International Conference on Science, Engineering & Management Trends

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Modelling Client Heterogeneity for Federated Intrusion Detection in IoT: A Dirichlet-Based Non-IID Evaluation Protocol

Sujata Pattnaik*[1], Dr. Ridhika Sidhi*[2]

*[1]Research Scholar, Department of Computer Science, William Carey University, Meghalaya, India

*[2]Phd. Guide, Department of Computer Science, William Carey University, Meghalaya, India

Abstract- Federated learning (FL) lets Internet of Things (IoT) devices, gateways and edge nodes train a shared intrusion detection model without moving raw traffic to a central server. In practice, however, IoT clients hold very different traffic: a smart home, a factory and a hospital differ in data volume, attack mix, resources and availability. Many federated intrusion detection studies evaluate under uniform random splits, which hides how the global model behaves under such heterogeneity. This paper proposes an evaluation protocol for federated intrusion detection that makes client heterogeneity an explicit, controlled variable. Label skew is generated with a Dirichlet distribution, quantity skew with a log-normal size model, and availability through participation and dropout rates. A simulation of the partitioning procedure shows that the concentration parameter α spans the full range from near-IID clients (mean Jensen–Shannon divergence ≈ 0.001 bits at α = 100) to clients dominated by one or two classes (≈ 0.39 bits and 1.7 effective classes at α = 0.1 with ten clients). The protocol pairs global macro-F1 with client-level measures—worst-client F1 and dispersion—and compares a centralised baseline, FedAvg and FedProx under identical splits, providing a reproducible basis for judging federated IDS under realistic IoT conditions.

Keywords- Internet of Things, federated learning, intrusion detection, non-IID data, Dirichlet partitioning, client heterogeneity, evaluation protocol

I. INTRODUCTION

IoT networks are attractive targets because devices are numerous, weakly protected and continuously connected [1]. Learning-based intrusion detection systems (IDS) can detect such attacks [2], and federated learning (FL) allows them to be trained collaboratively while raw traffic remains with each participant [3], [4]. FL-based IDS for IoT have shown promising results [5], [6].

A key difference between FL and centralised learning is that each client sees only part of the data distribution. IoT clients are naturally heterogeneous: their traffic volume, attack exposure, computing resources and availability differ. Federated averaging (FedAvg) can lose accuracy when client label distributions are skewed [7], and federated IDS evaluations confirm that non-IID partitions change outcomes significantly [8]. Yet many studies still split benchmark datasets uniformly at random, report only a global average and give too little detail on the partition to reproduce [9], [10]. An average can hide a client whose detection is poor—precisely the site an attacker would target.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

This paper addresses that gap with a heterogeneity-aware evaluation protocol for federated IDS in IoT. Its contributions are: (1) a variable model linking heterogeneity factors to measurable outcomes; (2) a partitioning procedure combining Dirichlet label skew, log-normal quantity skew, partial participation and dropout; (3) a simulation that quantifies how the Dirichlet parameter α controls divergence between clients; and (4) a reporting scheme that pairs global metrics with client-level metrics and fixed comparison configurations.

II. BACKGROUND AND RELATED WORK

1. Federated Averaging

With K clients holding n_k records each, FedAvg updates the global model in round t by averaging locally trained models of the participating set S_t [3]:

$$W^{t+1} = \Sigma_{k \in S_t} (n_k / n_S) \, w_k^{t+1} \qquad (1)$$

Under heterogeneous data, several local epochs cause local models to drift toward their own optima. FedProx limits this drift by adding a proximal term to each local objective [11]:

$$\mathrm{Min}_w \, F_k(w) + (\mu / 2) \, \|w - w^t\|^2 \qquad (2)$$

Other remedies include control variates [12]. Their behaviour on intrusion data, in which some attack classes appear at only a few clients, has received less attention [8].

2. Non-IID Partitioning in Federated IDS

Dirichlet label partitioning is a widely used way to generate controllable non-IID splits [13]. Federated IDS work has used shard-based or site-based splits [8], [14], and IoT datasets such as TON_IoT and Edge-IIoTset provide heterogeneous sources [15], [16]. Surveys nonetheless identify inconsistent heterogeneity settings as a barrier to comparing federated IDS results [10], [17].

III. HETEROGENEITY VARIABLE MODEL

Fig. 1 summarises the variables. Contextual factors—privacy level, client heterogeneity, communication and participation, and data characteristics—act on the federated IDS pipeline and produce four groups of measurable outcomes: detection performance, privacy protection, cost and resource usage, and robustness. This paper isolates the heterogeneity and participation factors while holding privacy settings fixed, so that their effect on detection can be attributed cleanly.

