Sistema de Detecção de Intrusões baseado em Aprendizagem por Reforço Federada

André Santos Rocha[1], **Allan M. de Souza**[1]

[1]Instituto de Computação – Universidade Estadual de Campinas (UNICAMP)
Campinas – SP – Brazil

a235887@dac.unicamp.br, allanms@unicamp.br

Abstract. *In this project, the goal is to apply Federated Reinforcement Learning (FRL) techniques to the context of intrusion detection in Internet of Things (IoT) scenarios. Various essential IoT systems share sensitive information that needs to be transmitted securely and privately. Therefore, it is crucial to monitor these systems to prevent and defend against attacks that aim to expose their contents or weaken them. In this regard, we propose an anomaly detection FRL model, which is capable of efficiently assessing and detecting attacks, without sharing local data in the process.*

Resumo. *Nesse projeto, objetiva-se aplicar técnicas de Aprendizagem Federada por Reforço, (Federated Reinforcement Learning - FRL) ao contexto de detecção de intrusões em cenários de Internet das Coisas (Internet of Things - IoT). Diversos sistemas IoT são essenciais e compartilham informações sensíveis que precisam trafegar com segurança e privacidade. Sendo assim, é indispensável monitorar tais sistemas, evitando e defendendo ataques que visem expor seus conteúdos ou fragilizá-los. Nesse sentido, propõe-se um modelo de detecção de anomalias com FRL capaz de avaliar e detectar ataques com eficiência, sem, no entanto, compartilhar os dados locais nesse processo.*

1. Introdução

Com o crescimento dos sistemas distribuídos e da IoT, cresce também a vulnerabilidade cibernética [Wang et al. 2022]. Cada dispositivo conectado a um sistema torna-se um alvo em potencial e uma intrusão a um dispositivo pode comprometer a rede como um todo. Consequentemente, surgem diversos problemas não só devido ao vazamento de dados, mas devido também à possibilidade de comprometimento, mal funcionamento ou até mesmo de indisponibilidade do sistema. Isso é extremamente problemático, principalmente em cenários como o da saúde, em que o mau funcionamento dos dispositivos conectados pode resultar na perda de uma vida [Otoum et al. 2021].

Nesse cenário, o desenvolvimento de um Sistema de Detecção de Intrusões (*Intrusion Detection System* - IDS) se faz necessário. Os avanços de técnicas de Aprendizado de Máquina (*Machine Learning* - ML) têm surgido como abordagens promissoras para sistemas de IDS, permitindo detecções mais eficientes. Dentre tais abordagens, é possível destacar a Aprendizagem por Reforço (*Reinforcement Learning* - RL), uma alternativa robusta para garantir adaptabilidade ao modelo [Qi et al. 2021].

Com o fato dos padrões de ataques serem variados e se transformarem constantemente para superar as ferramentas atuais de prevenção, a RL se revela essencial e interessante ao desenvolvimento de um IDS. Essa técnica de ML permite grande adaptabilidade

por se basear num processo de aprendizagem semelhante ao humano. Isto é, baseado em experiências e nas suas recompensas [Vadigi et al. 2023].

Outro fator relevante é a vasta disponibilidade atual de dados, conhecida como *"Big Data"*, que é primordial na criação de modelos confiáveis [Zhang et al. 2021]. Entretanto, o *Big Data* também carrega um desafio consigo: a privacidade. Com as políticas de proteção de dados, tais como a LGPD (Lei Geral de Proteção de Dados) no Brasil e a GDPR (*General Data Protection Regulation*) [Zhang et al. 2021] na União Europeia, é preciso cautela ao lidar com informações pessoais por questões jurídicas e de segurança.

