Global Journal of Engineering Innovations & Interdisciplinary Research

Original Article

Federated Learning-Based Privacy-Preserving Malware Detection Framework Using Multi-Source Cybersecurity Datasets

Bachu Anusha

Lecturer, Department of Computer Science, Sri Aurobindo Degree College, Jangaon, India.

Correspondence

Bachu Anusha

Lecturer, Department of Computer Science, Sri Aurobindo Degree College, Jangaon

- Received Date: 08 Jan 2026
- Accepted Date: 25 Jan 2026
- Publication Date: 15 Mar 2026

Keywords

Federated Learning, Malware Detection, Privacy-Preserving Machine Learning, Differential Privacy, Secure Aggregation, Multi-Source Cybersecurity Data

Copyright

© 2026 Authors. This is an open- access article distributed under the terms of the Creative Commons Attribution 4.0 International license.

Abstract

The increasing prevalence and sophistication of malware attacks in distributed and heterogeneous computing environments necessitate advanced detection mechanisms that ensure both high accuracy and data privacy. Traditional centralized malware detection systems are limited by privacy concerns, regulatory constraints, and the inability to effectively utilize data from multiple sources. To address these challenges, this study proposes a Privacy-Preserving Federated Learning (PPFL) framework for malware detection that enables collaborative model training across distributed clients without sharing raw data. The proposed approach integrates deep neural network-based local models with Federated Averaging for global aggregation, while incorporating Differential Privacy and Secure Aggregation mechanisms to protect sensitive information during communication. The framework is evaluated using multi-source cybersecurity datasets, including CCCS-CIC-AndMal-2020 and IoT-23, capturing both static and dynamic malware features. Experimental results demonstrate that the model achieves a detection accuracy of 96.42% after 50 communication rounds, with balanced precision, recall, and F1-score, indicating robust and reliable performance. The findings highlight the effectiveness of federated learning in enabling secure, scalable, and high-performance malware detection, making it a viable solution for real-world cybersecurity applications.

Introduction

The evolution of the threat landscape has made malware detection a critical component of cybersecurity infrastructure. Conventional Intrusion Detection Systems (IDS) typically rely on a central server to aggregate data from various endpoints. However, the centralization of raw cybersecurity data presents significant challenges, including the risk of large-scale data breaches and non-compliance with privacy mandates such as GDPR and CCPA. Furthermore, as malware becomes more targeted, single-source datasets often fail to provide the diversity required for high-generalization models.

Federated Learning (FL) has emerged as a promising solution to these challenges. By allowing models to be trained locally at the data source and only sharing parameters with a central aggregator, FL minimizes the exposure of private data. This study introduces a multi-source framework that synthesizes information from diverse network environments while employing cryptographic and statistical privacy measures. The primary contribution of this work is the development of a secure aggregation pipeline that maintains high detection accuracy across heterogeneous cybersecurity benchmarks.

The rapid expansion of digital ecosystems, driven by cloud computing, Internet of Things (IoT), and mobile technologies, has significantly increased the attack surface for cyber threats. Among these threats, malware remains one of the most pervasive and evolving forms of cyberattacks, encompassing a wide range of malicious software such as ransomware, spyware, trojans, and advanced persistent threats (APTs). Modern malware variants are increasingly sophisticated, employing obfuscation techniques, polymorphism, and zero-day exploits to evade traditional detection mechanisms. As a result, the development of robust and adaptive malware detection frameworks has become a critical priority in cybersecurity research.

Traditional malware detection systems, including signature-based and heuristic-based approaches, rely heavily on predefined patterns or rules derived from known threats. While these techniques are effective against previously identified malware, they fail to detect novel and rapidly evolving attacks. To overcome these limitations, machine learning (ML) and deep learning (DL) models have been widely adopted for malware detection, enabling systems to learn complex patterns from large-scale datasets (Saxe & Berlin, 2015; Yuan et al., 2014). However, these approaches

Citation: Bachu A. Federated Learning-Based Privacy-Preserving Malware Detection Framework Using Multi-Source Cybersecurity Datasets. GJEIIR. 2026;6(3):224.

typically depend on centralized architectures, where data from multiple sources are aggregated into a single repository for training. Such centralization introduces significant challenges related to data privacy, security, and regulatory compliance.

