MLSMM: Machine Learning Security Maturity Model

Felix Viktor Jedrzejewski [1] **Davide Fucci** [1] **Oleksandr Adamov** [1]

Abstract

Assessing the maturity of security practices during the development of Machine Learning (ML) based software components has not gotten as much attention as traditional software development. In this Blue Sky idea paper, we propose an initial Machine Learning Security Maturity Model (`MLSMM`) which organizes security practices along the ML-development lifecycle and, for each, establishes three levels of maturity. We envision `MLSMM` as a step towards closer collaboration between industry and academia.

1. Introduction

The release of ChatGPT and its fast popularity greatly contribute to the discussion about the role of AI in our society. For example, several studies conducted on behalf of the German Federal Office for Information Security[1] discuss the importance of regulations requiring industry to demonstrate their effort in addressing Machine Learning (ML) Security in their development practices. The last decade of Adversarial Machine Learning (AML) research (Biggio & Roli, 2018; Cinà et al., 2022) introduced multiple attacks and defense strategies. While attackers seem to use publicly-available resources (Tidjon & Khomh, 2022), multiple interviews with ML practitioners show that the industry is ill-prepared to handle potential attacks to its ML-based systems (Kumar et al., 2020) and reluctant to introduce security measures. In a few instances, AML researchers have directly approached industry practitiones (Boenisch et al., 2021; Mink et al., 2023; Grosse et al., 2023), conducted more realistic (e.g., in vivo) studies (Apruzzese et al., 2022), and proposed actionable ML development models incorporating security measures (Zhang & Jaskolka, 2022). In the *traditional* software development community, the need for companies evaluating and incorporating security in their lifecycle has been addressed by several security maturity models (Teodoro & Serrao, 2011; Lipner, 2004; Weir et al., 2021). Nevertheless, besides the heavyweight ISO21827, there is a lack of ML security maturity models studied in academia and adopted in the industry. Based on the perspective regulations, attackers using AML, and the current low awareness about it in the industry based on empirical evaluations, we propose a lightweight domain-agnostic Machine Learning Security Maturity Model (`MLSMM`). The goals of the model are to i) evaluate the state-of-practice concerning the security of ML-development process within an organization, ii) support the organization in creating a roadmap to improve and prioritize their ML security stance in specific areas, and iii) increase ML security awareness across different teams (e.g., developers, architects, quality assurance). `MLSMM` is based on the established Security Assurance Maturity Model (SAMM[2]) proposed by the Open Worldwide Application Security Project (OWASP) and the Adversarial Threat Landscape for Artificial Intelligence Systems (ATLAS) taxonomy[3] by MITRE. We foresee that `MLSMM` will reduce the gaps between academia and industry fostering closer collaboration in which the first develops supportive tools and the latter provides real case scenarios, data, and study validation opportunities. Additionally, the roadmap `MLSMM` can represent a starting point for compliance and certification procedures in the future.

OWASP SAMM is a state-of-practice maturity model facilitating the measurement, analysis, and improvement of software products' security and addressing essential stages in the software development process. Its content is based on the experience and domain knowledge of industry security experts (Brasoveanu et al., 2022). Software engineering for ML follows a different development process than traditional software products. Accordingly, for developing `MLSMM` , we follow Amerishi et al.'s ML workflow (Amershi et al., 2019) consisting of nine stages being categorized as either model-oriented (i.e., model requirements, feature engineering, training, evaluation, deployment, and monitoring) or data-oriented (i.e., collection, cleaning, and labeling). MITRE ATLAS is a taxonomy associating activities to mit-

[1]Department of Software Engineering, Blekinge Institute of Technology, Karlskrona, Sweden. Correspondence to: Felix Jedrzejewski <felix.jedrzejewski@bth.se>.

2^{nd} AdvML Frontiers workshop at 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]`https://www.bsi.bund.de/EN/Service-Navi/Publikationen/Studien/Projekt_P464/Projekt_P464_node.html`

[2]`https://owaspsamm.org/`

[3]`https://atlas.mitre.org/`

Table 1. Excerpt of the proposed Machine-Learning Security Maturity Model.

ML Phase	Security Practice	Level 1	Level 2	Level 3
Model Training	Model Hardening https://atlas.mitre.org/mitigations/AML.M0003/	Model hardening is performed based on best-efforts (e.g., simple defense against model erosion)	Model hardening is standardized within the organization	Models are proactively hardened within the organization
	Use Ensemble Methods https://atlas.mitre.org/mitigations/AML.M0006	Simple ensemble models (e.g., voting) are introduced	Ensemble approaches are introduced or removed based on specific threats against the model	Ensembles are continuously shuffled to avoid leaking information to attackers

igate attacks to real-world adversary tactics (Zhang et al., 2023).