Nesse cenário, a Aprendizagem Federada (*Federated Learning* - FL) surge como um recurso seguro e confiável. A FL permite criar modelos a partir de múltiplos dispositivos, sem a necessidade de centralizar informações. Além disso, a FL é uma solução muito mais adequada para analisar dados espalhados por um sistema distribuído [Zhang et al. 2021]. Isso ocorre porque a FL compartilha modelos treinados localmente, o que permite a exposição a diferentes distribuições de dados, cruciais para um melhor desempenho e generalização [Sun et al. 2024]. Dessa forma, é possível utilizar o modelo em redes corporativas, por exemplo, sem que os nós compartilhem dados locais, mas ainda assim contribuam com o modelo global, proporcionando um sistema robusto e confiável.

Diante disso, propõe-se a combinação de FL e RL para criar um IDS baseado em FRL. Essa junção proporcionará um sistema: a) adequado às novas demandas legais de privacidade e com uma camada a mais de segurança por não transmitir dados locais; b) robusto e com grande poder de generalização garantido tanto pelo heterogeneidade dos dados quanto pela RL. O sistema analisará o fluxo da rede e detectará tentativas de ataques, prevenindo que eles atinjam os dispositivos naquela conectados.

2. Referencial Teórico

Esta seção apresenta uma visão geral sobre as principais referências necessárias ao desenvolvimento do projeto. Primeiramente, há uma descrição sobre classificações de IDS em 2.1. Em 2.2, discorre-se sobre a RL e sua estrutura básica, avançando mais em um dos seus algoritmos em 2.3. Por fim, há uma menção ao funcionamento da FL em 2.4.

2.1. Sistema de Detecção de Intrusões

IDS podem ser classificados de duas formas distintas: *network-based* (baseados em redes) ou *host-based* (baseados em um único *host*). Além disso, também é possível segmentar *network-based* IDS (NIDS) em *flow-based* (baseado em fluxo) ou *packet-based* (baseado em pacotes). Nesse trabalho, propõe-se um *flow-based* NIDS, isto é, o sistema monitora a rede e analisa fluxos de conjuntos de pacotes, realizando suas predições com base nisso [MAZETTO]. Por fim, também é possível trabalhar com a detecção de anomalias (*anomaly-based* - AB) ou de assinaturas (*signature-based* - SB) IDS. Sistemas SB aprendem padrões de ataques específicos, detectando ataques de acordo com os padrões já analisados. Por outro lado, sistemas AB observam qualquer desvio do padrão de funcionamento normal da rede e o identificam como um ataque [Vadigi et al. 2023]. Neste trabalho, propõe-se um *flow-based* NIDS, explorando a detecção de anomalias.

2.2. Aprendizagem por Reforço

A RL é uma técnica de aprendizado de máquina utilizada em ambientes que requerem interações de tentativa e erro para se construir um modelo. Essas interações podem

ser modeladas como um Processo de Decisão Markoviano (*Markov Decision Process* - MDP) devido aos seus cinco componentes: estado, ação, recompensa, política e valor [Figueiredo Prudencio et al. 2024]. Nesse processo, ilustrado na Figura 2, o agente obtém o seu estado atual no ambiente e se baseia nele para realizar ações. Essas são executadas dentro do ambiente, gerando recompensas positivas ou negativas, e um novo estado. Estas recompensas são úteis como parâmetro de ajuste da política, isto é, a maneira como o agente se comporta diante de um dado estado [Wang et al. 2024]. Veja, na Figura 2, que o agente permanece nesse ciclo de aprendizagem até alcançar uma política ótima, que maximize o valor da recompensa recebida [Vadigi et al. 2023].

2.3. Deep Q-Network

Q-Learning é um algoritmo de RL classificado como *Model-Free*, ou seja, o agente aprende por meio de interações com o ambiente e a sua trajetória é utilizada para o aprendizado do modelo [Otoum et al. 2021]. Além disso, a técnica é baseada na estimativa do valor da função estado-ação $Q(s_t, a_t)$, onde s_t e a_t são, respectivamente, o estado e a ação tomada no instante (t). O valor de retorno de Q é a soma de todas as recompensas esperadas ao se tomar uma dada ação em um estado particular e seguir a política de decisão do agente até o fim da interação [Harmon and Harmon 1996].

