POLITECNICO DI TORINO
Repository ISTITUZIONALE

SCARF: Secure and Collaborative Reinforcement Learning-based Routing

Original
SCARF: Secure and Collaborative Reinforcement Learning-based Routing / Monaco, D., Marchetto, G., Sacco, A.. - (In corso di stampa). (IEEE Conference on Network Function Virtualization and Software-Defined Networking (NFV-SDN) Tokyo, Japan 2 - 4 November 2026).

Availability:
This version is available at: 11583/3016143 since: 2026-09-30T09:34:14Z

Publisher:
IEEE

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©9999 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

10 November 2026

SCARF: Secure and Collaborative Reinforcement Learning-based Routing

Doriana Monaco * Guido Marchetto * Alessio Sacco *
* *Department of Control and Computer Engineering*, Politecnico di Torino, Italy

***Abstract*—Modern communication networks require highly adaptive routing mechanisms to handle traffic growth and stringent Quality of Service demands. While traditional protocols struggle in these dynamic environments, Reinforcement Learning (RL) and Multi-Agent RL (MARL) have emerged as robust alternatives. However, deploying learning-based solutions across multi-domain networks is severely limited by strict data privacy constraints, as competing Internet Service Providers (ISPs) refuse to share sensitive topology or traffic data. To address this critical gap, we propose SCARF, a framework that integrates RL into Federated Learning (FL) and also incorporates Differential Privacy (DP) and Secure Aggregation (SA) to defend against attacks from untrusted entities. SCARF enables multiple network domains to collaboratively train generalized routing policies while providing privacy guarantees. Experimental evaluations demonstrate that our collaborative approach outperforms traditional RL by an average of 8% in meeting traffic demands and achieves zero-shot transferability across unseen topologies of varying scales without domain-specific retraining. Furthermore, despite the minor performance degradation and increased training time introduced by DP and SA, the fully secured SCARF framework maintains a highly competitive 88% average demand satisfaction, successfully balancing multi-domain collaboration with privacy.**

***Index Terms*—Federated Learning, Network Routing, Privacy**

I. Introduction

Modern communication networks have evolved rapidly, driven by the explosive growth of interconnected devices, massive traffic volumes, and stringent Quality of Service (QoS) demands. In these dynamic and complex environments, traditional routing protocols like Open Shortest Path First (OSPF) face significant challenges. These conventional protocols, designed for stable networks, struggle to perform effectively under dynamic environments [1]–[3].

Machine Learning (ML), particularly Reinforcement Learning (RL), has emerged as a powerful tool to create more flexible and robust network management systems. RL agents can automatically learn optimal routing policies from environmental feedback [4]–[8]. However, traditional RL approaches face critical limitations when deployed in large-scale networks, as they typically rely on centralized architectures that require heavy communication overhead and present a single point of failure. To address the scalability limits of centralized systems, several studies distribute decision-making to local nodes through Multi-Agent Reinforcement Learning (MARL), significantly improving network adaptability [9]–[12]. While RL and MARL are widely used to optimize dynamic traffic engineering within individual network domains, an Internet Service Provider's (ISP) routing policy is inherently shaped by the traffic patterns and topological characteristics of its local network. At the same time, ISPs are often reluctant to share sensitive operational information, such as network topologies, because of privacy and confidentiality concerns. Federated Reinforcement Learning (FRL) addresses this limitation by enabling multiple domains to collaboratively train routing policies by exchanging model updates rather than raw network data. However, shared updates may still leak sensitive information, as an untrusted server or adversary can potentially infer or reconstruct portions of the underlying private data. Therefore, standard FRL alone does not provide sufficient guarantees for strict operational privacy [13].

To overcome these challenges, we propose a **S**ecure and **C**oll**A**borative **R**ein**F**orcement learning-based routing (**SCARF**). Our solution integrates FRL with Differential Privacy (DP) and Secure Aggregation (SA), enabling multiple network domains to collaboratively train a highly generalizable global model that optimizes routing decisions while ensuring privacy protections. In our approach, each domain locally deploys a MARL instance to split local traffic demands across preconfigured paths. To build a unified global policy without exposing sensitive operational data, domains periodically exchange model weights with a central server. DP protects data characteristics by injecting noise into local updates, while SA cryptographically masks them, allowing the server to recover only their aggregate. Ultimately, SCARF yields a global policy that operators can deploy independently within their own domain, providing superior adaptability to dynamic network conditions and heterogeneous topologies.

