Membership Inference Attack: A Middleware-Based Approach for Privacy Preservation and Attack Mitigation in Machine Learning Systems

Ataque de Inferencia de Pertenencia: A Middleware-Based para la Preservación de la Privacidad y la Mitigación de Ataques en Sistemas de Aprendizaje Automático

DOI: http://dx.doi.org/10.17981/cesta.05.02.2024.02

Scientific Research Article.
Date of Receipt: 22/10/2024 Date of Acceptance: 01/11/2024.

Yair Enrique Rivera
Corporación Universitaria Americana. Barranquilla, (Colombia)
yrivera@americana.edu.co

Julio Jiménez
Universidad Simón Bolívar. Barranquilla, (Colombia)
j.jimenez@usb.edu.co

How to cite:
Y. Rivera, J. Jiménez "Membership Inference Attack: A Middleware-Based Approach for Privacy Preservation and Attack Mitigation in Machine Learning Systems", J. Comput. Electron. Sci.: Theory Appl., vol. 5 no. 2, pp. 28-35, 2024. DOI: https://doi.org/10.17981/cesta.05.02.2024.02

Abstract

This article explores using middleware as a robust solution to mitigate Membership Inference Attacks (MIA) in Machine Learning (ML) systems. These attacks allow attackers to deduce whether a specific data point was part of a model's training set, compromising data confidentiality and privacy. The proposed approach uses middleware that implements data randomization techniques, prediction obfuscation, dynamic regularization, and real-time monitoring to prevent such attacks. The results reveal that this middleware architecture provides an additional layer of security, minimizing the risk of data exposure while maintaining model accuracy. This research offers a novel perspective on using middleware to mitigate membership inference attacks, providing valuable insights into machine learning security.

Keywords— Membership Inference Attacks, Machine Learning, Cybersecurity, Computer Attack Mitigation, Middleware.

Resumen

Este artículo explora el uso de middleware como solución robusta para mitigar los Ataques de Inferencia de Membresía (MIA) en sistemas de Aprendizaje Automático (ML). Estos ataques permiten a los atacantes deducir si un punto de datos específico formó parte del conjunto de entrenamiento de un modelo, comprometiendo la confidencialidad y privacidad de los datos. El enfoque propuesto se centra en el uso de middleware que implementa técnicas de aleatorización de datos, ofuscación de predicciones, regularización dinámica y monitorización en tiempo real para prevenir estos ataques. Los resultados revelan que esta arquitectura de middleware proporciona una capa adicional de seguridad, minimizando el riesgo de exposición de los datos y manteniendo al mismo tiempo la precisión del modelo. Esta investigación ofrece una perspectiva novedosa sobre el uso de middleware para mitigar los ataques de inferencia de membresía, proporcionando valiosos conocimientos sobre la seguridad del aprendizaje automático.

Palabras clave— Ataques de inferencia de membresía, Aprendizaje Automático, Ciberseguridad, Mitigación de ataques informáticos, Middleware.

I. INTRODUCTION

Machine learning has revolutionized many fields in recent years, providing transformative insights and capabilities in domains such as healthcare, finance, autonomous systems, and natural language processing. However, with the proliferation of these advanced technologies comes an increasing concern regarding data privacy, one of the most pressing challenges that has emerged.

Machine learning models are vulnerable to various attacks, particularly membership inference attacks (MIAs). MIAs allow adversaries to determine whether a specific data point was used to train a machine-learning model. This attack poses significant risks, especially in applications dealing with sensitive information, such as personal medical records or financial data [1].

An attacker's ability to deduce training data membership undermines the privacy guarantees often promised by machine learning models, especially those deployed in real-world scenarios through APIs. These attacks become particularly potent in overfitted models, where subtle distinctions between training and non-training data are more easily exploitable. Existing techniques to safeguard models, such as differential privacy or adversarial training, have proven to be only partially effective or computationally expensive, leaving room for further exploration of robust solutions [2].