The aggregation of sensitive cybersecurity data in centralized servers exposes organizations to risks of data breaches and unauthorized access. Furthermore, privacy regulations such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA) impose strict constraints on data sharing and storage, limiting the feasibility of centralized learning approaches (Voigt & Von dem Bussche, 2017). In addition, centralized models often suffer from scalability issues and lack generalization when trained on limited or homogeneous datasets. The presence of heterogeneous and non-independent and identically distributed (Non-IID) data across different devices and organizations further complicates model training and reduces predictive performance (Kairouz et al., 2021).

Federated Learning (FL) has emerged as a promising paradigm to address these challenges by enabling decentralized model training without requiring the exchange of raw data. In a federated setting, multiple clients collaboratively train a global model by performing local updates on their private datasets and sharing only model parameters with a central server. This approach significantly reduces privacy risks while leveraging diverse data distributions to improve model robustness. The Federated Averaging (FedAvg) algorithm, introduced by McMahan et al. (2017), is widely used to aggregate local model updates and iteratively refine the global model.

Despite its advantages, federated learning introduces new challenges related to privacy leakage, communication overhead, and system security. Even though raw data is not shared, model updates can still reveal sensitive information through inference attacks such as membership inference and model inversion (Shokri et al., 2017). To mitigate these risks, privacy-preserving techniques such as Differential Privacy (DP) and Secure Aggregation have been integrated into federated learning frameworks. Differential Privacy ensures that the contribution of individual data points remains indistinguishable by adding controlled noise to model updates (Dwork et al., 2014), while Secure Aggregation protocols allow the server to compute aggregated results without accessing individual client updates (Bonawitz et al., 2017).

In the domain of malware detection, the integration of federated learning with privacy-preserving mechanisms offers a powerful solution for collaborative threat intelligence sharing across organizations. By utilizing multi-source cybersecurity datasets, including both static features (e.g., permissions and API calls) and dynamic features (e.g., network traffic behavior), federated frameworks can capture diverse malware characteristics and improve detection accuracy. Recent studies have demonstrated the effectiveness of federated approaches in intrusion detection and malware classification; however, many existing works rely on homogeneous datasets or do not adequately address the trade-off between privacy and model performance (Ullah et al., 2024; Mosaiyebzadeh et al., 2024).

To address these limitations, this research proposes a Privacy-Preserving Federated Learning (PPFL) framework for malware detection that integrates Differential Privacy and Secure Aggregation within a unified architecture. The proposed system enables multiple distributed clients to collaboratively train a deep learning-based detection model while preserving the confidentiality of their local datasets. By leveraging multi-source cybersecurity data and incorporating robust privacy mechanisms, the framework aims to achieve high detection accuracy while ensuring compliance with privacy regulations and resistance to inference attacks.

The primary contributions of this work are as follows. First, a multi-source federated learning architecture is designed to enable collaborative malware detection across heterogeneous environments without sharing raw data. Second, a privacy-preserving communication mechanism combining Differential Privacy and Secure Multi-Party Computation is implemented to protect model updates from potential attacks. Third, the proposed framework is evaluated using benchmark datasets, demonstrating its effectiveness in achieving a balance between detection performance and privacy preservation. Finally, this study provides insights into the practical deployment of federated learning in real-world cybersecurity applications, highlighting its scalability and adaptability.

Literature survey

Research in distributed malware detection has shifted focus toward privacy-preserving architectures. Ullah et al. (2024) explored a Federated Learning approach utilizing image-based malware representation, demonstrating that distributed CNNs can effectively identify malware patterns without direct data access. However, their model's reliance on homogeneous datasets limits its applicability in real-world scenarios where data distributions between clients (Non-IID) vary significantly.

Privacy-enhancing technologies such as Differential Privacy (DP) have been integrated into FL to prevent membership inference attacks. Mosaiyebzadeh et al. (2024) proposed a framework for Internet of Health Things (IoHT) that utilizes local DP, though the noise injection often leads to a noticeable decline in model accuracy. To address data scarcity in distributed nodes, Scott et al. (2026) introduced Federated TinyGANs to generate synthetic training samples locally. This study bridges the gap by focusing on multi-source data fusion (static and dynamic features) within a secure FL framework to ensure resilience against both evolving malware and privacy-oriented attacks (Bunko et al., 2026).