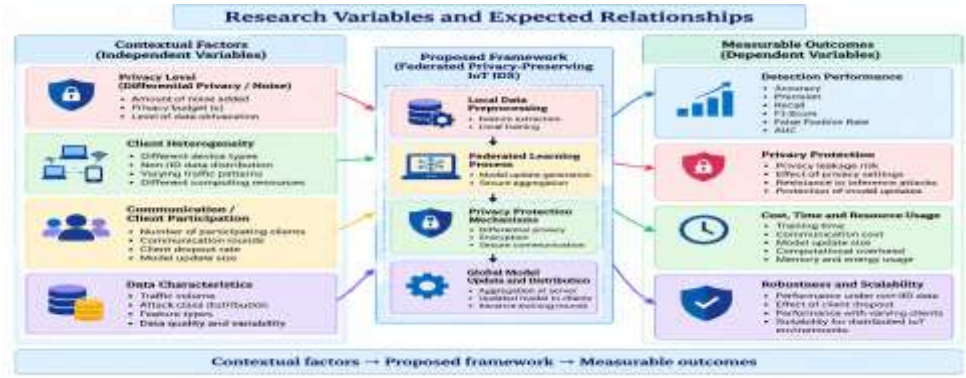


Fig. 1. Research variables and expected relationships. Contextual (independent) factors act on the federated privacy-preserving IDS to produce measurable (dependent) outcomes. This paper varies client heterogeneity and participation.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Fig. 2 makes the heterogeneity concrete. Clients representing a smart home, an industrial IoT site and a healthcare edge environment differ in data volume, data type, computational resources, network condition and attack characteristics, but contribute to one global model.

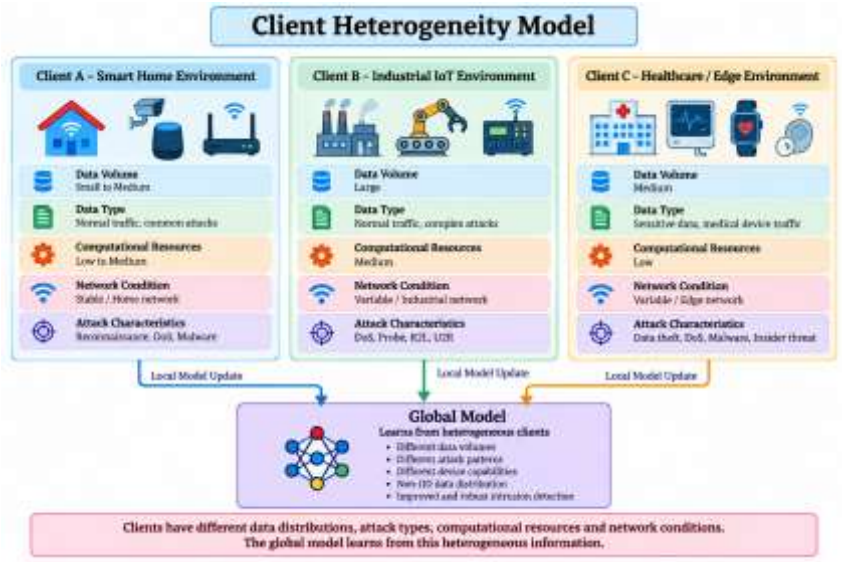


Fig. 2. Client heterogeneity model: smart-home, industrial IoT and healthcare/edge clients differ in data volume, data type, resources, network condition and attack characteristics while training one global model.

IV. PROPOSED PARTITIONING PROTOCOL

1. Label Skew

Let M be the number of traffic classes (normal plus attack categories). For each client k, a class-proportion vector is drawn from a symmetric Dirichlet distribution, and records of each class are allocated to clients in proportion to these vectors [13]:

$$P_k = (p_{k,1}, \ldots, p_{k,M}) \sim \mathrm{Dir}(\alpha, \ldots, \alpha) \quad (3)$$

Large α yields near-identical class mixes; small α concentrates each client on a few classes. When a class has fewer records than required, the allocation is renormalised, and the realised class counts are recorded.

2. Quantity Skew, Participation and Dropout

Client sizes are drawn as $nk \propto exp(zk)$with $z_k \sim N(0, s^2)$, where $s = 0$ gives equal sizes. In each round a fraction q of clients is sampled. Independently, each sampled client drops out before returning its update with probability d, which models intermittent IoT connectivity. Table I lists the resulting scenarios.

