UNICAMP

Seminário

IoTDefender: A Federated Transfer Learning Intrusion Detection Framework for 5G IoT

MO809 - Tópicos em Computação Distribuída
Prof. Luiz Fernando Bittencourt

Autor: Marcos Paulo
RA: 173700
Data: 27/10/2022

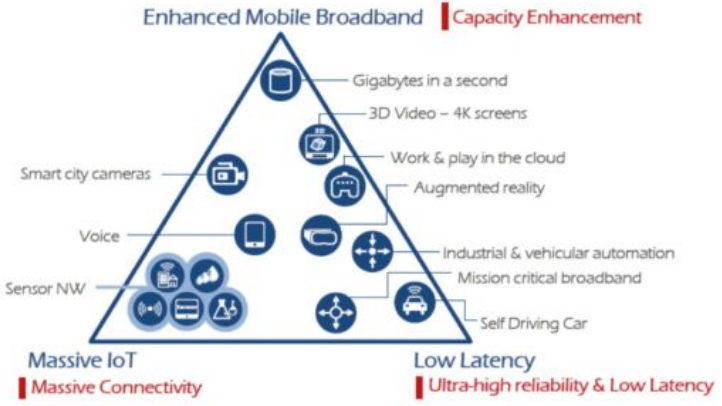
Sumário

- Motivação
- Conceitos
- Trabalhos Correlatos
- Contribuições
- Detalhes do IoTDefender
- Implementação
- Avaliação
- Discussão e Conclusão

5G IoT, Mobile Edge Computing(MEC), Intrusion detection, Federated transfer learning

I. Introdução

- Expansão do IoT
- Aplicações de 5G - eMBB, uRLLC, mMTC
- Mobile Edge Computing (MEC)
- Intrusion Detection Systems (IDS)



Mobile Edge Computing (MEC)

Processamento massivo de dados rápido e em tempo real.

Segurança de dados e proteção de privacidade

Suporte móvel e de posicionamento para dispositivos IoT

Intrusion Detection System (IDS) Desafios

5G IoT são sistemas heterogêneos e distribuídos

5G IoT precisa assegurar isolamento de dados e privacidade.

Dispositivos IoT com pouca quantidade de dados para treinamento

Contribuições

1º framework de federated transfer learning para IDS em 5G IoT
Framework Hierárquico, flexível e extensível
Testes com redes de dispositivos reais
Privacidade de dados preservada e boa performance

II. Trabalhos Correlatos - IDS

- Inflexível e difícil de estender
- Restringido por protocolos
- Aplicado apenas a alguma aplicação IoT específica

Trabalhos Correlatos - Federated Learning

- DioT
- Limitações:
  - Usa um modelo de arquitetura unificado que não considera modelos de IoT personalizados
  - Dificilmente detecta ataques novos ou desconhecidos

Transfer Learning

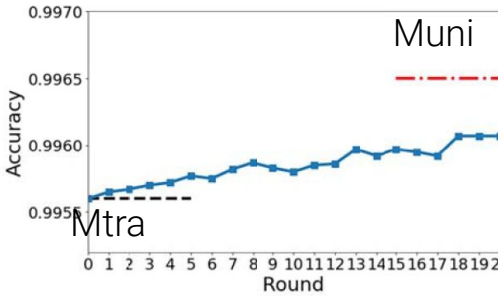
- Transferir conhecimento de um domínio existente para um novo domínio
- Resolve treinamento de modelos com poucos dados
- Ainda não foi aplicado na segurança da rede
- Sua combinação com aprendizado federado pode quebrar as barreiras de dados entre organizações

III. Definição do problema

- Proteger a privacidade de diferentes dados de rede IoT em vez de compartilhá-los diretamente

- Melhorar o desempenho do Modelo IoTDef muito além do modelo individual M-TRA (modelo personalizado)

- garantir sua precisão o mais próximo possível do M-UNI (modelo unificado)