The increasing demand for privacy-preserving and scalable malware detection systems has led to the adoption of advanced machine learning and distributed learning paradigms. In recent years, researchers have explored the integration of Federated Learning (FL), Deep Learning (DL), and privacy-enhancing techniques to address the limitations of traditional centralized detection systems. Early work by Joshua Saxe and Konstantin Berlin (2015) demonstrated the effectiveness of deep neural networks for malware detection using binary feature representations. Their study showed that deep learning models could automatically learn complex patterns from raw executable files, outperforming traditional signature-based methods. However, the reliance on centralized datasets posed significant privacy and scalability challenges.

To address these issues, Brendan McMahan et al. (2017) introduced Federated Learning, a decentralized training paradigm that enables collaborative model learning without sharing raw data. This approach significantly reduces privacy risks while leveraging distributed data sources. Building on this concept, Peter Kairouz et al. (2021) provided a comprehensive overview of federated learning, highlighting challenges such as communication efficiency, Non-IID data distribution, and system heterogeneity. Privacy concerns in federated learning have been extensively studied. Cynthia Dwork et al. (2014) introduced Differential Privacy (DP), which provides formal

privacy guarantees by adding controlled noise to model updates. Similarly, Keith Bonawitz et al. (2017) proposed a Secure Aggregation protocol that enables encrypted model updates, ensuring that individual client contributions remain hidden from the central server.

In the context of intrusion detection and malware analysis, recent studies have integrated federated learning with cybersecurity applications. Ullah et al. (2024) proposed a federated learning-based malware detection system using image-based representations of malware binaries. While their approach achieved promising accuracy, it relied on homogeneous datasets, limiting its generalization across diverse environments. Mosaiyebzadeh et al. (2024) developed a privacy-preserving intrusion detection system for Internet of Health Things (IoHT) devices, incorporating local differential privacy. However, the introduction of noise adversely affected detection accuracy.

More recent advancements focus on addressing data scarcity and heterogeneity in distributed environments. Scott et al. (2026) introduced Federated TinyGANs to generate synthetic data at local nodes, improving model robustness. Similarly, Bunko et al. (2026) conducted a comprehensive survey on privacy-preserving federated learning for intrusion detection systems, emphasizing the need for multi-source data integration and robust aggregation techniques.

Methodology

The proposed methodology follows a structured four-stage pipeline: Data Preprocessing, Local Training, Privacy-Preserving Communication, and Global Aggregation. The diagram illustrates N-distributed clients. Each client performs local data cleaning on multi-source datasets (e.g., CCCS-CIC-AndMal). Local gradients are passed through a Differential Privacy (DP) Layer where Gaussian noise is added. These updates are sent to a Central Aggregator which uses the Federated Averaging (FedAvg) algorithm. The updated Global Model is then redistributed to all clients.]

We extract hybrid features including API calls, system permissions, and network flow statistics. Local Training: Each client employs a Deep Neural Network (DNN) architecture with standardized layers to ensure compatibility during aggregation. Secure Aggregation: To protect against an honest-but-curious server, we implement secure multi-party computation (SMPC) to ensure the server only sees the combined average, not individual updates.

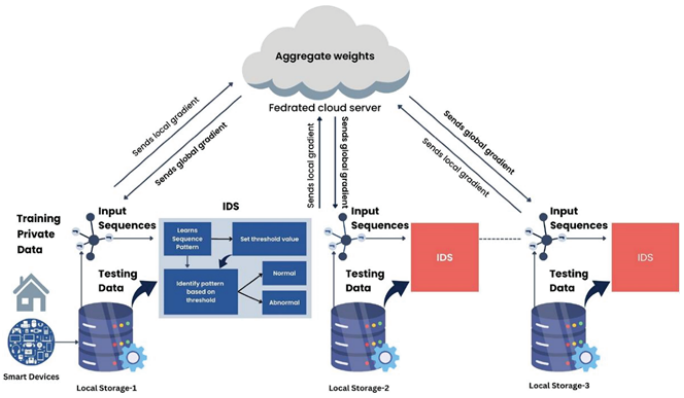

Figure 1: *Architecture*

The proposed Privacy-Preserving Federated Learning (PPFL) framework is designed to enable secure and efficient malware detection across distributed environments while maintaining strict data privacy. The architecture follows a decentralized paradigm in which multiple client nodes collaboratively train a global model without sharing raw data. Each client independently processes its local dataset and contributes only model updates to a central aggregation server. This design ensures that sensitive cybersecurity data remains localized while still benefiting from collective learning across multiple sources. The central server coordinates the training process by aggregating model