Table 2: Heterogeneity Scenarios and Parameter Ranges

ID	Condition	Parameter varied	Values
S1	IID reference	None	$\alpha = 100$, $s = 0$, $q = 1$, $d = 0$
S2	Label skew	Dirichlet α	{1, 0.5, 0.1}
S3	Label and quantity skew	α and s	$\alpha \in \{1, 0.1\}$; $s \in \{0.5, 1.0\}$
S4	Partial participation	Sampling rate q	{0.2, 0.5, 1.0}
S5	Client dropout	Dropout probability d	{0.1, 0.3}

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

3. Quantifying the Degree of Heterogeneity

To report heterogeneity in an interpretable way, the protocol computes the mean Jensen–Shannon divergence (JSD) between each client's class distribution and the global distribution $\bar{p}$, together with the number of classes needed to cover 90% of each client's records:

$$H(\alpha) = (1/K) \Sigma_{k=1}^{K} JSD(p_k \| \bar{p}) \qquad (4)$$

Fig. 3 shows these quantities for the partitioning procedure itself, averaged over 30 seeds for five classes. At $\alpha = 100$ the mean divergence is about 0.001 bits and every client covers all five classes; at $\alpha = 1$ it is about 0.12 bits; at $\alpha = 0.1$ it rises to about 0.39 bits ($K = 10$) and clients cover only about 1.7 classes on average. The steepest change occurs between $\alpha = 0.1$ and $\alpha = 1$, which motivates the values chosen in Table I. The figure characterises the data splits only—it contains no detection results.

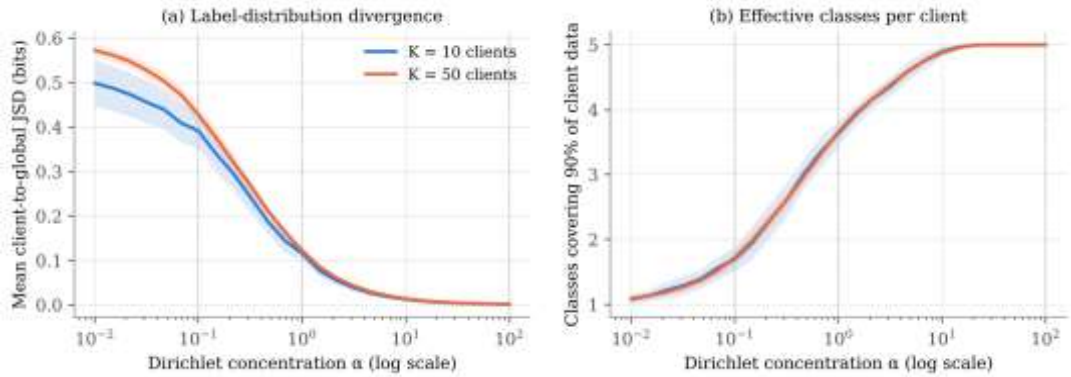


Fig. 3. Heterogeneity produced by Dirichlet partitioning (five classes, 30 seeds; shaded band = ±1 standard deviation). (a) Mean Jensen–Shannon divergence between client and global label distributions. (b) Mean number of classes covering 90% of a client's records. Dotted lines mark α = 0.1, 1 and 100.

V. EVALUATION DESIGN

1. Configurations and Data

Three configurations are trained with the same model, features and held-out test set: a centralised baseline on pooled training data, FedAvg and FedProx. A compact multilayer perceptron over flow features is used so that the effect of heterogeneity is not confounded by model capacity. CICIoT2023 serves as the primary dataset because of its many devices and attack types [18], and TON_IoT as a secondary dataset [15]. Device identifiers, IP addresses and timestamps are removed before partitioning.

2. Global and Client-Level Metrics

Because intrusion data is imbalanced, macro-F1 and per-class recall are primary, and the false-positive rate is reported to reflect alert burden [19]. In addition, the global model is evaluated on each client's local test split to obtain client scores F_k, summarised as

$$F_{min} = \min_k F_k, \quad \Delta_F = \max_k F_k - \min_k F_k \qquad (5)$$

Worst-client F1 captures the weakest protected site, and the spread captures fairness across clients. Table II summarises the full reporting scheme. Each setting is repeated with five seeds; paired differences are tested with the Wilcoxon signed-rank test.