We evaluate SCARF across multiple topologies and first demonstrate that our collaborative setting outperforms individual RL models on the same tasks seen during training, while achieving comparable performance on unseen scenarios compared with specialized models. Next, we evaluate the impact of the privacy mechanisms: while the noise introduced by DP slightly drops routing performance, and SA cryptographic primitives require additional training time, SCARF maintains 85% average demand satisfaction. Together, these mechanisms deliver a secure model that effectively supports multi-domain collaboration.

The remainder of the paper is organized as follows. Section II reviews related FRL routing and privacy-preserving FL approaches, while Section III presents the system overview. Sections IV and V discuss FRL vulnerabilities and the security mechanisms of SCARF, respectively. Section VI evaluates performance, and Section VII concludes the paper.

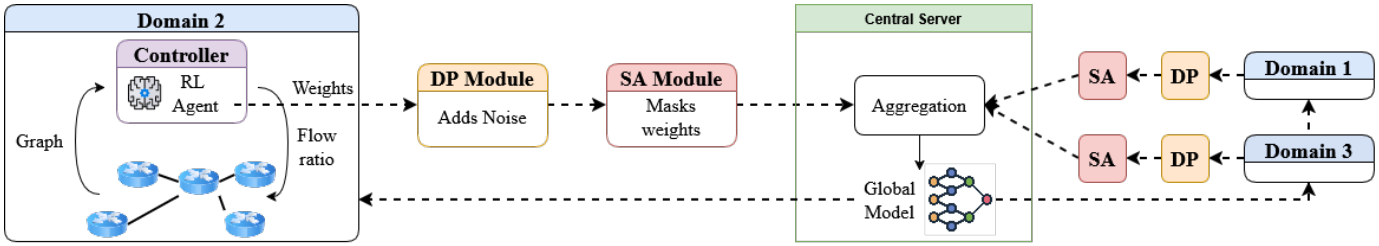


Fig. 1: Overall SCARF learning process. ISPs can collaborate for network management while keeping their data private.

II. Related Work

In this section, we review existing works that adopt RL and FRL for routing and privacy-preserving mechanisms to secure federated frameworks.

FRL routing. RL has been widely adopted for adaptive routing and traffic engineering due to its ability to optimize network performance under dynamic traffic conditions [4], [5]. To address the scalability limitations of centralized RL approaches, several studies have explored MARL, where routing decisions are distributed among multiple agents [11], [12]. For instance, TEAL [9] integrates FlowGNN-based representation learning with a shared MARL policy and ADMM refinement to compute near-optimal traffic allocations efficiently. At the same time, MATE [10] introduces a hierarchical decomposition that enables distributed policy computation over abstracted network representations.

The need for distributed learning becomes even more pronounced in multi-domain networks, where administrative entities, such as Internet Service Providers, are often unwilling to expose their internal topologies, traffic statistics, or routing policies. More recently, FRL has emerged as a promising paradigm for collaborative network optimization. By exchanging model updates rather than raw network data, FRL enables geographically distributed agents to learn routing policies while jointly preserving data privacy. Existing FRL-based routing solutions have primarily focused on wireless, IoT, and UAV networks. For example, FDRL [14] leverages federated training to improve routing robustness in dynamic wireless environments, while FEDRDR [15] reduces synchronization overhead through knowledge distillation in resource-constrained UAV networks. While these studies demonstrate the potential of FRL to support collaborative learning, its application to multi-domain collaboration remains largely unexplored. SCARF enables multiple ISP domains to collaboratively optimize routing policies while preserving operational privacy, limiting communication overhead, and adapting to heterogeneous network conditions.