No algoritmo de *Deep Q-Network* (DQN), uma rede neural representa Q. Para atualizar o seu valor, utiliza-se a diferença temporal (*Temporal Diference* - TD), que mede a adaptação do agente ao ambiente. A equação que atualiza o valor de Q é:

$$Q^{new}(s_t, a_t) = Q(s_t, a_t) + \alpha * [r_t + \gamma * max_a\{Q(s_{t+1}, a)\} - Q(s_t, a_t)] \quad (1)$$

onde r_t, α, γ, $max_a\{Q(s_{t+1}, a)\}$ são a recompensa obtida no instante t, a taxa de aprendizado, o fator de desconto e a estimativa futura máxima de Q, respectivamente. Nessa equação, TD é dado por $[r_t + \gamma * max_a\{Q(s_{t+1}, a)\} - Q(s_{t+1}, a_t)]$ [Vadigi et al. 2023].

2.4. Aprendizagem Federada

A FL é um processo distribuído, no qual diversos dispositivos (i.e., clientes), treinam modelos locais e são coordenados por um servidor global [Pinto Neto et al. 2023].

Inicialmente, os clientes recebem um modelo global não treinado e o treinam localmente com seus dados, como mostra a Figura 1. Após essa etapa, perceba na Figura 1 que cada cliente obterá um modelo local, cujos pesos são compartilhados com um servidor central onde serão agregados, criando um novo modelo global a ser utilizado na próxima rodada do aprendizado [Zhang et al. 2021].

Para a agregação, é possível utilizar diversas estratégias. Uma das abordagens tradicionais para a agregação é o *Federated Average* (Média federada - FedAVG), que consiste em realizar a média ponderada entre os pesos dos modelos dos clientes, levando em consideração a quantidade de dados de cada cliente [MAZETTO].

3. Trabalhos Relacionados

Nos trabalhos sobre IDS, frequentemente encontra-se a utilização de técnicas de FL, RL e FRL. Aqui, propõe-se uma breve revisão dos principais trabalhos encontrados.

Em [Alavizadeh et al. 2022], discutem-se os resultados de um modelo baseado em DQN para IDS. O autor explora a técnica para um IDS SB e demonstra resultados

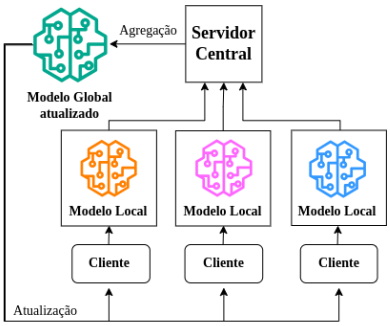


Figura 1. Figura Ilustrativa FL

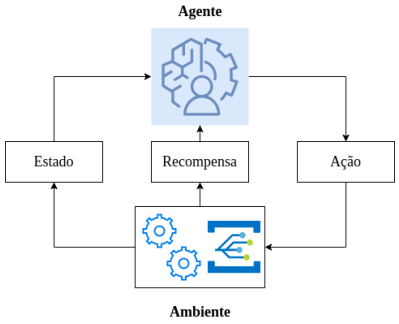


Figura 2. Figura Ilustrativa RL

superiores a outras técnicas anteriormente implementadas no conjunto de dados NSL-KDD. Já o trabalho [Lopez-Martin et al. 2020] vai além, pois utiliza diversos algoritmos de RL (DQN, *Double Deep Q-Network*, *Policy Gradient* e *Actor Critic*), tanto no conjunto NSL-KDD quanto no AWID, ambos amplamente utilizados na literatura. Sobre o uso de FL, vê-se, em [MAZETTO], a FL aplicada a um IDS para redes domésticas, destacando o treinamento executado com diferentes conjuntos de dados para diferentes clientes.