International Journal of Science, Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

Table 2: Reporting Scheme for Each Scenario

Measure	Definition	Purpose
Global macro-F1, FPR	On the common held-out test set	Overall detection quality and alert burden
Per-class recall	Recall of each attack category	Exposes rare-class loss under skew
Worst-client F1, spread	Eq. (5) on client test splits	Weakest site and fairness
Rounds to target F1	First round reaching 95% of final macro-F1	Convergence cost of heterogeneity
Partition record	α, s, q, d, seed, class counts per client	Reproducibility

VI. DISCUSSION

Three expectations follow from the literature and are stated as hypotheses to be tested. First, as α falls below 1, global macro-F1 should decline and worst-client F1 should decline faster, because clients that lack a class contribute no gradient for it [7]. Second, FedProx should narrow the spread at low α with little cost at high α [11]. Third, partial participation and dropout should mainly slow convergence, but at low α they can also remove the only clients holding a rare attack class in a given round, causing oscillation. Reporting the partition record makes these effects attributable rather than anecdotal. The protocol uses logical partitions of public datasets, which approximate but do not reproduce real deployments; adding site-based partitions where datasets provide source metadata is a natural extension.

VII. CONCLUSION

Client heterogeneity is a defining property of IoT and a principal source of risk for federated intrusion detection, but it is often treated informally. This paper presented a protocol that controls label skew, quantity skew, participation and dropout explicitly, quantifies the resulting heterogeneity, and reports client-level as well as global detection metrics under fixed comparison configurations. The simulation shows that the Dirichlet parameter provides a smooth, interpretable control from near-IID to highly skewed clients. Applying the protocol to centralised, FedAvg and FedProx training on IoT intrusion datasets is the next step of this research.

REFERENCES

1. C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017, doi: 10.1109/MC.2017.201.
2. N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, "Network intrusion detection for IoT security based on learning techniques," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2671–2701, 2019, doi: 10.1109/COMST.2019.2896380.
3. H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist. (AISTATS)*, Fort Lauderdale, FL, USA, 2017, pp. 1273–1282, doi: 10.48550/arXiv.1602.05629.
4. D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, "Federated learning for Internet of Things: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1622–1658, 2021, doi: 10.1109/COMST.2021.3075439.
5. T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi, "DÏoT: A federated self-learning anomaly detection system for IoT," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Dallas, TX, USA, 2019, pp. 756–767, doi: 10.1109/ICDCS.2019.00080.

6. S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of Things intrusion detection: Centralized, on-device, or federated learning?," *IEEE Netw.*, vol. 34, no. 6, pp. 310–317, 2020, doi: 10.1109/MNET.011.2000286.
7. Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv:1806.00582*, doi: 10.48550/arXiv.1806.00582.
8. E. M. Campos *et al.*, "Evaluating federated learning for intrusion detection in Internet of Things: Review and challenges," *Comput. Netw.*, vol. 203, Art. no. 108661, 2022, doi: 10.1016/j.comnet.2021.108661.
9. S. Agrawal *et al.*, "Federated learning for intrusion detection system: Concepts, challenges and future directions," *Comput. Commun.*, vol. 195, pp. 346–361, 2022, doi: 10.1016/j.comcom.2022.09.012.
10. A. Belenguer, J. A. Pascual, and J. Navaridas, "A review of federated learning applications in intrusion detection systems," *Comput. Netw.*, vol. 258, Art. no. 111023, 2025, doi: 10.1016/j.comnet.2024.111023.
11. T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst. (MLSys)*, vol. 2, 2020, pp. 429–450, doi: 10.48550/arXiv.1812.06127.
12. S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. 37th Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 5132–5143, doi: 10.48550/arXiv.1910.06378.
13. T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," 2019, *arXiv:1909.06335*, doi: 10.48550/arXiv.1909.06335.
14. S. I. Popoola, R. Ande, B. Adebisi, G. Gui, M. Hammoudeh, and O. Jogunola, "Federated deep learning for zero-day botnet attack detection in IoT-edge devices," *IEEE Internet Things J.*, vol. 9, no. 5, pp. 3930–3944, 2022, doi: 10.1109/JIOT.2021.3100755.
15. N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustain. Cities Soc.*, vol. 72, Art. no. 102994, 2021, doi: 10.1016/j.scs.2021.102994.
16. M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40281–40306, 2022, doi: 10.1109/ACCESS.2022.3165809.
17. A. Vyas, P.-C. Lin, R.-H. Hwang, and M. Tripathi, "Privacy-preserving federated learning for intrusion detection in IoT environments: A survey," *IEEE Access*, vol. 12, pp. 127018–127050, 2024, doi: 10.1109/ACCESS.2024.3454211.
18. E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, Art. no. 5941, 2023, doi: 10.3390/s23135941.
19. J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn. (ICML)*, Pittsburgh, PA, USA, 2006, pp. 233–240, doi: 10.1145/1143844.1143874.