review for usability, organizations can comprehensively assess the viability of LLM-based security vulnerability detection in network configurations. These metrics ensure that models are not only accurate but also efficient, robust, and interpretable, making them suitable for deployment in real-world cybersecurity applications.

## Conclusion

The integration of Large Language Models (LLMs) in detecting security vulnerabilities within network configurations marks a transformative step in the field of cybersecurity. By leveraging their advance natural language processing capabilities, LLMs like ChatGPT and GPT-4 excel in analysing complex datasets, automating tasks, and generating actionable insights, significantly enhancing existing methodologies. Tools like LLM-driven synthetic dataset generation ensure the creation of diverse and realistic data, critical for robust model training and evaluation.

Despite all this, challenges such as resource demands, data privacy concerns, and model opacity must be addressed for effective deployment. Enhancing explainability, optimizing resource efficiency, and ensuring compliance with security benchmarks are key steps forward.

In summary, LLMs offer innovative solutions for strengthening network security. By refining their applications and aligning them with ethical and technical standards, they hold immense potential for combating evolving cyber threats and ensuring resilient defences.

[1] H. Li and L. Shan, “LLM-based Vulnerability Detection,” in 2023 International Conference on Human-Centered Cognitive Systems (HCCS), Dec. 2023, pp. 1–4. doi: [10.1109/HCCS59561.2023.10452613](https://doi.org/10.1109/HCCS59561.2023.10452613).

[2] “AI, ML, and Large Language Models in Cybersecurity,” *International Research Journal of Modernization in Engineering Technology and Science*, Mar. 2024, doi: https://doi.org/10.56726/irjmets49546.

‌[3] M. P. Heim, N. Starckjohann, and M. Torgersen, “The Convergence of AI and Cybersecurity: An Examination of ChatGPT’s Role in Penetration Testing and its Ethical and Legal Implications,” Bachelor thesis, NTNU, 2023. Accessed: Jan. 13, 2025. [Online]. Available: <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3076387>

[4] D. Sobania, M. Briesch, C. Hanna, and J. Petke, “An Analysis of the Automatic Bug Fixing Performance of ChatGPT,” in *2023 IEEE/ACM International Workshop on Automated Program Repair (APR)*, May 2023, pp. 23–30. doi: [10.1109/APR59189.2023.00012](https://doi.org/10.1109/APR59189.2023.00012).

[5] N. Shenoy and A. V. Mbaziira, “An Extended Review: LLM Prompt Engineering in Cyber Defense,” in *2024 International Conference on Electrical, Computer and Energy Technologies (ICECET*, Jul. 2024, pp. 1–6. doi: [10.1109/ICECET61485.2024.10698605](https://doi.org/10.1109/ICECET61485.2024.10698605).

[6] K. Okutu and H. Yumetoshi, “Explainability of Large Language Models (LLMs) in Providing Cybersecurity Advice,” Jun. 03, 2024, *OSF*. doi: [10.31219/osf.io/q84rk](https://doi.org/10.31219/osf.io/q84rk).

[7] M. I. Khan, A. Arif, and A. R. A. Khan, “The Most Recent Advances and Uses of AI in Cybersecurity,” *BULLET : Jurnal Multidisiplin Ilmu*, vol. 3, no. 4, Art. no. 4, Oct. 2024.

[8] R. Mondal, A. Tang, R. Beckett, T. Millstein, and G. Varghese, “What do LLMs need to Synthesize Correct Router Configurations?,” Jul. 11, 2023, *arXiv*: arXiv:2307.04945. doi: [10.48550/arXiv.2307.04945](https://doi.org/10.48550/arXiv.2307.04945).

[9] F. Wang, ‘Using Large Language Models to Mitigate Ransomware Threats’, Nov. 08, 2023, *OSF*. doi: [10.31219/osf.io/mzsnh](https://doi.org/10.31219/osf.io/mzsnh).