O artigo [Qi et al. 2021] propõe uma análise ampla sobre a FRL, explicando seus pilares e exibindo suas principais aplicações. Dentre elas, o trabalho cita a detecção de ataques, pois modelos centralizados tradicionais apresentam alta taxa de falsos positivos e desconsideram a privacidade dos usuários. Além disso, comenta importantes aplicações, como [Mowla et al. 2020], em que é apresentado um modelo de defesa de ataques contra *unmanned aerial vehicles* (UAVs) baseado em FRL.

No trabalho [Vadigi et al. 2023], um IDS é desenvolvido com múltiplos agentes posicionados na rede. Cada agente executa um algoritmo de DQN localmente e compartilha os pesos com um servidor central. Ademais, a cada agente é atribuído um valor de atenção, útil para ponderar a participação dos pesos locais na agregação dos modelos. O valor de atenção é projetado com base na acurácia e no número de dados de treinamento. De maneira análoga, [Otoum et al. 2021] aplica a FRL no desenvolvimento de um IDS voltado para infraestruturas IoT no contexto da saúde, prevenindo o vazamento de dados pessoais e assegurando robustez ao modelo. O trabalho também utiliza o algoritmo de DQN, mas opta pelo uso do conjunto de dados CICIDS2017.

Por outro lado, [Wang et al. 2022] desenvolveram um IDS AB aplicado à IoT industrial. O texto apresenta o uso do algoritmo *Deep Deterministic Policy Gradient* para treinamento de modelos locais e implementação de entidades regionais, locais e globais, que administram a detecção de anomalias.

Sobre o trabalho proposto, o objetivo é analisar a utilização da FRL para desenvolver um IDS AB. Semelhantemente aos trabalhos [Vadigi et al. 2023] e [Otoum et al. 2021], há múltiplos agentes posicionados na rede, mas explora-se diferentes métricas de desempenho - inclusive em um conjunto de dados com ataques desconhecidos - e compara-se os resultados com outras alternativas tanto centralizadas quanto federadas.

4. Metodologia de Desenvolvimento

Para o sistema, utilizou-se a FRL. Além disso, dividiu-se os dados em cinco partições, cada uma representando uma rede e simbolizada por um cliente. A cada rodada, executou-se localmente o algoritmo de DQN e compartilhou-se os pesos da *Q-network* com o servidor, onde foram agregados com FedAVG, a fim de formar um novo modelo. Então, este foi compartilhado com os clientes para atualização local.

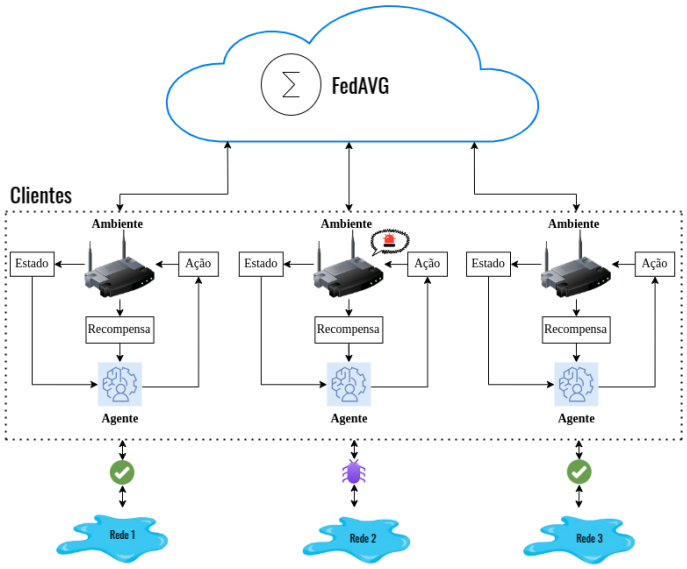


Figura 3. Figura ilustrativa IDS com FRL

Na Figura 3, é possível visualizar o sistema em funcionamento. Nota-se que o IDS proposto pode ser executado em *gateway* de uma rede corporativa, por exemplo, identificando possíveis ataques em redes distintas. Como cada rede pode sofrer ataques diversos e heterogêneos, o processo de agregação torna, a cada rodada, o modelo mais robusto. Isso porque, como mostra a Figura 3, os agentes atuantes nas Rede 1 e 3 aprendem com os dados do ataque sofrido na Rede 2, prevenindo-se antes mesmo de sofrê-lo.