Privacy-preserving FL. Research has shown that an attacker or untrusted central server may reverse-engineer shared model updates to reconstruct a client's private training data [16]. To mitigate such privacy risks, prior work has investigated Secure Aggregation (SA) and Differential Privacy (DP). For example, SparseSecAgg [17] combines SA with sparsification mechanisms to support privacy-preserving aggregation of model updates, whereas FLDP and GDPFed [18], [19] integrate DP and SA to provide stronger end-to-end privacy guarantees in federated learning systems. Complementing these approaches, theoretical studies have also examined communication-efficient aggregation frameworks to enable scalable privacy-preserving learning [20].

More recently, DP has also been incorporated into Federated Reinforcement Learning to protect privacy in distributed decision-making systems. For example, CFRL [21] applies local Differential Privacy through gradient clipping and Gaussian noise injection at edge devices before sharing policy updates, enabling privacy-preserving policy learning in IoT resource allocation scenarios. These works demonstrate that DP and SA are effective mechanisms for securing Federated Learning and FRL systems. Motivated by these advances, this work adopts DP and SA as key privacy-preserving components within SCARF, enabling collaborative routing across distributed domains while maintaining formal privacy guarantees.

III. Control Loop and System Components

We here describe how multiple network domains can collaborate to train a global TE model. Each network domain is managed independently by a network operator, who may be unwilling to share sensitive information such as traffic matrices, routing policies, or network topology. However, collaborative training enables learning more generalizable policies across heterogeneous environments. To achieve this, we propose a secure FRL approach.

As shown in Fig. 1, each domain locally trains a Reinforcement Learning agent, deployed atop its SDN controller, to split traffic demands across multiple paths based on local network conditions. When a new traffic demand matrix arrives, the neural architecture FlowGNN [9] extracts feature representations that are then passed to a shared policy network. Multiple COMA agents operate on these representations, where each agent is responsible for a specific flow demand under a shared global objective. Ultimately, the MARL algorithm determines how to split traffic demand into multiple flows across different preconfigured paths.

Periodically, the local model weights are securely exchanged with other parties through a federated approach: a central server receives all domain updates, aggregates them, and then redistributes the global model parameters. At the end of the training, all domains share a single policy trained on various topologies and traffic patterns, with a single objective: to allocate the most traffic.

To secure the federated updates against adversarial inference and interception, our framework integrates Differential

Privacy and Secure Aggregation. We detail the mathematical mechanics of these defenses in Section V. At deployment time, each network operator deploys the shared model locally on its own domain, benefiting from more general capabilities that enable it to adapt to network changes without additional communication.

IV. Security concerns of FRL

In centralized Machine Learning, data is collected in a single location for training, which increases the risk of data breaches and unauthorized access. Federated Learning offers a solution by distributing the training process across multiple clients, ensuring that raw data remains local to each client while only model updates are communicated to a central server. However, this architecture does not completely eliminate the risk of information leakage, with the risk of compromising sensitive data such as source-destination traffic patterns or topology graphs.

A. Threat Model

We consider an FRL setting in which a central server coordinates multiple network domains. The server follows an honest-but-curious threat model: it faithfully executes the prescribed training protocol, including model aggregation and communication, but may attempt to infer sensitive information from client updates. Accordingly, the server can observe and analyze all messages exchanged throughout the training process, including information collected across multiple communication rounds. However, it does not alter client updates, deviate from the protocol, or collude with participating clients. We also consider an external adversary that does not participate directly in the training process but may gain access to the communication channel. Thus, depending on the threat scenario, the adversary may be either the honest-but-curious central server or an external entity. In both cases, the adversarial objective is to infer private information about client data, for example, by reconstructing training samples or performing membership inference attacks. All participating client domains are assumed to be trustworthy and to comply with the prescribed protocol.

B. Membership Inference Attacks (MIAs)

This attack aims to determine whether a specific data sample was included in the training dataset [22] by analyzing the model's outputs or parameters. In FL settings, membership inference attacks can be categorized into two main variants, depending on the adversary's objective and level of access.

Direct Server MIA. In this setting, the adversary seeks to infer whether a particular data sample belongs to the local dataset of a specific client. This attack leverages the server's visibility into individual client updates during training. By analyzing parameter updates associated with a given client across multiple communication rounds, the adversary can attempt to link learned patterns to that client's private data. As a result, the attack enables fine-grained attribution of membership at the client level.