2. MLSMM Prototype

MLSMM combines state-of-practice maturity evaluation techniques for software product security with state-of-the-art mitigation techniques assigned to particular stages within the ML development process. Following OWASP SAMM, our proposed MLSMM is prescriptive in nature—i.e., it provides high-level guidance and advices to an organization—rather than descriptive—i.e., providing a summary of what other organizations do.[4] Table 1 presents an excerpt from the *Model Training* phase of ML-components development. The model is hierarchical; it starts with the nine phases of ML development (Amershi et al., 2019) each with a variable number of security practices from MITRE ATLAS associated with them. Each security practice has three possible maturity levels where the activities on a lower level are typically easier to execute and require less formalization than the ones on a higher level. At this initial stage, MLSMM consists of 19 practices. A complete draft is available on the project website. [5] Similarly to SAMM, we propose a simple questionnaire measuring the maturity levels. We use ordinal-value answers to assess how well an organization fulfills the activities associated with a level. Based on the example in Table 1, an organization reaches *Maturity Level 1* in *Model Hardening* once it performs activities such as adversarial training and network distillation. However, these activities are performed ad-hoc and in an unstructured fashion [6]. The organization reaches the next level once the answers to the questionnaire show evidence that hardening is a standardized practice for *every* model. The final level implies that model hardening is part of the model training process *by design* rather than done in reaction to specific events. In Table 1, the next security practice assessed for *Model Training* is *Use Ensemble Methods*. The lowest maturity level indicates the presence of simple ensemble methods introduced during training without providing any security context. Level 2 is reached once the use of ensemble methods is grounded in security activities identified before model development, such as threat modeling. At Maturity Level 3, the organization continuously applies ensemble method shuffling to avoid information leakage. MLSMM does not insist that an organization achieves the maximum maturity in every category as each organization should determine the target level, for each *Security Practice*, that best fits their needs.

3. Conclusion and Future Work

We presented our idea for MLSMM —an actionable, domain- and model-agnostic security maturity model to assess ML components developments based on existing industrial practices and procedures. Our next steps are i) expand the model to cover additional ML security practices not included within MITRE ATLAS, ii) create a questionnaire to gather evidence to instantiate the model in practice, iii) validate the model and questionnaire with our industry partners regarding their usefulness and usability.

[4]For an example of descriptive security maturity model see BSIMM https://bsimm.com

[5]https://anonymous.4open.science/r/MLSMM-EA81/

[6]A maturity level of zero indicates the complete lack of such activities

References

Amershi, S., Begel, A., Bird, C., DeLine, R., Gall, H., Kamar, E., Nagappan, N., Nushi, B., and Zimmermann, T. Software engineering for machine learning: A case study. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, pp. 291–300. IEEE, 2019.

Apruzzese, G., Andreolini, M., Ferretti, L., Marchetti, M., and Colajanni, M. Modeling realistic adversarial attacks against network intrusion detection systems. *Digital Threats: Research and Practice (DTRAP)*, 3(3):1–19, 2022.

Biggio, B. and Roli, F. Wild patterns: Ten years after the rise of adversarial machine learning. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2154–2156, 2018.

Boenisch, F., Battis, V., Buchmann, N., and Poikela, M. "i never thought about securing my machine learning systems": A study of security and privacy awareness of machine learning practitioners. In *Proceedings of Mensch und Computer 2021*, pp. 520–546. 2021.

Brasoveanu, R., Karabulut, Y., and Pashchenko, I. Security maturity self-assessment framework for software development lifecycle. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pp. 1–8, 2022.

Cinà, A. E., Grosse, K., Demontis, A., Vascon, S., Zellinger, W., Moser, B. A., Oprea, A., Biggio, B., Pelillo, M., and Roli, F. Wild patterns reloaded: A survey of machine learning security against training data poisoning. *arXiv preprint arXiv:2205.01992*, 2022.

Grosse, K., Bieringer, L., Besold, T. R., Biggio, B., and Krombholz, K. Machine learning security in industry: A quantitative survey. *IEEE Transactions on Information Forensics and Security*, 18:1749–1762, 2023. doi: 10.1109/TIFS.2023.3251842.

Kumar, R. S. S., Nyström, M., Lambert, J., Marshall, A., Goertzel, M., Comissoneru, A., Swann, M., and Xia, S. Adversarial machine learning-industry perspectives. In *2020 IEEE security and privacy workshops (SPW)*, pp. 69–75. IEEE, 2020.

Lipner, S. The trustworthy computing security development lifecycle. In *20th Annual Computer Security Applications Conference*, pp. 2–13. IEEE, 2004.

Mink, J., Kaur, H., Schmüser, J., Fahl, S., and Acar, Y. "security is not my field, i'ma stats guy": A qualitative root cause analysis of barriers to adversarial machine learning defenses in industry,". In *Proceedings of the 32nd USENIX Security Symposium*, 2023.

Teodoro, N. and Serrao, C. Web application security: Improving critical web-based applications quality through in-depth security analysis. In *International Conference on Information Society (i-Society 2011)*, pp. 457–462. IEEE, 2011.

Tidjon, L. N. and Khomh, F. Threat assessment in machine learning based systems, 2022.

Weir, C., Migues, S., Ware, M., and Williams, L. Infiltrating security into development: Exploring the world's largest software security study. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pp. 1326–1336, 2021.

Zhang, X. and Jaskolka, J. Conceptualizing the secure machine learning operations (secmlops) paradigm. In *2022 IEEE 22nd International Conference on Software Quality, Reliability and Security (QRS)*, pp. 127–138. IEEE, 2022.

Zhang, Y., Malacaria, P., Loukas, G., and Panaousis, E. Cross: a framework for cyber risk optimisation in smart homes. *Computers & Security*, 130:103250, 2023.