This paper proposes an innovative middleware-based approach to address these vulnerabilities. As an intermediary layer, middleware can be leveraged to manage the data flow between external users and the machine learning model. By implementing real-time data randomization, prediction obfuscation, and dynamic regularization techniques, this middleware significantly reduces models' susceptibility to membership inference attacks. In addition, the middleware includes a real-time anomaly detection system to monitor suspicious queries, further enhancing security [3].

This research highlights the potential of middleware as a security layer and contributes to the broader discourse on machine learning model vulnerabilities and privacy preservation. By analyzing the effectiveness of this approach through experimental evaluation, this study provides valuable insights into the trade-offs between model accuracy and privacy in modern machine learning systems. This paper also contributes to understanding how middleware can be systematically designed to combat complex attack vectors, potentially influencing future developments in secure machine-learning applications.

II. RELATED WORKS

Recent web application security efforts have identified and addressed critical vulnerabilities using innovative methodologies and specific tools based on Open Web Application Security Project (OWASP) principles. Thus, some notable research is presented below.

Studied vulnerabilities in open banking architectures by analyzing financial APIs using the OWASP Top 10 standard. They proposed a technological framework that employs tools like Flask and Heroku, enhancing the secure integration of banking APIs [4]. Researchers developed threat models to understand the vulnerabilities associated with Cross-Site Request Forgery (CSRF) attacks. This work illustrated' real-world scenarios and created attack tree models that facilitate the validation of protection features in web applications [5]. In [6], a comparative analysis of tools such as Nikto and OWASP ZAP is performed to identify security weaknesses in web applications. Their research concluded that Nikto is more effective in detecting critical vulnerabilities.

In addition, the researchers proposed a machine learning-based approach for detecting SQL injection attacks (SQLi). This method, called PALOSDM, significantly improved the attack detection rate [7]. The paper presents an automated web application security testing system using tools such as OWASP Dependency Check and ZAP using Docker containers. This approach reduced false positives and facilitated security evaluation [8]. A methodology based on the shared responsibility model of computer security is proposed for the secure development of cloud applications. It ensures the integrity of applications migrating to cloud services [9]. Another work explored web application firewall (WAF) evasion techniques, highlighting configurations and levels of paranoia in the OWASP CRS rule set. Their results highlighted the importance of designing more robust firewall strategies [10].

The investigation developed AS-PAHI, an educational tool to raise awareness of security and privacy risks in-home IoT devices. Their academic program significantly improved participants' knowledge [11]. [12], emphasized the importance of secure coding in software development. Their approach includes practical exercises and tools like WebGoat to teach developers to avoid common vulnerabilities. Finally, reviewed advanced techniques for preventing SQL injection attacks using machine learning introduce an approach that significantly improves vulnerability detection accuracy [13].

III. RESEARCH FUNDAMENTALS

A. Fundamentals of membership inference attacks membership inference

Definition and Nature of Attack

A MIA occurs when an attacker exploits differences in a model's behavior by processing data from the training set rather than new data. By performing multiple queries to the model, the attacker can infer whether a specific data point was used to train it, compromising user privacy [14].

B. Operating Mechanism

Membership inference attacks are based on the attacker's ability to obtain the model's responses and detect patterns in the results. These patterns can reveal the inclusion of specific data in the training set, especially when the model is overfitted or misconfigured in privacy [15].

C. Middleware proposal to mitigate attacks

Middleware Architecture

The proposed middleware is an intermediate layer that manages the interaction between the machine learning model and the users or external applications. This middleware acts as an access controller, implementing various security techniques before queries reach the model and the results are returned to the user.

Techniques Implemented in Middleware

Prediction Obfuscation

The middleware includes functionalities that slightly alter the model's predictions by adding random noise or implementing differential privacy techniques. These modifications ensure that the returned results are inaccurate, making it difficult for an attacker to extract accurate information about the training data [1].

Randomization of Input Data

Another key feature of middleware is the randomization of input data before it is used for queries to the model. This means that the data is randomized even if an attacker attempts to perform repeated queries with slight variations. The model responses vary with each request, making it more difficult to infer patterns.