Global MIA. In contrast, global membership inference operates on the global model. It aims to determine whether a given data sample was used by any client (record-level) or whether a client participated in a federated round (client-level). Typically, the adversary queries the trained model with either specific candidate data points or a set of inputs representative of a client's data distribution, and analyzes the resulting outputs, such as action selection consistency or temporal-difference (TD) error. Based on these signals, it decides whether a specific sample was likely encountered during training or the client participated in the round. This form of attack exploits the tendency of machine learning models to behave differently on seen versus unseen data, thereby revealing potential membership information.

C. Model Inversion Attacks (MIs)

Model inversion attacks aim to reconstruct information about the training experience from access to a trained model's parameters or outputs [23]. It doesn't require prior access to candidate samples, instead, it aims to generate inputs that reveal information about the training experience. The underlying intuition is that the learned policy or value function encodes information about the states and interactions observed during training. By probing this learned mapping, an adversary can generate inputs that are consistent with the model's internal representation of the training experience.

The adversary can exploit access to the local client updates to perform such attacks. The adversary iteratively refines candidate inputs to produce characteristic responses from the model, such as highly consistent action preferences or stable value estimates. The resulting inputs can reveal information about the state inputs that were encountered during training. Two variants of model inversion attacks can be distinguished.

Instance-level model inversion. In this setting, the adversary attempts to reconstruct specific states observed during training. The attack typically starts from arbitrary inputs and progressively modifies them so that the model reacts to them in a way that is indicative of previously seen experience.

Class-level model inversion. In this variant, the adversary aims to reconstruct representative or prototypical states that capture common patterns in the training experience, rather than recovering specific instances [24]. This can expose aggregate information about the environment dynamics or shared features across clients' data, even without recovering exact inputs.

D. Eavesdropping

Eavesdropping is a passive attack in which the adversary observes the communication between clients and the server without altering it [25]. By intercepting model updates exchanged during training, an external attacker may be able to reconstruct or approximate the global or local models. This information can then be exploited to infer sensitive data through attacks such as model inversion or membership inference.

V. Privacy-Preserving SCARF

In this section, we formalize the mathematical execution of SCARF (Alg. 1) and its security properties. The protocol operates over R global communication rounds. During each round, domains execute local training to compute unconstrained updates Δ^j. Rather than sharing these raw updates, the framework enforces a two-stage privacy guarantee. First, DP bounds the sensitivity of the update via a clipping norm C and masks it with Gaussian noise σ (Lines 12-15). Second, SA wraps the noisy parameters $\hat{\theta}^j$ in a cryptographic mask before transmission (Lines 17-18). This ensures the server only ever computes the secure weighted sum θ^g (Lines 22-23), fulfilling the dual requirements of data privacy and model convergence.

The combination of Differential Privacy and Secure Aggregation provides complementary protection against multiple attack types.

Algorithm 1 SCARF with DP and SA

Require: Domains $j \in \{1, \ldots, N\}$, Server, Global rounds R, Local steps T
Require: Learning rate α^j, Clipping norm C, Privacy params (ϵ, δ)
1: Server initializes global policy θ^g
2: **for** round $r = 1$ **to** R **do**
3: Server broadcasts θ^g to all N domains
4: **for** each domain j **in parallel do**
5: $\theta^j \leftarrow \theta^g$ {Initialize local weights}
6: **for** step $t = 1$ **to** T **do**
7: Execute action a^j, observe reward r^j and next states
8: Compute COMA advantage A_i^j for all agents i
9: Compute gradient g^j w.r.t. policy π_{θ^j}
10: $\theta^j \leftarrow \theta^j + \alpha^j g^j$ {Unconstrained local update}
11: **end for**
12: $\Delta^j \leftarrow \theta^j - \theta^g$ {Compute model update}
13: $\overline{\Delta}^j \leftarrow \Delta^j / \max(1, ||\Delta^j||_2 / C)$ {Clip update}
14: $\sigma \leftarrow C\sqrt{2\log(1.25/\delta)}/\epsilon$
15: $\tilde{\Delta}^j \leftarrow \overline{\Delta}^j + \mathcal{N}(0, \sigma^2 I)$ {Add DP noise}
16: $\hat{\theta}^j \leftarrow \theta^g + \tilde{\Delta}^j$ {Reconstruct noisy weights}
17: Generate cryptographic mask^j
18: Send $(\hat{\theta}^j + \text{mask}^j)$ to Server
19: **end for**
20: **Server Aggregation:**
21: Receive masked updates and compute secure sum:
22: $\sum_{j=1}^{N}(\hat{\theta}^j + \text{mask}^j) = \sum_{j=1}^{N} \hat{\theta}^j$ {Masks cancel out}
23: $\theta^g \leftarrow \sum_{j=1}^{N} \frac{n^j}{n} \hat{\theta}^j$ {Weighted global update}
24: **end for**