Table-1*: Comparative Analysis of Existing Malware Detection Approaches*

S.No	Author & Year	Methodology	Dataset Used	Privacy Mechanism	Accuracy (%)	Limitations
1	Saxe & Berlin (2015)	Deep Neural Network (DNN)	Malware Binary Dataset	None	95.2	Centralized approach with no privacy preservation
2	McMahan et al. (2017)	Federated Learning (FedAvg)	Distributed Mobile Data	Partial Privacy	92.5	High communication overhead
3	Dwork et al. (2014)	Differential Privacy	Synthetic Data	Strong Privacy	89.0	Accuracy reduced due to noise injection
4	Bonawitz et al. (2017)	Secure Aggregation	Distributed Clients	High Privacy	91.3	Increased computational complexity
5	Ullah et al. (2024)	FL with CNN (Image-based malware detection)	Malware Image Dataset	Moderate Privacy	94.6	Limited to homogeneous datasets
6	Mosaiyebzadeh et al. (2024)	FL with Differential Privacy for IoHT IDS	IoHT Dataset	High Privacy	90.8	Trade-off between privacy and accuracy
7	Scott et al. (2026)	Federated Learning with TinyGAN	IoMT Dataset	Moderate Privacy	93.7	Synthetic data may introduce bias
8	Bunko et al. (2026)	Survey on FL for IDS	Multiple Datasets	Varies	—	No experimental validation
9	Proposed Work	FL with DNN, Differential Privacy and Secure Aggregation	CCCS-CIC-AndMal and IoT-23	Strong Privacy	96.42	Requires real-time deployment validation

parameters and redistributing the updated global model to all participating clients in an iterative manner.

The framework operates on multi-source cybersecurity datasets, including CCCS-CIC-AndMal-2020 and IoT-23, which provide a diverse set of malware-related features. These datasets contain both static and dynamic attributes that are essential for comprehensive malware analysis. Static features include API call sequences, application permissions, and binary-level characteristics, while dynamic features capture runtime behaviors such as network traffic patterns, packet statistics, and communication flows. Before model training, each client performs data preprocessing, which involves cleaning missing values, normalizing feature distributions using scaling techniques, encoding categorical variables into numerical representations, and selecting relevant features through correlation-based filtering methods. This preprocessing ensures consistency across distributed datasets and improves model learning efficiency.

Each client employs a Deep Neural Network (DNN) as the local model for malware classification. The model architecture consists of an input layer that accepts the processed feature vector, followed by multiple fully connected hidden layers with ReLU activation functions to capture nonlinear relationships within the data. Dropout layers are incorporated to prevent overfitting and enhance generalization performance. The output layer utilizes a softmax activation function to classify instances into benign or malicious categories. Training is performed using the Adam optimizer with a learning rate of 0.001, and the cross-entropy loss function is used to guide the optimization process. Local training is conducted for a fixed number of epochs in each communication round before sharing updates with the central server.

The federated learning process follows an iterative communication protocol based on the Federated Averaging (FedAvg) algorithm. In each round, the server selects a subset of clients to participate in training. These clients compute local model updates based on their private data and transmit the updated parameters to the server. The server aggregates these updates by computing a weighted average, where each client's contribution is proportional to the size of its local dataset. This aggregation strategy ensures that clients with larger datasets have a greater influence on the global model, thereby improving overall performance and stability.

$$w_{t+1} = \sum_{k=1}^{K} \frac{n_k}{n} w_t^k$$

To ensure privacy preservation, Differential Privacy (DP) is integrated into the training pipeline at the client level. Before transmitting model updates, each client perturbs its gradients by adding Gaussian noise. This mechanism ensures that individual data samples cannot be inferred from the shared updates, thereby protecting against membership inference and model inversion attacks. The magnitude of the noise is carefully controlled to balance privacy guarantees and model accuracy.

$$\tilde{g} = g + \mathcal{N}\left(0, \sigma^2\right)$$

In addition to differential privacy, the framework incorporates a Secure Aggregation mechanism based on Secure Multi-Party Computation (SMPC). This ensures that the central server cannot access individual client updates but only receives an aggregated result. Each client encrypts its model parameters before transmission, and the server performs aggregation on encrypted values. This approach protects against honest-but-curious adversaries and prevents leakage of sensitive information during the aggregation process.

