GENERATIVE AI (LLMS) FOR DETECTING ABNORMAL BEHAVIOR THROUGH EXECUTION TRACE ANALYSIS

1 Context and Approach

This project aims to study the contributions of Generative Artificial Intelligence (AI) and Large Language Models (LLMs) to certain aspects of defensive computer security. This internship provides an opportunity to initiate research work that will continue in a thesis, in collaboration with DiverSE Inria and the Exploration and Research Laboratory in Detection (LED) at ANSSI.

Ambition. The objective is to create a monitoring program capable of automatically detecting and characterizing when a computer system deviates from its nominal behavior (including in its interactions with the outside). The supervisor can then raise alerts. The result of the analysis is an actionable report for experts.

Approach and Methodology. In this context, LLMs show promise for analyzing execution traces (by classifying, summarizing, or extracting important information from one or more traces). LLMs have recently been at the forefront with initiatives and tools such as BERT, BLOOM, GPT-3, GPT-4, PaLM, Alphacode, Code-Parrot, Codex, ChatGPT, and CoPilot. The ability of LLMs to process or synthesize technical artifacts (code, semi-structured documents, or traces) encourages us to explore their use in a cybersecurity context [Liu et al., 2021, Steenhoek et al., 2022, Zhou et al., 2022]. It is then a matter of studying LLMs in the context of **detecting abnormal behaviors of computer programs and systems** [Vaccaro and Liepins, 1989, Oliner et al., 2011, Li et al., 2017, Sultana et al., 2019, Khraisat et al., 2019, Thakkar and Lohiya, 2023].

To achieve this, **execution traces** (e.g., logs) of various types (system calls [da Costa et al., 2017, Nissim et al., 2018], memory [Panker and Nissim, 2021], network exchanges/packets [Sikos, 2020], etc.) will be collected. Execution traces can be seen as text obeying certain rules: they are semi-structured data.

Large Language Models have demonstrated their ability to process this type of data in an agnostic and generic manner, i.e., without the need for syntactic or grammatical analysis. Due to their versatility, LLMs should have excellent capability to classify anomalous behaviors (i.e., executions) of programs and systems, thus enabling the detection of errors, bugs, malicious software, or cyber-attacks.

The implemented system should take into account existing tools, catalogs, and vulnerability databases to link detections, as much as possible, to these vulnerabilities (e.g., CVEs). Embedding techniques and information retrieval methods need to be developed to make the interaction between LLMs, traces, and data sources effective [Liu et al., 2021, Andrus et al., 2022]. Our vision is to synthesize reports that manage to match traces with vulnerability information; these reports can be utilized by experts to make defensive decisions;

Project Architecture. Figure 1 provides a general overview of the project. Given a cyber system (black box), it is possible to observe it through traces (gray boxes). From a defensive standpoint, these traces can be analyzed to quantify and qualify the cyber system in terms of vulnerabilities or ongoing attacks.

2 Internship Work

The work to be carried out is structured into three axes:

- Study the bibliography to gain a good understanding of the relevant domains and existing tools. The references cited in this document are a starting point, but the state of the art evolves rapidly, whether it's on the side of LLMs, software engineering, or security.
- Based on the bibliographic work and in collaboration with ANSSI, design a playground with cyber systems, traces, etc., to be able to experiment with LLMs. Open data or realistic scenarios can be used, and a test bench

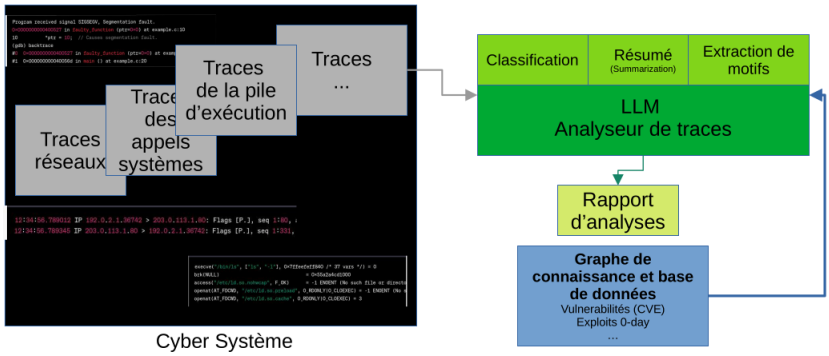

FIG. 1 – *Defensive cyber operations (trace analysis) based on LLMs*

will be established with the ambition to eventually have reference results for detecting abnormal behaviors from execution traces.

– Implement an experimental prototype of an LLM detecting abnormal behaviors by analyzing traces of a given type. This prototype will be developed by the intern based on the articles and by reusing libraries or available tools as open-source software. Experimental results will be reported, analyzed, and discussed.