A. Differential Privacy

DP [26] aims at protecting the privacy of data contributions during model training. It operates by introducing controlled randomness into the data to obscure its influence on the output model, while preserving the overall statistical utility. The DP mechanism in SCARF operates at the client level, where the local model update for each network domain constitutes the privacy unit and is perturbed with calibrated noise before being transmitted to the central server. The noise is carefully calibrated by the privacy budget ϵ, which quantifies the trade-off between privacy and data utility, and δ, which represents the probability of privacy violation. By injecting noise, we can ensure that the presence or absence of each client's training set has minimal impact on the final model. This way, an observer cannot link the model's updates to the specific operational profile of any particular domain, thereby formally protecting against any global MIA.

It's important to note that our framework is vulnerable to record-level membership attacks against the global model, since client-level DP only bounds whether a given client participated with some data.

B. Secure Aggregation

SA [27] is a cryptographic protocol that enables the secure aggregation of model updates in distributed learning settings. Its main goal is to ensure that a central server can compute only the aggregated result of client contributions, without gaining access to any individual update. This is achieved through multi-party computation techniques that rely on cryptographic primitives to mask locally trained model parameters, typically represented as vectors of numerical weights. Each network operator generates private masks using a combination of private keys and random seeds, effectively encrypting its model update before transmission. These masks are constructed such that, when all masked vectors are aggregated at the server, they cancel each other out, revealing only the sum of the original unmasked updates.

SA guarantees that individual model updates remain confidential while still enabling accurate computation of the global aggregate, thereby protecting against the Direct Server MIA and instance-level MIs. Additionally, in the event of eavesdropping by an external attacker, the attacker can only observe the encrypted data, making it impossible to infer the original data even if it is intercepted during transmission.

TABLE I: Network topologies used in evaluation.

Topology	Nodes	Links	Topology	Nodes	Links
Abilene	12	15	B4	12	19
Geant	22	36	Germany50	50	88
Nobel Germany	17	26	UsCarrier	158	189

VI. Evaluation

In this section, we present the experimental evaluation of SCARF. We compare its performance against non-collaborative baselines, and analyze the impact of the proposed privacy mechanisms on both routing performance and computational overhead.

A. Experimental Settings

We train 3 federated agents (network operators), each managing a different network domain as described in Table I. The training is performed with Flower [28], a well-known

Federated Learning simulator, which we enhance with local RL training following the Teal model [9]. We implement the security functions through the Flower *LocalDPMod* and *SecAgg+* modifiers. We evaluate the performance achieved by SCARF measuring the amount of demand that is successfully allocated, in the following way:

$$\text{Satisfied_demand } (\%) = \frac{\text{Demand_allocated}}{\text{Total_demand}} \times 100 \quad (1)$$

For each experiment, we perform 15 runs with different seeds and report the average value, along with 95% confidence intervals.

We evaluate and compare the following benchmarks:

- RL: a custom model trained independently for each network domain.
- SCARF: the baseline collaborative model that is trained without any security mechanisms.
- SCARF+DP: provides protection against global membership attacks and instance inversion attacks.
- SCARF+SA: focuses on security against direct server membership attacks and eavesdropping.
- SCARF+DP+SA: combines both DP and SA, making it the most secure option available.