Overview do framework

- 3 camadas
  - Nuvem de segurança
  - Plataformas MEC
  - Dispositivos IoT

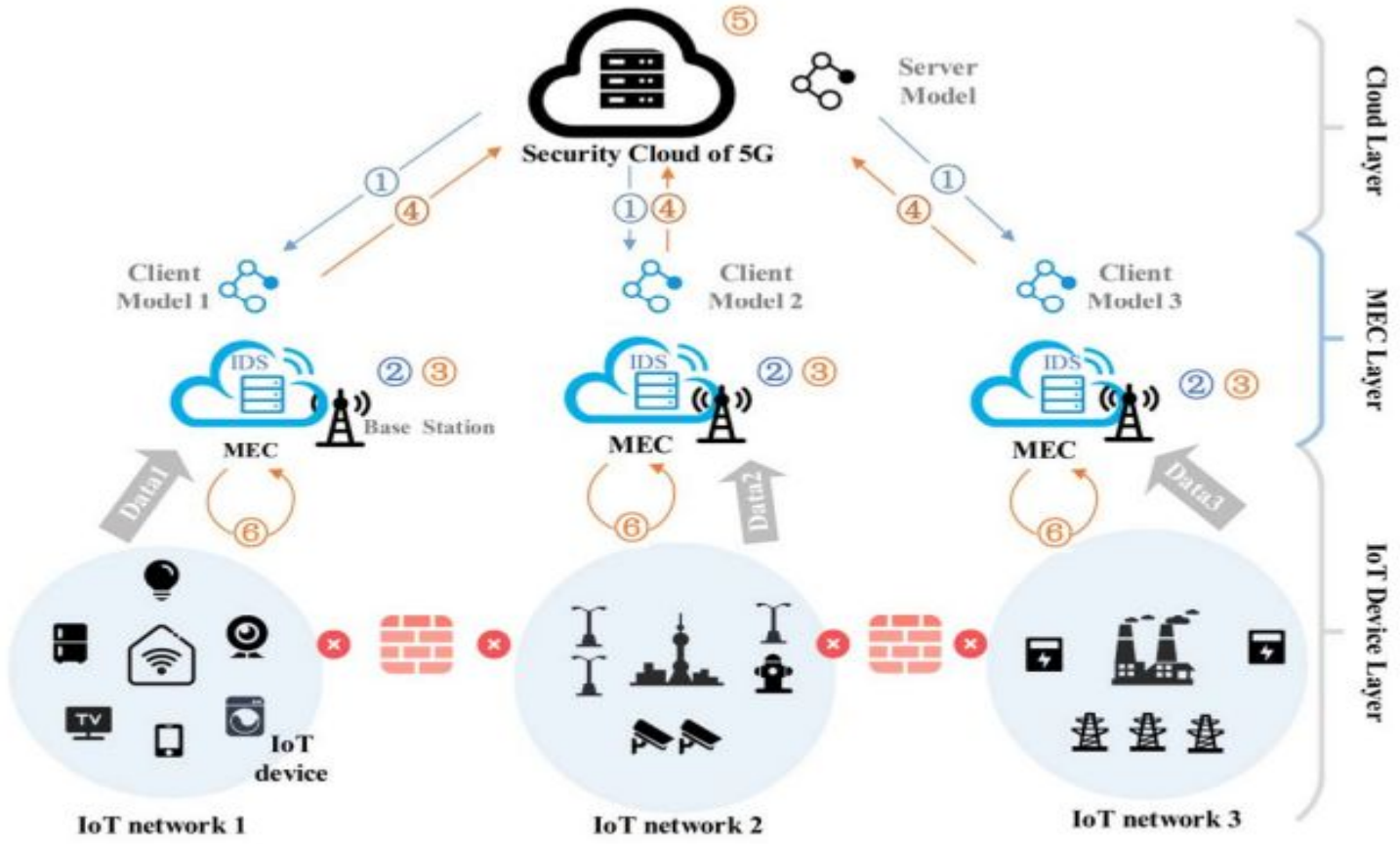


Fig. 1. Architecture of IoTDefender.

Model Training

1- Modelo do servidor é treinado com o dataset público e distribuído para todos os MEC

2- Cada MEC treina seu modelo de cliente com seu dataset local privado. (Transfer Learning)

3- Cada MEC computa o “logits” de cada modelo cliente com base no dataset público como entrada.

4- Cada MEC faz o upload de “logits” para a nuvem de segurança

5- O servidor integra eles, e transmite o novo “logit” para os clientes MEC.

6- Cada MEC treina um modelo de cliente novamente no conjunto de dados públicos por algumas épocas para obter um modelo de cliente personalizado.