The aim of the internship is to familiarize oneself with the subject and obtain initial results that will then be further developed **as part of a 3-year thesis, still in partnership between DiverSE Inria and ANSSI.**

3 Supervision and Contacts

The internship will take place within the DiverSE team at Inria/IRISA Rennes, in collaboration with LED at ANSSI.

The DiverSE team has internationally recognized expertise in software engineering, software variability, and automatic techniques for software. DiverSE has a strong activity in cybersecurity through past or ongoing collaborations, for example recently with Software Heritage (SWH-Sec). DiverSE is co-responsible for an Inria challenge on LLMs and software engineering.

The National Cybersecurity Agency of France (ANSSI) is the national authority in cybersecurity. Its mission is to understand, prevent, and respond to cyber risk. LED is responsible for the domain of detection and analysis of cyber attacks against information systems, including intrusion detection, analysis of compromised systems or malicious software.

Supervisors:
Mathieu ACHER, Professor at INSA Rennes (mathieu.acher@inria.fr), DiverSE.
Olivier ZENDRA, Inria Research Scientist (olivier.zendra@inria.fr), DiverSE.
Romain BRAULT, Data Science Expert at ANSSI (romain.brault@ssi.gouv.fr), LED.

The aim of the internship is to prepare the candidate for research work that will continue with a three-year thesis, carried out in collaboration between DiverSE Inria and ANSSI.

Références

[Andrus et al., 2022] Andrus, B. R., Nasiri, Y., Cui, S., Cullen, B., and Fulda, N. (2022). Enhanced story comprehension for large language models through dynamic document-based knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10436–10444.

[da Costa et al., 2017] da Costa, V. G. T., Barbon, S., Miani, R. S., Rodrigues, J. J. P. C., and Zarpelão, B. B. (2017). Detecting mobile botnets through machine learning and system calls analysis. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6.

[Khraisat et al., 2019] Khraisat, A., Gondal, I., Vamplew, P., and Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecur*, 2(20).

[Li et al., 2017] Li, T., Jiang, Y., Zeng, C., Xia, B., Liu, Z., Zhou, W., Zhu, X., Wang, W., Zhang, L., Wu, J., Xue, L., and Bao, D. (2017). FLAP: an end-to-end event log analysis platform for system management. In *Proceedings of*

the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017, pages 1547–1556. ACM.

[Liu et al., 2021] Liu, J., Shen, D., Zhang, Y., Dolan, B., Carin, L., and Chen, W. (2021). What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

[Nissim et al., 2018] Nissim, N., Lapidot, Y., Cohen, A., and Elovici, Y. (2018). Trusted system-calls analysis methodology aimed at detection of compromised virtual machines using sequential mining. *Knowledge-Based Systems*, 153:147–175.

[Oliner et al., 2011] Oliner, A. J., Ganapathi, A., and Xu, W. (2011). Advances and challenges in log analysis. *Queue*, 9:30 – 40.

[Panker and Nissim, 2021] Panker, T. and Nissim, N. (2021). Leveraging malicious behavior traces from volatile memory using machine learning methods for trusted unknown malware detection in linux cloud environments. *Knowledge-Based Systems*, 226:107095.

[Sikos, 2020] Sikos, L. F. (2020). Packet analysis for network forensics: A comprehensive survey. *Forensic Science International: Digital Investigation,*, 32:200892.

[Steenhoek et al., 2022] Steenhoek, B., Rahman, M. M., Jiles, R., and Le, W. (2022). An empirical study of deep learning models for vulnerability detection. *arXiv preprint arXiv:2212.08109*.

[Sultana et al., 2019] Sultana, N., Rao, A., Jin, Z., Pashakhanloo, P., Zhu, H., Yegneswaran, V., and Loo, B. T. (2019). Trace-based behaviour analysis of network servers. In Lutfiyya, H., Diao, Y., Zincir-Heywood, A. N., Badonnel, R., and Madeira, E. R. M., editors, *15th International Conference on Network and Service Management, CNSM 2019, Halifax, NS, Canada, October 21-25, 2019*, pages 1–5. IEEE.

[Thakkar and Lohiya, 2023] Thakkar, A. and Lohiya, R. (2023). A review on challenges and future research directions for machine learning-based intrusion detection system. *Arch Computat Methods Eng*.

[Vaccaro and Liepins, 1989] Vaccaro, H. and Liepins, G. (1989). Detection of anomalous computer session activity. In *Proceedings. 1989 IEEE Symposium on Security and Privacy*, pages 280–289.

[Zhou et al., 2022] Zhou, Z., Bo, L., Wu, X., Sun, X., Zhang, T., Li, B., Zhang, J., and Cao, S. (2022). Spvf: security property assisted vulnerability fixing via attention-based models. *Empirical Software Engineering*, 27(7):171.