Dynamic Regularization

Middleware also manages the application of L1 and L2 regularization techniques in the model, dynamically adjusting the regularization parameters based on the query type. This reduces the model's tendency to overfit the training data, minimizing the attacker's ability to infer whether a specific data point was present.

Real-Time Anomaly Detection and Monitoring

The middleware includes a continuous monitoring layer that analyzes queries to the model in real time to detect suspicious or malicious patterns. If behavior indicative of an attack is detected, the middleware can block queries or alter results before sending them back to the attacker [2].

IV. METHODOLOGY

A. Mathematical model for the mitigation of MIA

To demonstrate its effectiveness, the mitigation of membership inference attacks through middleware can be modeled mathematically. The process involves introducing noise into the model predictions and randomizing the inputs. An extended mathematical model that formalizes these concepts is presented below.

Obfuscation Predictions

Let *f(x)* be the prediction function of a machine learning model, where x represents the input data. The middleware introduces a noise term r(x) in the predictions so that the new prediction function (1) is:

$$f'(x) = f(x) + r(x) \qquad (1)$$

where *r(x)* is a normally distributed random variable with mean zero and variance σ^2.

Impact of Noise on Prediction Variance

To analyze the impact of noise on predictions, let us consider the total variance of *f'(x)*. Since *r(x)* and *f(x)* are independent, the variance of *f'(x)* is given by (2):

$$\mathrm{Var}\big(f'(x)\big) = \mathrm{Var}\big(f(x)\big) + \mathrm{Var}\big(r(x)\big) = \sigma_f\ \hat{}\ 2 + \sigma^2, \qquad (2)$$

Where σ^2 is the prediction's original variance, and σ^2 is the variance of the introduced noise. This increase in variance makes it difficult for an attacker to accurately identify the training data, as it increases the uncertainty in the predictions.

Input Data Randomization

The middleware also randomizes the model input data. Suppose X is the original input data set. The middleware generates a randomized set X' as follows (3):

$$X' = X + \epsilon \ (3)$$

where ϵ is a slight random disturbance with distribution *N(0, σ²)*.

Statistical Analysis of Randomization

The perturbation ϵ introduces sufficient input variability so that the model responses are nondeterministic, reducing susceptibility to inference attacks. The new input X' follows a distribution with increased variance (4):

$$Var(X') = Var(X) + \sigma^2.(4)$$

Evaluating the Probability of Success of the Attack

The probability of a successful membership inference attack depends on the attacker's ability to distinguish between model predictions for training and untrained data. If D represents the distance between the model predictions for training and untrained data, the probability of a successful PSucces attack can be modeled by a sigmoid function:

$$P_{\text{Succes}} = \frac{1}{1+e^{-(D-\mu)}} \ (5)$$

where μ is the threshold that separates the predictions for training and untrained data. Introducing noise and randomization through the middleware increases the variability in D, reducing the probability of a successful attack.

B. Risk and threat evaluation

Risk Factors Associated with Middleware Usage

While middleware provides an additional layer of security, it also introduces certain risks if not implemented correctly. Possible vulnerabilities include exposure to middleware-to-middleware attacks or manipulation of the middleware by malicious actors.

Attack and Mitigation Example

An attacker may attempt to make multiple queries to a publicly exposed machine learning model to infer users' financial data. With the middleware running, the model's predictions are obfuscated with noise, making the answers inaccurate and thwarting attack attempts. In addition, the middleware detects an unusual volume of queries from the same source, triggering preventative measures, including limiting access to the model.

V. EXPERIMENTAL DESIGN

To evaluate the effectiveness of the proposed middleware in mitigating membership inference attacks (MIA), an experiment was designed to compare the probability of success of an attacker under two conditions: with and without middleware. The main objective is to quantify the impact of obfuscation and randomization techniques on model predictions and their relationship with attack success.

A. Experiment Description

The experiment was conducted using a sample data set of 10,000 input instances. The following conditions were evaluated:

Without Middleware: The model runs without applying obfuscation or randomization techniques.