Para implementar a FRL, foram utilizadas as ferramentas *Flower* e *Stable-Baselines3*. O primeiro auxilia no gerenciamento da relação cliente-servidor e possui estratégias de agregação, como o FedAVG, já implementadas. Na Tabela 1, visualizamos os parâmetros de inicialização do modelo determinados experimentalmente.

5. Conjunto de dados e pré-processamento

O conjunto de dados utilizado foi o CICIDS2017. Este conjunto foi publicado em 2017 e contém registros de 14 diferentes tipos de ataques. Os dados foram coletados ao longo de cinco dias pelo *Canadian Institute of Cybersecurity* por meio do monitoramento de diversos dispositivos conectados a uma rede. O conjunto possui 2830540 instâncias e um total de 83 atributos sobre o fluxo e os pacotes que passam pela rede.

Para o pré-processamento, foi preciso balancear o conjunto, pois alguns ataques representavam proporções muito baixas frente ao número total de instâncias apresentadas na Tabela 2. Após o balanceamento, aplicou-se também a normalização dos dados e

Tabela 1. Tabela com Parâmetros do Sistema

Parâmetros	Descrição	Valores
num-episodes	Número de episódios por rodada	125
num-iterations	Número de iterações por episódio	100
hidden layers	Número de camadas escondidas	2
num-units	Número de neurônios escondidos	2 x 64
learning rate	Taxa de aprendizado da Q-Network	0.00003
activation function	Função não-linear de ativação	ReLU
epsilon	Valor inicial da probabilidade da escolha de uma ação aleatória	1.0
gamma	Fator de desconto para a predição da melhor ação	0.99
batch-size	Tamanho do *batch* para cada atualização do gradiente	32

Tabela 2. Tabela de distribuição do conjunto de dados

Tráfego	**Normal**	***Hulk***	***PortScan***	***DDoS***	***Goldeneye***	***FTP-P.***	***SSH-P.***	***Bot***
Instâncias	*2273097*	*231073*	*158930*	*128027*	*10293*	*7938*	*5897*	*1966*
Tráfego	***Slowloris***	***Slowhttptest***	***Brute Force***	***XSS***	***Infiltration***	***Sql Injection***	***Heartbleed***	
Instâncias	*5796*	*5499*	*1507*	*652*	*36*	*21*	*11*	

uma redução de dimensões por meio da análise de componentes principais (PCA). Com o balanceamento, o conjunto passou a ter apenas 8 ataques distintos.

Por fim, gerou-se um conjunto de dados com natureza binária, sendo que, para construí-lo, foram mantidas 7000 instâncias de cada um dos 8 ataques e adicionadas 56000 instâncias de cenários "*benignos*", isto é, cenários sem ataques. Posteriormente, as instâncias foram rotuladas apenas como "*ataque*" ou "*benigno*".

6. Experimentos

Para conduzir os experimentos, utilizaram-se as técnicas Regressão Logística, *XGBoost* e *Multilayer-Perceptron* (MLP) federadas. A técnica DQN foi utilizada federada e centralizada. Além disso, os experimentos foram conduzidos com 5 clientes ao longo de 100 rodadas. Veja os resultados de Acurácia, Precisão e Revocação na Figura 4.