7- Repete etapas 3 a 6 até a convergência

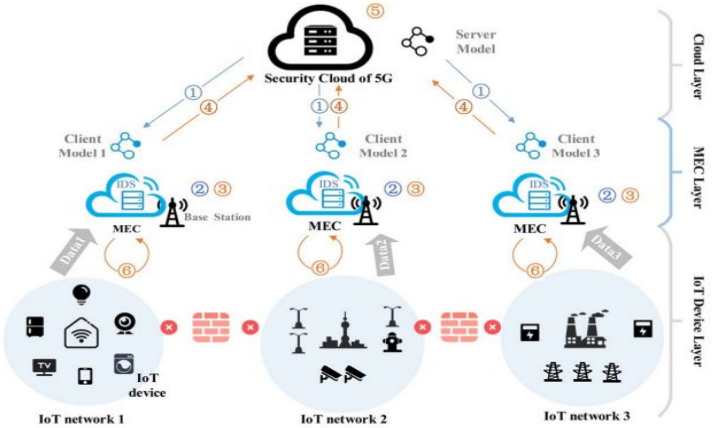


Fig. 1. Architecture of IoTDefender.

Model Training - Federated Learning

Resolve problema de isolamento de dados

Etapas:

-Aprendizado do modelo de servidor

-Aprendizado do modelo cliente

Model Training - Transfer Learning

Resolve o problema de privacidade de dados, isolamento de dados e de heterogeneidade dos dados do modelo servidor e de cliente.

Usado após gerar o modelo de servidor para criar um modelo cliente utilizável em dispositivos IoT

Dados públicos: Redes tradicionais

Dados privados: Dados da rede IoT

Neural Network

2 redes convolucionais

Input: Dados de rede

Output: Classes de tráfego (Normal ou anormal)

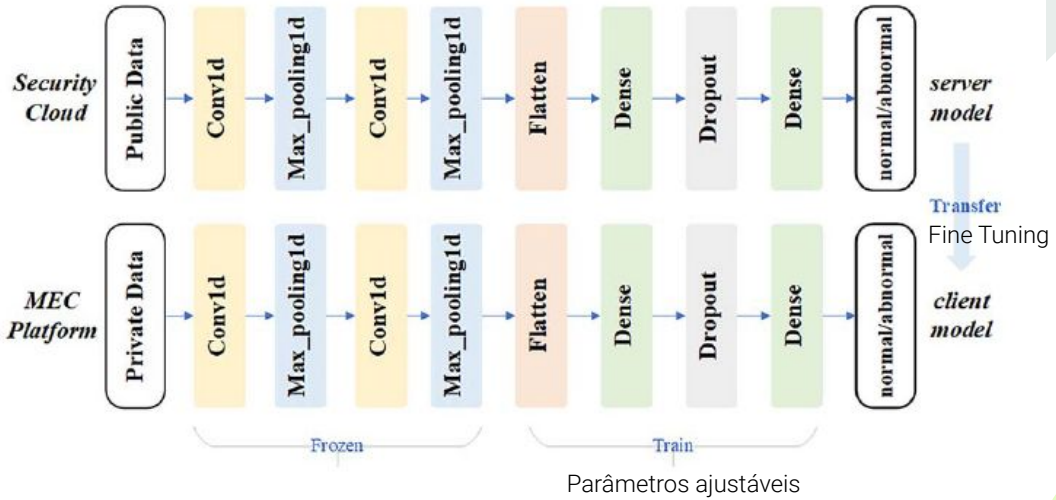


Fig. 2. Transfer learning process.

Learning Process

Adaptação de domínio

Maximum Minimum Variance (MMD) para calcular diferença entre datasets

Quanto maior MMD maior a diferença entre datasets

Soma a distância MMD com a perda de treinamento de classificação(Lc) para obter a função de perda

Loss Function

$$L = L_C(X_L, y) + \lambda D^2_{MMD}(X_S, X_T) \qquad (4)$$

Xs e Xt = Domínios de origem(source) e destino(target)
λ = balanço do peso do MMD

Algoritmo

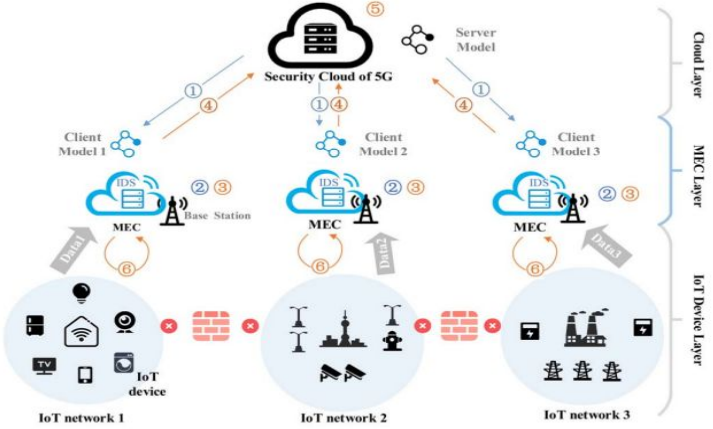


Fig. 1. Architecture of IoTDefender.