With Middleware: The model uses middleware to introduce noise in the predictions and randomize the inputs.

The distance D between the predictions for training and untrained data was measured for each condition. From these values, the probability of a successful membership inference attack P_{Succes} was calculated. The results were compared using a boxplot to analyze the dispersion and variability of the data.

VI. RESULTS

Middleware significantly reduced the probability of attack success by increasing the variability in model predictions, as shown in Fig. 1.

The results shown in Figure 1 clearly show the impact of middleware in reducing the probability of success of membership inference attacks. The increased variability in predictions with middleware introduces enough uncertainty to hinder the attacker's ability to differentiate between training and untrained data.

This suggests that the techniques employed by the middleware, such as obfuscation and randomization, effectively improve model privacy without significant performance degradation. The box plot also shows a more excellent dispersion of success probabilities in the presence of middleware, indicating a more randomized and unpredictable outcome for potential attackers.

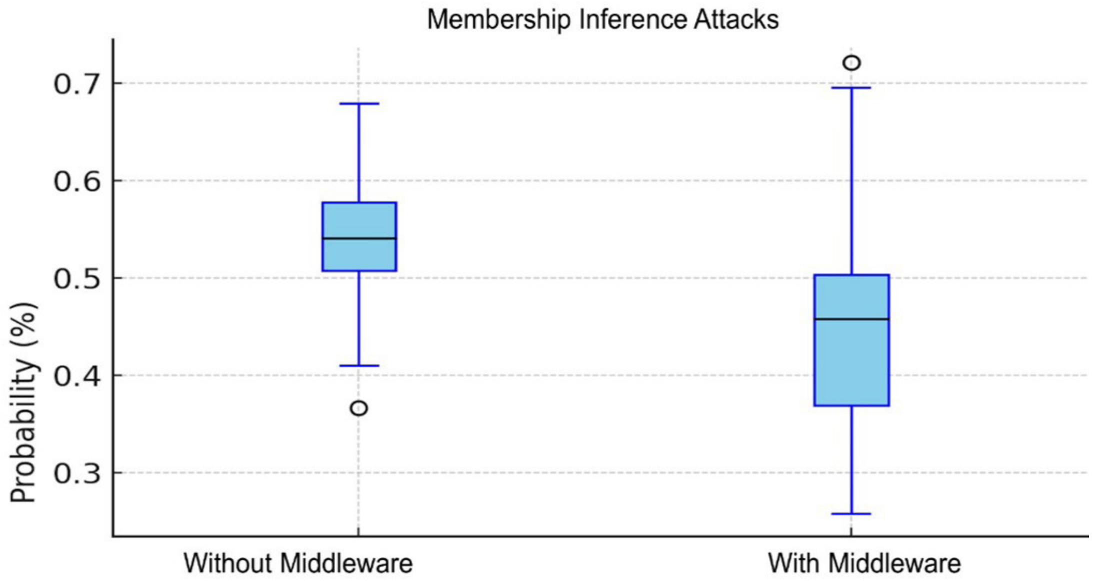

Fig. 1. Comparison of the probability of success of the Membership Inference Attack.

VII. CONCLUSIONS

This paper has presented a middleware-based approach to mitigate membership inference attacks in machine learning systems. By implementing prediction obfuscation, input randomization, dynamic regularization, and real-time monitoring, the middleware significantly reduces the probability of successful attacks while maintaining the integrity of model predictions. This approach offers a robust and scalable solution to the growing privacy vulnerabilities in machine learning models, making a valuable contribution to secure AI applications.

Furthermore, this middleware architecture provides insights into the tradeoff between security measures and model performance, showing that privacy preservation does not necessarily imply a tradeoff with accuracy.

Future research could extend this work by exploring how different machine learning models respond to middleware protections, further improving the security of AI systems in various application domains.

FUNDING

This research was developed with the resources of Universidad Corporación Universitaria Americana, Colombia.

AUTHORS' CONTRIBUTION

The authors' contributions to this article are as follows:
Yair Rivera: Software development, methodology, research, visualization, and supervision.
Julio Jiménez: Research, data processing, manuscript preparation, review, and editing.
All authors participated in reviewing the results and approved the final version of the manuscript.