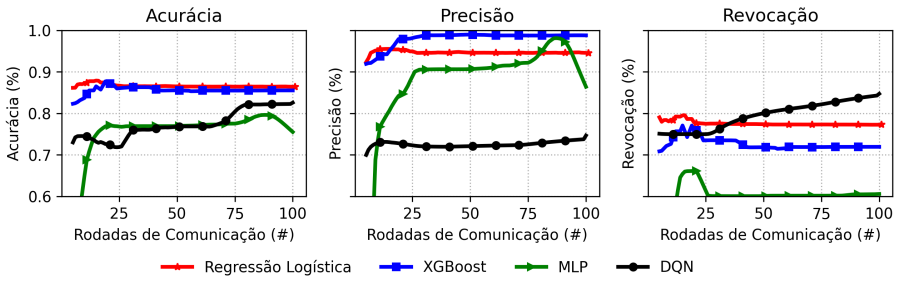
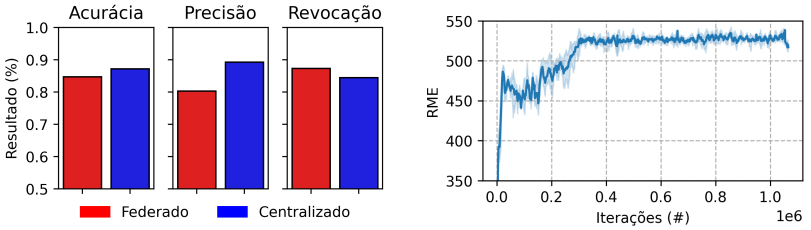


Figura 4. Métricas x *Rodadas*

Na Figura 4, verifica-se que a Regressão Logística superou nosso modelo em acurácia, o qual obteve 82.28% na métrica. Sobre as métricas de precisão e revocação, verifica-se que os resultados da FRL foram o quarto e o primeiro melhores, respectivamente. Inclusive, o alto resultado na métrica revocação, 84.41%, evidencia a superioridade do modelo de FRL em evitar que ataques despercebidos invadam a rede.

É possível também verificar que federar a DQN não prejudicou o desempenho significativamente. Veja, na Figura 5(a), que a comparação entre os resultados da DQN federada e centralizada mostra desempenhos semelhantes.



(a) Centralizado vs. FL (b) Evolução da RME no treinamento

Figura 5. Comparação de métricas entre DQN centralizada e federada e evolução da RME

Foi realizado também um teste com uma base de ataques desconhecidos - isto é, ataques ausentes nos dados de treino - para verificar o desempenho do modelo diante de um cenário inesperado. Na Figura 6, fica evidente que modelo de FRL é mais robusto no que se refere à generalização com dados desconhecidos, o que permite ao sistema impedir ataques inéditos que utilizem técnicas avançadas ou incomuns. Nosso modelo atingiu 86.48% na acurácia, 84.66% na precisão e 89.09% na revocação nessa etapa dos experimentos. Veja os resultados abaixo na Figura 6.

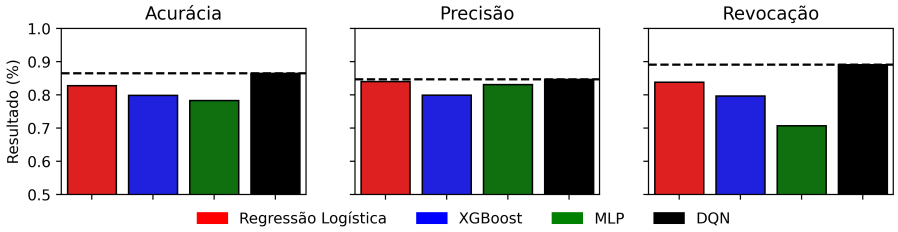


Figura 6. Comparação de métricas no cenário com ataques desconhecidos

Uma outra métrica observada é a evolução da recompensa média por episódio (RME) dos clientes. As recompensas foram ajustadas para a favorecer a revocação, crucial por ser uma aplicação crítica. Veja, na Figura 5(b), o crescimento da curva enquanto o modelo se ajusta aos dados.