B. RL vs SCARF

We first evaluate the performance of the federated setting against the non-collaborative RL scenario. For each topology, we compare a custom-trained RL model against our global SCARF model. We report the average satisfied demand across the four topologies in Fig. 2. SCARF outperforms RL in each training setting by 8% on average, achieving near-optimal performance in two scenarios. This gain is primarily due to its ability to aggregate knowledge from multiple network domains, thereby exposing the model to a broader range of traffic patterns that may not be present in a single domain. On the other hand, RL models are trained independently within each domain and thus learn only from locally observed traffic.

We also evaluate the zero-shot transferability of SCARF by deploying the trained model on unseen networks: B4, Germany50, and UsCarrier. Despite lacking any a-priori knowledge, SCARF achieves performance comparable to that of the specialized RL model. The collaboration among various network domains builds a robust model that benefits not only the networks used during training but also new, unseen environments without additional training.

C. Differential privacy budget study

To enable DP, we need to carefully tune two parameters: δ and ϵ. To maintain a sufficiently strong security guarantee, δ must be kept significantly smaller than the inverse of the number of training samples, so that the chance of a privacy leak is statistically not significant. We set $\delta = 10^{-5}$, similarly to other works [18], and derive empirically $C = 0.01$. To properly choose the ϵ parameter, we train the model using various ϵ values, ranging from 0.1 to 1, and then measure the satisfied demand at deployment time. As shown in Fig.3, smaller ϵ values lead to degraded performance because adjacent datasets become less distinguishable due to excessive noise injection. This also means that the model learns less useful information. Conversely, higher ϵ values reduce security since less noise is applied, resulting in better performance. In our case, the model achieves an average of 90% satisfied demand with higher values. For the remainder of our experiment, we select the model trained with $\epsilon = 1$ as a balance between performance and privacy. Accounting for composition across $R = 20$ rounds via Rényi DP [29], per-round $\epsilon_0 = 1, \delta_0 = 10^{-5}$ correspond to a cumulative privacy budget of $\epsilon_{total} \approx 4.2$ at $\delta_{total} = 2 \cdot 10^{-4}$.

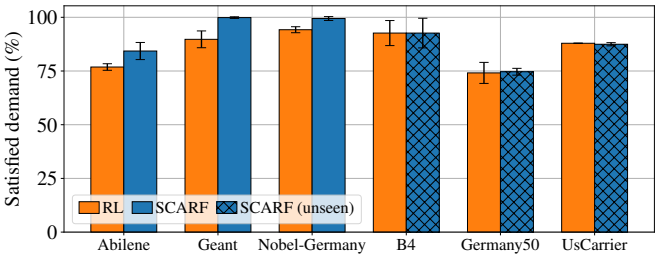


Fig. 2: Percentage of satisfied demand. SCARF outperforms RL in topologies seen during training, and achieves comparable performance when employed in unseen topologies.

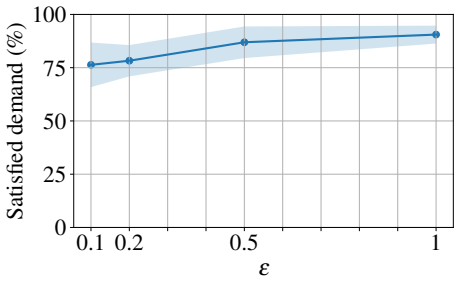
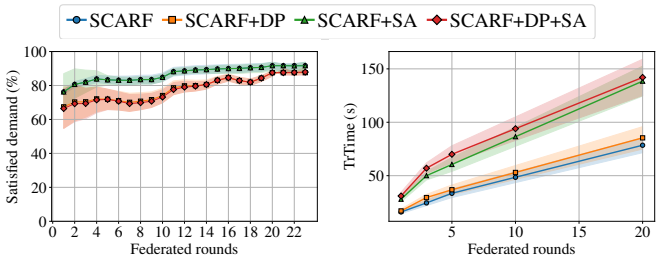


Fig. 3: Satisfied demand achieved by when varying ϵ values. $\epsilon = 1$ offers a reasonable privacy-performance trade-off.