Algorithm 1 The learning procedure of IoTDefender

Input: public dataset D_0, private dataset D_k

Output: trained model f_k , $k = 1, 2, \ldots, n$

1: **Initialization:** Train a CNN model f_S with public dataset D_0 on Security Cloud.
2: **Distribution:** The server model f_S is distributed to all the MEC platforms.
3: **Transfer learning:** Each MEC platform trains client model f_k on public and private dataset D_0, D_k using (4).
4: **Federated Process:**
5: **for** round = 1, 2..., r **do**
6: Each MEC platform calculate the logits l_k on public dataset D_0 and upload it to the Security Cloud platform.
7: The Security Cloud aggregates the logits of all MEC platforms and calculates the average logits l_{avg}.
8: The Security Cloud sends l_{avg} to all MEC platforms.
9: Each MEC platform trains client model f_k on public dataset D_0 to make its logits close to l_{avg}.
10: **Transfer learning:** Each MEC platform trains client model f_k on the private dataset D_k again based on Fine-tune.
11: **end for**

IV. Experimentos

TABLE I
THE DATASETS TO EVALUATE IOTDEFENDER

Dataset	Attack type	# Total Packets
CVE-CIC-IDS	PortScan, DDoS, FTP-Patator, SSH-Patator, Bot, Heartbleed	2300825
IOT-dataset 1(P1)	ARP Spoofing, Dos, Scanning, Mirai	127358
IOT- dataset 2(P2)	Mirai	764137
IOT- dataset 3(P3)	ARP MitM, Dos, Fuzzing, OS Scan	721276
NSL-KDD(P4)	DoS, Probe, R2L, U2R	494020

Smart Home

Smart Home (9 aparelhos)

8 Câmeras de vigilância

CVE-CIC-IDS e NSL-KDD = datasets públicos
IOT Datasets = datasets privados

Implementação

Uso de CNN, 60% de dados para treinamento, learning rate = 0.1, batch size=64, épocas de treinamento = 10. Experimentos simulados 5 vezes e computado a média dos valores.

Experimento 1 - habilidade de detecção de ataques

Experimento 2 - habilidade de generalização

Comparação de performance entre:

- Métodos tradicionais de ML (KNN, Adaboost, Random Forest, CNN)
- Modelo somente com TF
- Modelo somente com FED - Conceito do DIoT

Resultados Detecção Acurácia

TABLE II
DETECTION ACCURACY(%) OF THE TEST CLIENT

Client	AB	RF	KNN	CNN	TF	FED	IoTDef
P1	97.12	98.96	99.57	99.54	99.56	99.56	99.60
P2	99.84	98.14	97.65	99.10	99.31	99.66	99.75
P3	77.82	77.99	74.67	84.23	82.84	85.07	86.37
P4	80.58	80.67	80.62	77.71	75.71	78.06	81.99
AVG	88.84	88.94	88.12	90.14	89.35	88.84	91.93