CONFLICT OF INTERESTS

The authors hereby declare that there are no conflicts of interest about the reporting of this study.

REFERENCES

[1] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP). IEEE*, 2017, pp. 3–18.

[2] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," in *2019 IEEE Symposium on Security and Privacy (SP). IEEE*, 2019, pp. 739–753.

[3] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "*ML-Leaks: Model and data independent membership inference attacks and defenses on machine learning models*," in Network and Distributed System Security Symposium (NDSS), 2019.

[4] A. Kumar and A. Gandhi, "A study using owasp on secure open banking architecture," in *2023 3rd Interna- tional Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2023, pp. 2062– 2067.

[5] X. Lin, P. Zavarsky, R. Ruhl, and D. Lindskog, "Threat modeling for csrf attacks," in *2009 International Conference on Computational Science and Engineering*, 2009, pp. 486–491.

[6] R. S. Devi and M. M. Kumar, "Testing for security weakness of web applications using ethical hacking," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)*, 2020, pp. 354–361.

[7] A. A. Ashlam, A. Badii, and F. Stahl, "A novel approach exploiting machine learning to detect sqli attacks," in *2022 5th International Conference on Advanced Systems and Emergent Technologies (ICASET)*, 2022, pp,513 − 517.

[8] P. P. W. Pathirathna, V. A. I. Ayesha, W. A. T. Imihira, W. M. J. C. Wasala, N. Kodagoda, and E. A. T. D. Edirisinghe, "Security testing as a service with docker containerization," in *2017 11th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, 2017, pp. 1–7.

[9] G. J. Nieves Arreaza, "Methodology for developing se- cure apps in the clouds. (mdsac) for ieeecs conferences," in *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/ 2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 2019, pp. 102–106.

[10] K. Nagendran, S. Balaji, B. A. Raj, P. Chanthrika, and R. G. Amirthaa, "Web application firewall evasion techniques," in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2020, pp. 194–199.

[11] V. Visoottiviseth, T. Khengthong, K. Kesorn, and J. Patcharadechathorn, "Aspahi: Application for security and privacy awareness education for home iot devices," in *2021 25th International Computer Science and Engineering Conference (ICSEC)*, 2021, pp. 388–393.

[12] N. A. Aziz, S. N. Z. Shamsuddin, and N. A. Hassan, "Inculcating secure coding for beginners," in *2016 International Conference on Informatics and Computing (ICIC)*, 2016, pp. 164–168.

[13] A. Rai, M. M. I. Miraz, D. Das, H. Kaur, and Swati, "Sql injection: Classification and prevention," in *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM),* 2021, pp. 367–372.

[14] S. Truex, L. Liu, M. E. Gursoy, L. Yu, and W. Wei, "Towards demystifying membership inference attacks," arXiv preprint arXiv:1807.09173, 2018.

[15] L. Li and Z. Zhang, "Membership leakage in label-only exposures," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 884–901.

Author 1: He is a researcher and professor in Cybersecurity and Telecommunications at Corporacion Universitaria Americana, Barranquilla, Colombia. I hold a Ph.D. in Computer Science, an M.Sc. in Telematics and Telecommunications, and a B.Sc. in Systems Engineering. Their research focuses on network security, forensic computing, IoT, blockchain technology, and decentralized coding. He specializes in server hardening, risk mitigation, intrusion detection, and distributed security architecture. I have worked extensively on Random Linear Network Coding (RLNC), Fulcrum coding, and security frameworks for decentralized architectures. He is an IEEE member and a recognized researcher by Minciencias. Their contributions include decentralized coding techniques for resilient and fault-tolerant network infrastructures, secure data transmission, and distributed computing in IoT and edge environments.

Author 2: He is a teacher in the district of Barranquilla, Colombia. He has a master's degree in education and completed his doctorate in Educational Sciences at the Universidad Simón Bolívar. His research interests are mainly focused on mathematics education.