D. Comparative Analysis of Security Configurations

We now evaluate the impact of DP and SA in terms of performance and training time overhead. As shown in Fig. 4, both SCARF and SCARF+SA converge at approximately 11 rounds, achieving up to 92% satisfied demand on average across all topologies. This is because SA protects the contributions of individual domains while accurately reconstructing the global average, without information loss. Both methods, therefore, converge to the same global model.

Conversely, the use of DP degrades the performance, as previously observed, due to the noise introduced into each domain's gradients. This not only slows convergence, which is reached after approximately 20 rounds, but also yields a lower level of satisfied demand. Despite this degradation, incorporating DP introduces only a limited training time overhead, as evident from Fig. 4b. In contrast, the cryptographic operations required during the SA aggregation process add extra time, slowing training. Notably, SCARF+DP+SA still achieves around 88% of satisfied demand, which represents



(a) (b)

Fig. 4: Evaluation of the benchmarks in terms of (a) satisfied demand and (b) training time. Despite additional training time, SCARF+DP+SA maintains a strong 88% satisfied demand, securely building a robust and generalizable model.

strong performance given the combined privacy guarantees of DP and SA. Although this configuration incurs the highest training time due to the added overheads, the increase is not particularly impactful in practice, as training is performed only once prior to deployment and does not affect runtime operation.

VII. Conclusion

In this paper, we present SCARF, a secure and collaborative framework for network routing across multi-domain environments. Because ISPs are typically unwilling to expose sensitive internal topology and traffic data, SCARF integrates DP and SA to FRL, ensuring operational privacy. Our experimental evaluations demonstrate that SCARF successfully balances routing performance with security. SCARF outperforms RL models by an average of 8% on known topologies and exhibited promising zero-shot adaptation across three unseen networks. While the necessary integration of DP noise injection and SA cryptographic primitives led to a slight performance drop and longer training times, the fully secure framework remained highly effective, maintaining an average demand satisfaction of around 88%.

References

[1] Z. A. El Houda, D. Nabousli, and G. Kaddoum, "Cost-efficient federated reinforcement learning- based network routing for wireless networks," in *2022 IEEE Future Networks World Forum (FNWF)*. IEEE, 2022, pp. 243–248.

[2] C. Alhachem, M. Kellil, and A. Bouabdallah, "Adaptive Federated Reinforcement Learning for Network Management in Complex Communication Networks," in *2026 IEEE Consumer Communications & Networking Conference*. IEEE, 2026.

[3] A. Sacco, D. Monaco, G. Marchetto, and P. Montuschi, "Dealing with challenged iot networks in hierarchical federated learning," *IEEE Internet of Things Journal*, vol. 12, no. 18, pp. 36 979–36 992, 2025.

[4] P. Sun, Z. Guo, J. Lan, J. Li, Y. Hu, and T. Baker, "Scaledrl: A scalable deep reinforcement learning approach for traffic engineering in sdn with pinning control," *Computer Networks*, vol. 190, p. 107891, 2021.

[5] D. M. Casas-Velasco, O. M. C. Rendon, and N. L. da Fonseca, "Drsir: A deep reinforcement learning approach for routing in software-defined networking," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 4807–4820, 2021.

[6] A. Angi, A. Sacco, F. Esposito, and G. Marchetto, "Routing with ART: Adaptive Routing for P4 Switches With In-Network Decision Trees," in *GLOBECOM 2024-2024 IEEE Global Communications Conference*. IEEE, 2024, pp. 3291–3296.

[7] D. Monaco, A. Sacco, F. Esposito, and G. Marchetto, "Adaptive sdn autoscaling via generalizable multi-agent reinforcement learning with eagle," *IEEE Transactions on Network and Service Management*, 2026.

[8] X. Tao, D. Monaco, A. Sacco, S. Silvestri, and G. Marchetto, "Delay-aware routing in software-defined networks via network tomography and reinforcement learning," *IEEE Transactions on Network Science and Engineering*, vol. 11, no. 4, pp. 3383–3397, 2024.

[9] Z. Xu *et al.*, "Teal: Learning-accelerated optimization of wan traffic engineering," in *Proceedings of the ACM SIGCOMM 2023 Conference*, 2023, pp. 378–393.