$$ACC = (TP + TN)/(TP + TN + FP + FN)$$

Acurácia na classificação de ataques nos clientes

Resultados Detecção Acurácia

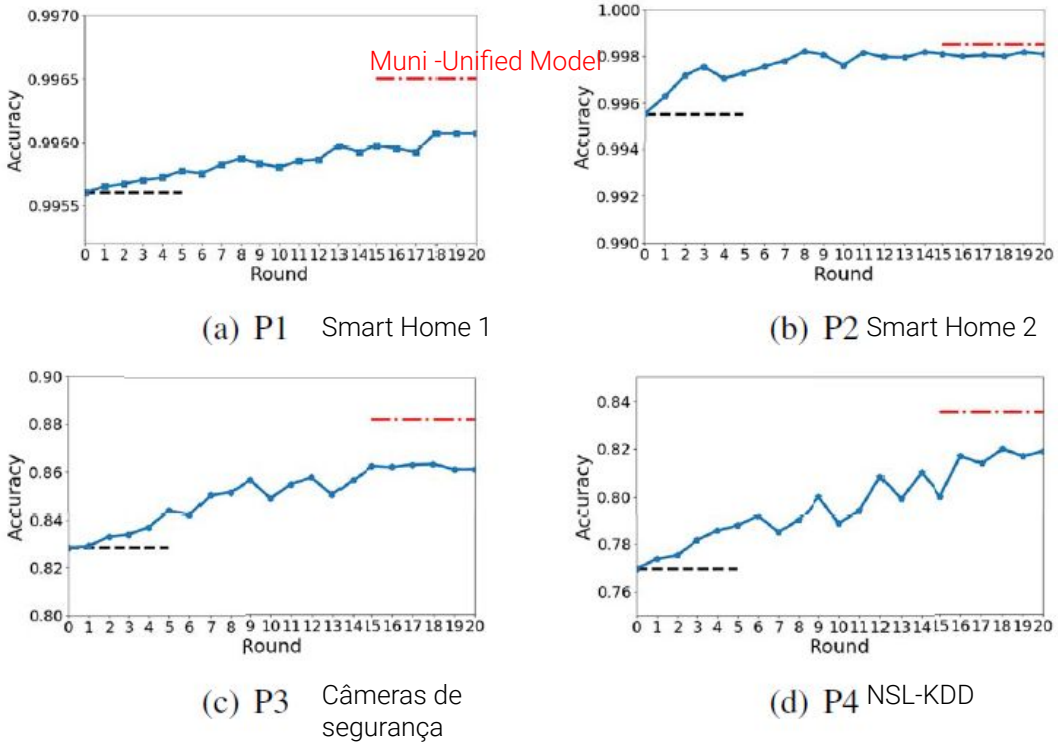


Fig. 3. The test accuracy of the client models.

Linha Vermelha (ideal) - Dataset público + Todos os datasets privados

Linha preta (mínimo) - Dataset público + dataset privado individual

Resultados do modelo generalizado

Capacidade do modelo detectar ataques desconhecidos com ajuda do dataset público

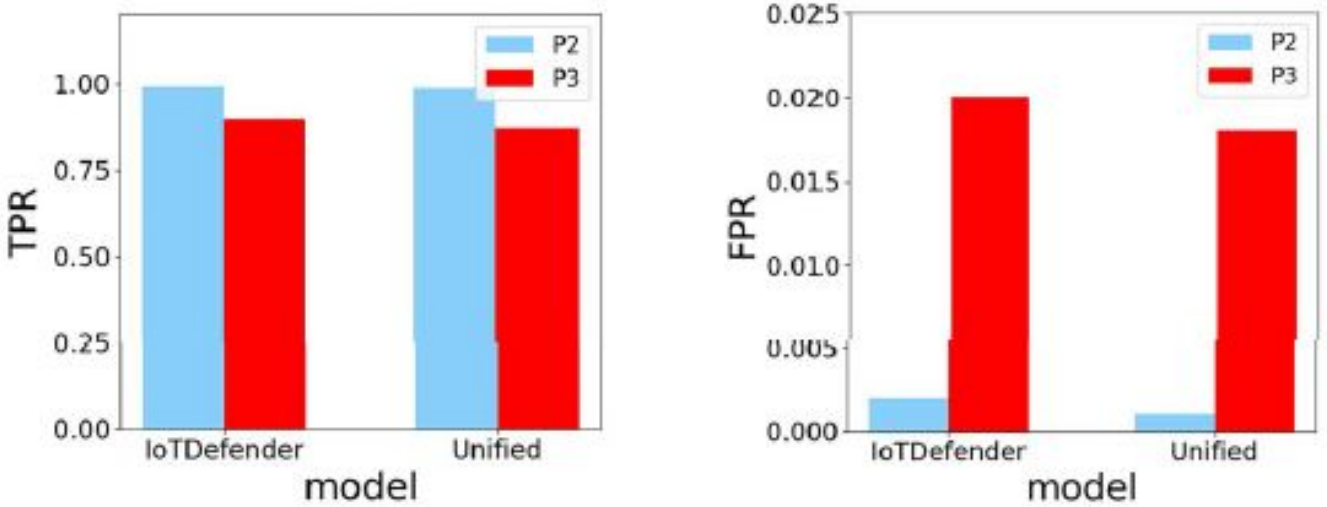
Remove o ataque do dataset de treino de 1 dos clientes, mas mantém no demais clientes e no dataset de teste.