Logo, os experimentos sugerem que FRL pode ser uma abordagem promissora para IDS. Com o compartilhamento de conhecimento entre os sistemas, foi possível assegurar uma melhor generalização do modelo à detecção de ataques. Ademais, com a adaptabilidade do aprendizado por reforço, foi possível a detecção de ataques para os quais o modelo não havia sido treinado, superando as outras técnicas exploradas.

7. Conclusão

Neste trabalho, propôs-se um IDS baseado em FRL. O modelo foi desenvolvido com o conjunto de dados CICIDS2017. Os experimentos discutidos na seção 6 evidenciam um sistema com bom desempenho, quando comparado ao modelo centralizado e às outras técnicas de ML. Para trabalhos futuros, visamos ganho de desempenho, testes em ambientes reais e em *testbeds*.

8. Agradecimentos

Este projeto foi apoiado pelo Ministério da Ciência, Tecnologia e Inovações, com recursos da Lei nº 8.248, de 23 de outubro de 1991, no âmbito do PPI-SOFTEX, coordenado pela Softex e publicado Arquitetura Cognitiva (Fase 3), DOU 01245.003479/2024-10. Além disso, os autores agradecem ao Programa Institucional de Bolsas de Iniciação Científica e Tecnológica da UNICAMP e ao projeto FAEPEX/PIND UNICAMP projeto #519.287.

Referências

Alavizadeh, H., Alavizadeh, H., and Jang-Jaccard, J. (2022). Deep q-learning based reinforcement learning approach for network intrusion detection. *Computers*, 11(3):41.

Figueiredo Prudencio, R., Maximo, M. R. O. A., and Colombini, E. L. (2024). A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8):10237–10257.

Harmon, M. E. and Harmon, S. S. (1996). Reinforcement learning: A tutorial. *WL/AAFC, WPAFB Ohio*, 45433:237–285.

Lopez-Martin, M., Carro, B., and Sanchez-Esguevillas, A. (2020). Application of deep reinforcement learning to intrusion detection for supervised problems. *Expert Systems with Applications*, 141:112963.

MAZETTO, B. O. Aprendizado federado para detecção de intrusões.

Mowla, N. I., Tran, N. H., Doh, I., and Chae, K. (2020). Afrl: Adaptive federated reinforcement learning for intelligent jamming defense in fanet. *Journal of Communications and Networks*, 22(3):244–258.

Otoum, S., Guizani, N., and Mouftah, H. (2021). Federated reinforcement learning-supported ids for iot-steered healthcare systems. In *ICC 2021 - IEEE International Conference on Communications*, pages 1–6.

Pinto Neto, E. C., Sadeghi, S., Zhang, X., and Dadkhah, S. (2023). Federated reinforcement learning in iot: Applications, opportunities and open challenges. *Applied Sciences*, 13(11).

Qi, J., Zhou, Q., Lei, L., and Zheng, K. (2021). Federated reinforcement learning: techniques, applications, and open challenges. *Intelligence amp; Robotics*.

Sun, Z., Niu, X., and Wei, E. (2024). Understanding generalization of federated learning via stability: Heterogeneity matters. In Dasgupta, S., Mandt, S., and Li, Y., editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 676–684. PMLR.

Vadigi, S., Sethi, K., Mohanty, D., Das, S. P., and Bera, P. (2023). Federated reinforcement learning based intrusion detection system using dynamic attention mechanism. *Journal of Information Security and Applications*, 78:103608.

Wang, X., Garg, S., Lin, H., Hu, J., Kaddoum, G., Jalil Piran, M., and Hossain, M. S. (2022). Toward accurate anomaly detection in industrial internet of things using hierarchical federated learning. *IEEE Internet of Things Journal*, 9(10):7110–7119.

Wang, X., Wang, S., Liang, X., Zhao, D., Huang, J., Xu, X., Dai, B., and Miao, Q. (2024). Deep reinforcement learning: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):5064–5078.

Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., and Gao, Y. (2021). A survey on federated learning. *Knowledge-Based Systems*, 216:106775.