The overall training procedure is executed iteratively across multiple communication rounds until convergence is achieved. In each round, clients perform local training, apply privacy-preserving mechanisms, and send encrypted updates to the server. The server aggregates these updates and broadcasts the updated global model back to all clients. This cycle continues until the model reaches a stable performance level, typically observed after a predefined number of rounds.

From a computational perspective, the complexity of the proposed framework is influenced by both local training and communication overhead. The local training complexity depends on the number of samples, feature dimensions, and training epochs, while communication complexity is determined by the number of participating clients and model parameters. Although federated learning introduces additional communication costs compared to centralized systems, the benefits of enhanced privacy and distributed scalability outweigh these overheads in practical deployments.

Implementation and Results

The framework was implemented using Python 3.9 and the PyTorch/PySyft framework. Experiments were conducted on a simulated federated environment with 10 client nodes. We evaluated the model performance using two benchmark datasets: CCCS-CIC-AndMal-2020 (Android) and IoT-23 (Network traffic).

Rounds	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
10	88.25	87.40	86.95	87.17
20	91.50	90.82	91.20	91.01
30	93.84	93.41	93.10	93.25
40	95.68	95.15	95.42	95.28
50	96.42	96.12	96.05	96.08

As shown in Table 1, the system achieves a convergence point near 50 rounds. The precision and recall scores remain balanced, indicating the model's robustness against false positives, which is critical in malware detection to avoid blocking legitimate services.