TABLE III
DETECTION ACCURACY (%) OF UNKNOWN ATTACKS

Client	Unknown attack	AB	RF	CNN	TF	IoTDef
P1	Mirai	92.10	96.63	99.51	99.75	99.89
	Dos	77.03	71.18	77.80	99.90	99.97
	AVG1	84.56	83.90	88.65	99.82	99.93
P2	Mirai	-	-	-	42.31	81.30
P3	OS Scan	79.3	77.46	71.07	81.22	82.98
AVG					84.60	92.81

*Em P2: O dataset público não contém dados do Mirai. TF usa o modelo de servidor e o IoTDef os dados de P1 e P3

Comparação com modelo unificado



(a) TPR (b) FPR

Fig. 4. TPR and FPR of P2 and P3.

*FPR do IoTDefender é menor, e a acurácia e o TPR de uma rede unificada é maior

V. Discussão e Trabalhos Futuros

- Quais clientes selecionar os para o aprendizado federado(5G Slice)
- Aplicar métodos para assegurar a qualidade da comunicação e evitar perdas. Reduzir número de rodadas.
- Treinar dados de tráfego de rede online com método de aprendizado incremental para prevenir ataques completamente novos à rede.

VI. Conclusão

- Manutenção da privacidade e segurança dos dados
- Framework de transferência de aprendizado federado para segurança de IoT 5G
- Detecção de pacotes maliciosos com 91,93% de precisão
- Habilidade de generalização

Novos trabalhos

D¨IOT: A Federated Self-learning Anomaly Detection System for IoT [2019]

Primeiro sistema a empregar uma abordagem de aprendizado federado para detecção de intrusão baseada em detecção de anomalias

- Altamente eficaz (taxa de detecção de 95,6%)
- Rápido (≈ 257 ms)
- Não requer nenhuma intervenção humana ou dados rotulados para operar.

Local Differential Privacy-Based Federated Learning for Internet of Things [10-2020]

- Internet dos Veículos (IoV).
- Evitar a ameaça à privacidade e reduzir o custo de comunicação
- Propõe quatro mecanismos de privacidade diferencial local (LDP) para perturbar gradientes gerados por veículos
- Uso intensivo de dados experimentais.

Novos Trabalhos

Federated Deep Learning for Zero-Day Botnet Attack Detection in IoT-Edge Devices [06-2021]

O método tradicional de DL centralizado (CDL) não pode ser usado para detectar o ataque de botnets anteriormente desconhecidos (dia zero) sem violar os direitos de privacidade de dados dos usuários. O artigo, propõe o método DL federado (FDL) para detecção de ataque de botnet desde o dia zero para evitar vazamento de privacidade de dados em dispositivos IoT-edge.

Novos Trabalhos

Internet of Things Intrusion Detection System based on Transfer Learning [05-2022]

Ao contrário do trabalho anterior na extração de recursos projetados manualmente, esse método mantém o desempenho de aprendizado de ponta a ponta do Deep Learning (DL), reduz o risco de migração de conceito e reduz a intervenção humana.

FL-Defender: Combating Targeted Attacks in Federated Learning [07-2022]

Neste artigo, são analisados ataques direcionados contra FL e deles os autores descobrem que os neurônios na última camada de um modelo de aprendizado profundo (DL) que estão relacionados aos ataques exibem um comportamento diferente dos neurônios não relacionados, tornando os gradientes da última camada recursos valiosos para detecção de ataques. O FL-Defender atinge as menores taxas de sucesso de ataque, mantém o desempenho do modelo global na tarefa principal e causa sobrecarga computacional mínima no servidor.

Referência:

Yulin Fan, Yang Li, Mengqi Zhan, Huajun Cui, Yan Zhang.

“**IoTDefender: A Federated Transfer Learning Intrusion Detection Framework for 5G IoT**”.

In IEEE 14th International Conference on Big Data Science and Engineering (**BigDataSE**), 2020.

https://ieeexplore.ieee.org/document/9343358

Referências auxiliares

https://arxiv.org/pdf/1804.07474.pdf

https://ieeexplore.ieee.org/document/9499122

https://ieeexplore.ieee.org/document/9832387

https://arxiv.org/abs/2207.00872

https://ieeexplore.ieee.org/abstract/document/9253545

https://opens3-lab.com/projects/iotdefender-iot-anomaly-detection/