[10] “Impact of Big Data Analytics and ChatGPT on Cybersecurity,” ResearchGate. Accessed: Nov. 26, 2024. [Online]. Available: <https://www.researchgate.net/publication/369358499_Impact_of_Big_Data_Analytics_and_ChatGPT_on_Cybersecurity>

[11] I. Martín, J. A. Hernández, and Ó. González de Dios, “Netgen: A Fast and Scalable Tool for the Generation and Labeling of Networking Datasets,” in *2019 21st International Conference on Transparent Optical Networks (ICTON)*, Jul. 2019, pp. 1–4. doi: [10.1109/ICTON.2019.8840020](https://doi.org/10.1109/ICTON.2019.8840020).

[12] Y. Huang *et al.*, ‘Advancing Transformer Architecture in Long-Context Large Language Models: A Comprehensive Survey’, Feb. 23, 2024, *arXiv*: arXiv:2311.12351. doi: [10.48550/arXiv.2311.12351](https://doi.org/10.48550/arXiv.2311.12351).

[13] A. P. H and G. Sujatha, ‘System Hardening using CIS Benchmarks’, in *2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, May 2024, pp. 1–6. doi: [10.1109/ACCAI61061.2024.10602274](https://doi.org/10.1109/ACCAI61061.2024.10602274).

[14] D. de-Fitero-Dominguez, E. Garcia-Lopez, A. Garcia-Cabot, and J.-J. Martinez-Herraiz,

‘Enhanced Automated Code Vulnerability Repair using Large Language Models’, *Engineering Applications of Artificial Intelligence*, vol. 138, p. 109291, Dec. 2024, doi: [10.1016/j.engappai.2024.109291](https://doi.org/10.1016/j.engappai.2024.109291).

[15] C. Zhang, H. Liu, J. Zeng, K. Yang, Y. Li, and H. Li, ‘Prompt-Enhanced Software Vulnerability Detection Using ChatGPT’, Apr. 12, 2024, *arXiv*: arXiv:2308.12697. doi: [10.48550/arXiv.2308.12697](https://doi.org/10.48550/arXiv.2308.12697).

[16] J. Cao, M. Li, M. Wen, and S. Cheung, ‘A study on Prompt Design, Advantages and Limitations of ChatGPT for Deep Learning Program Repair’, Apr. 17, 2023, *arXiv*: arXiv:2304.08191. doi: [10.48550/arXiv.2304.08191](https://doi.org/10.48550/arXiv.2304.08191).

[17] Y. Abdullin, D. Molla-Aliod, B. Ofoghi, J. Yearwood, and Q. Li, “Synthetic Dialogue Dataset Generation using LLM Agents,” Jan. 30, 2024, *arXiv*: arXiv:2401.17461. doi: [10.48550/arXiv.2401.17461](https://doi.org/10.48550/arXiv.2401.17461).

[18] R. I. T. Jensen, V. Tawosi, and S. Alamir, “Software Vulnerability and Functionality Assessment using LLMs,” in *2024 IEEE/ACM International Workshop on Natural Language-Based Software Engineering (NLBSE)*, Apr. 2024, pp. 25–28. Accessed: Nov. 19, 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10647161>

[19] L. Long *et al.*, “On LLMs-Driven Synthetic Data Generation, Curation, and Evaluation: A Survey,” Jun. 14, 2024, *arXiv*: arXiv:2406.15126. doi: [10.48550/arXiv.2406.15126](https://doi.org/10.48550/arXiv.2406.15126).

[20] R. Afsharmazayejani, M. M. Shahmiri, P. Link, H. Pearce, and B. Tan, “Toward Hardware Security Benchmarking of LLMs,” in *2024 IEEE LLM Aided Design Workshop (LAD)*, Jun. 2024, pp. 1–7. doi: [10.1109/LAD62341.2024.10691745](https://doi.org/10.1109/LAD62341.2024.10691745).