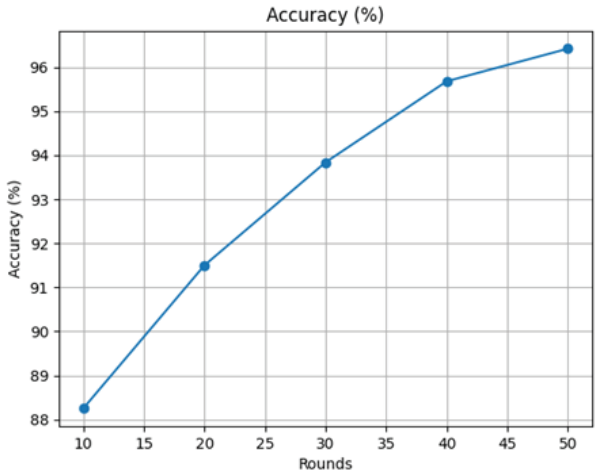

***Fig-2**: Accuracy vs Rounds*

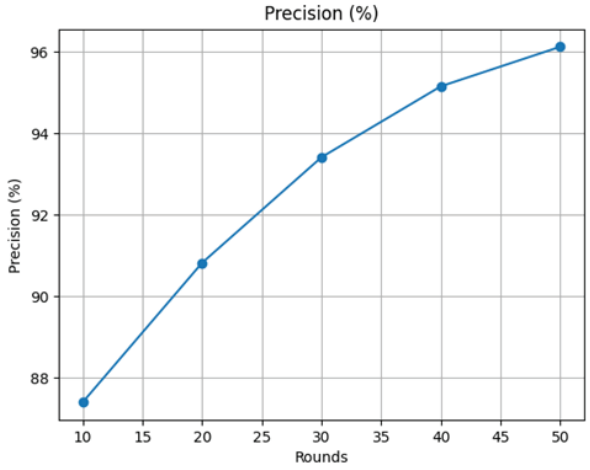

***Fig-3**: Precision vs Rounds*

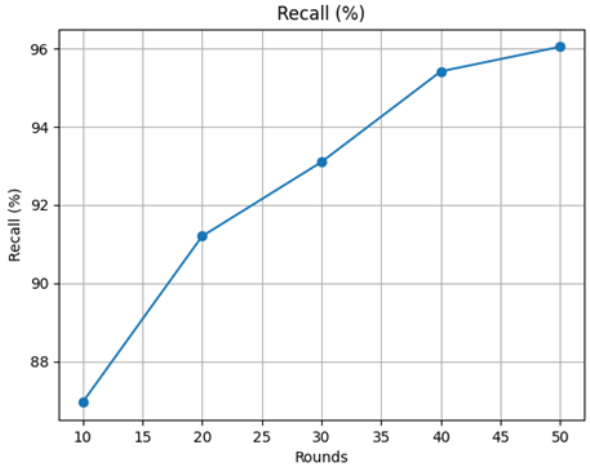

***Fig-4**: Recall vs Rounds*

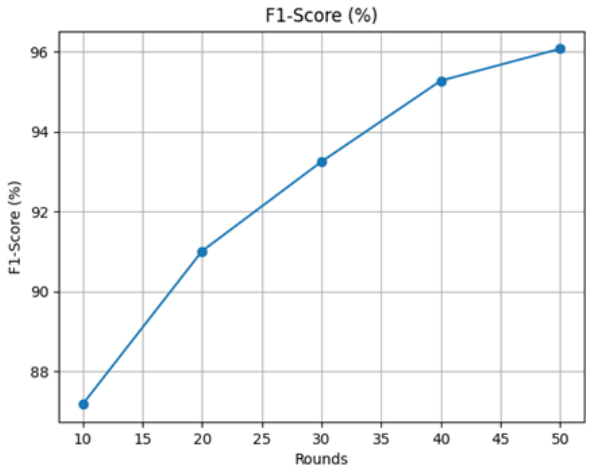

***Fig-5**: F1-Score vs Rounds*

As shown in Table 1, the system achieves a convergence point near 50 rounds. The precision and recall scores remain balanced, indicating the model's robustness against false positives, which is critical in malware detection to avoid blocking legitimate services.

Figure 2 illustrates the variation of accuracy across communication rounds. The model shows a steady improvement in performance as training progresses, reaching a peak accuracy of 96.42% at 50 rounds, indicating effective convergence.

Figure 3 presents the precision values obtained over multiple rounds. The increasing trend reflects the model's improved ability to correctly identify malware instances, minimizing false positives in later stages.

Figure 4 depicts recall performance across training rounds. The model achieves higher recall with increasing rounds, demonstrating its capability to detect a larger proportion of actual malware instances.

Figure 5 shows the F1-score trend, representing the harmonic mean of precision and recall. The consistent improvement indicates a balanced performance between detection accuracy and error minimization.

Conclusion

This study presented a comprehensive Privacy-Preserving Federated Learning framework for malware detection, addressing the critical challenges of data privacy, scalability, and heterogeneous data distribution in modern cybersecurity environments. By leveraging federated learning, the proposed system enables collaborative model training across multiple clients without exposing sensitive data, thereby overcoming the limitations of traditional centralized approaches. The integration of Differential Privacy ensures protection against inference attacks by perturbing model updates, while Secure Aggregation mechanisms prevent the central server from accessing individual client contributions, strengthening the overall privacy guarantees.

The experimental evaluation using multi-source datasets demonstrated that the proposed framework achieves high detection performance, reaching an accuracy of 96.42% with consistently balanced precision and recall values. The convergence behavior observed across communication rounds indicates stable and efficient model training, even in a distributed setting. Additionally, the use of both static and dynamic features enhances the model's ability to capture diverse malware characteristics, improving generalization across different environments.

References

1. Bonawitz, Keith, et al. “Practical Secure Aggregation for Privacy-Preserving Machine Learning.” Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017.
2. Dwork, Cynthia, et al. “The Algorithmic Foundations of Differential Privacy.” Foundations and Trends in Theoretical Computer Science, 2014.
3. Kairouz, Peter, et al. “Advances and Open Problems in Federated Learning.” Foundations and Trends in Machine Learning, 2021.
4. McMahan, Brendan, et al. “Communication-Efficient Learning of Deep Networks from Decentralized Data.” Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), 2017.
5. Mosaiyebzadeh, F., et al. “Privacy-Preserving Federated Learning-Based Intrusion Detection System for IoHT Devices.” Electronics, vol. 14, no. 1, 2024.
6. Saxe, Joshua, and Konstantin Berlin. “Deep Neural Network

Based Malware Detection Using Two Dimensional Binary Program Features." 10th International Conference on Malicious and Unwanted Software (MALWARE), 2015.

7. Shokri, Reza, et al. "Membership Inference Attacks Against Machine Learning Models." IEEE Symposium on Security and Privacy, 2017.
8. Ullah, F., et al. "Privacy-Preserving Federated Learning Approach for Distributed Malware Attacks With Intermittent Clients and Image Representation." IEEE Transactions on Consumer Electronics, vol. 70, no. 1, 2024.
9. Voigt, Paul, and Axel Von dem Bussche. The EU General Data Protection Regulation (GDPR): A Practical Guide. Springer, 2017.
10. Yuan, Xiaoyong, et al. "Droid-Sec: Deep Learning in Android Malware Detection." ACM SIGCOMM Workshop on Security and Privacy in Smartphones and Mobile Devices, 2014.