[10] Z. Luan *et al.*, "Mate: When multi-agent deep reinforcement learning meets traffic engineering in multi-domain networks," *Computer Networks*, vol. 247, p. 110399, 2024.

[11] G. Bernárdez *et al.*, "Magnneto: A graph neural network-based multi-agent system for traffic engineering," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, no. 2, pp. 494–506, 2023.

[12] Y. Guo, B. Lin, Q. Tang, Y. Ma, H. Luo, H. Tian, and K. Chen, "Distributed traffic engineering in hybrid software defined networks: A multi-agent reinforcement learning framework," *IEEE Transactions on Network and Service Management*, vol. 21, no. 6, pp. 6759–6769, 2024.

[13] Y. Jing *et al.*, "Federated multi-agent reinforcement learning: A comprehensive survey of methods, applications and challenges," *Expert Systems with Applications*, vol. 293, p. 128729, 2025.

[14] S. S. Suresh, V. Prabhu, V. Parthasarathy, G. Senthilkumar, and V. Gundu, "Intelligent data routing strategy based on federated deep reinforcement learning for iot-enabled wireless sensor networks," *Measurement: Sensors*, vol. 31, p. 101012, 2024.

[15] J. Li, A. Liu, G. Han, S. Cao, F. Wang, and X. Wang, "Fedrdr: Federated reinforcement distillation-based routing algorithm in uav-assisted networks for communication infrastructure failures," *Drones*, vol. 8, no. 2, p. 49, 2024.

[16] J. Fu, Y. Hong, X. Ling, L. Wang, X. Ran, Z. Sun, W. H. Wang, Z. Chen, and Y. Cao, "Differentially private federated learning: A systematic review," *ACM Computing Surveys*, vol. 58, no. 11, pp. 1–38, 2026.

[17] I. Ergün, H. U. Sami, and B. Güler, "Communication-efficient secure aggregation for federated learning," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 3881–3886.

[18] T. Stevens, C. Skalka, C. Vincent, J. Ring, S. Clark, and J. Near, "Efficient differentially private secure aggregation for federated learning via hardness of learning with errors," in *31st USENIX security symposium (USENIX Security 22)*, 2022, pp. 1379–1395.

[19] J. Xu, R. Hu, and O. Kotevska, "Optimal client sampling in federated learning with client-level heterogeneous differential privacy," *IEEE Internet of Things Journal*, 2026.

[20] W.-N. Chen, C. A. C. Choo, P. Kairouz, and A. T. Suresh, "The fundamental price of secure aggregation in differentially private federated learning," in *International Conference on Machine Learning*. PMLR, 2022, pp. 3056–3089.

[21] W. Zhou *et al.*, "A concurrent federated reinforcement learning for iot resources allocation with local differential privacy," *IEEE Internet of Things Journal*, vol. 11, no. 4, pp. 6537–6550, 2023.

[22] L. Bai, H. Hu, Q. Ye, H. Li, L. Wang, and J. Xu, "Membership inference attacks and defenses in federated learning: A survey," *ACM Computing Surveys*, vol. 57, no. 4, pp. 1–35, 2024.

[23] W. Yang, S. Wang, D. Wu, T. Cai, Y. Zhu, S. Wei, Y. Zhang, X. Yang, Z. Tang, and Y. Li, "Deep learning model inversion attacks and defenses: a comprehensive survey," *Artificial Intelligence Review*, vol. 58, no. 8, p. 242, 2025.

[24] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.

[25] K. N. Kumar, C. K. Mohan, and L. R. Cenkeramaddi, "The impact of adversarial attacks on federated learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 2672–2691, 2023.

[26] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of cryptography conference*. Springer, 2006, pp. 265–284.

[27] K. Bonawitz *et al.*, "Practical secure aggregation for federated learning on user-held data," *arXiv preprint arXiv:1611.04482*, 2016.

[28] D. J. Beutel *et al.*, "Flower: A friendly federated learning research framework," *arXiv preprint arXiv:2007.14390*, 2020.

[29] I. Mironov, "Rényi differential privacy," in *2017 IEEE 30th computer security foundations symposium (CSF)*. IEEE, 2017, pp. 263–275.