Anomaly Detection in IoT Infrastructures using Federated Learning

*Jithin T Chandran, *Pranoy P Sundar, *Sidheeque Fasal CPA, *Jaseem CK, **Sruthy Manmadhan

*UG Student, **Assistant Professor, Department of Computer Science and Engineering

NSS College of Engineering, Palakkad

Introduction

Anomaly Detection in IoT

- IOT devices are used widely these days and it propagates a large amount of data.
- IOT devices use wireless medium to broadcast data which makes them an easier target for an attack.
- There are many conventional and novel attacks and anomalies seen in IOT infrastructure[1].
- The data communicated in the IoT systems are shared for improving user interactions, easy processing and for inference also by third party organisations.
- Most of the data in the IoT system are sensitive and confidential and the misuse or disclosure can have serious consequences.
- Attack in IoT system expands over a larger area and has devastating effects on IoT sites. Hence a secured IOT infrastructure is necessary for the protection from cybercrimes.

Problem Statement

A anomaly detection system is necessary in the IoT platform for the protection from cybercrime. The traditional Machine Learning approaches doesn't cover privacy in it's full sense. A novel method of Federated Learning is used to approach this issue. It is a family of Machine Learning algorithms introduced by Google in 2016 for the sole reason of privacy preserving Artificial Intelligence.

Objective

- To detect anomalies and attacks in a IOT network and to classify them into known cyber attack categories using supervised machine learning.
- To develop a smart, secure and reliable IoT based infrastructure which can detect its vulnerability and keep all its data safe and secure.

Approach of Federated Learning

Federated Learning Concept

- This technique is most popular with systems deployed at scale. Federated learning[2] is used where the data should not be shared with the cloud but requires information and analysis from the data at hand.
- This method is used where confidential data are involved and is ideal for edge devices[3].
- Federated learning trains an algorithm across multiple decentralized edge devices or servers holding local data samples, without exchanging their data samples.

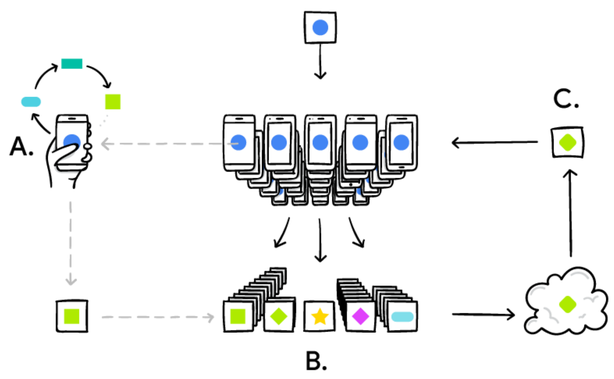

Figure 1: Federated Learning Approach[2]

Working of FL

- Machine learning models are trained on your edge devices and then the stuff which can be either weights in neural networks or other types of machine learning models, are the only ones sent to the central server.
- The central server averages those stuff which it receives from connected edge devices and then uses them to train its central machine learning model.
- After it undergoes training up to some epochs, that central machine learning model is distributed back to the devices to be used for predictive purposes or for further training.

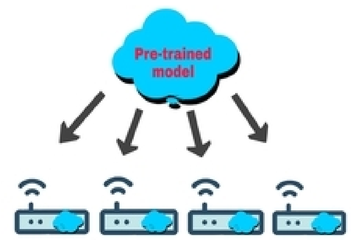

Figure 2: Initial pre-trained model send to edge

Figure 3: Updates model using local data

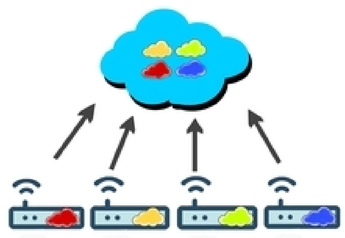

Figure 4: Updates sent from federated to server

Figure 5: Server aggregates the model

Benefits of FL

Benefits of using Federated Learning can be enlisted under these points:

- Decentralized learning
- Secure computing
- Preserve privacy

Advantages

- Federated learning is based on the data parallelism model. The training data is split between the copies of the network in the data parallelism model, such that each copy is trained on an independent section of data.
- Once all copies of the model are trained, the resulting weights are aggregated at a central repository. This is usually accomplished by averaging the weights of the independently trained models.
- The trained new network weights are sent to the center for aggregation, and the resulting model is then distributed or shared to the edge devices.
- In this way, the model in each edge device will reflect the patterns of the whole data without sending them all to the central server.
- This can reduce bandwidth requirements and can occur when a connection to the central server is available.
- The main advantage of introducing federated approaches to machine learning is to ensure data privacy or data secrecy. Indeed, no local data is uploaded externally, concatenated or exchanged in this method.
- In this way, Federated learning[4] is ideal for anomaly detection in the IoT platform.

Experiments and Results

Dataset

The open source dataset was collected from UCI database provided by KDD cup 1999[5]. This is the data set used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99 The Fifth International Conference on Knowledge Discovery and Data Mining. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated. In the dataset, there are 48,98,431 samples and 42 features. But for the experiment, we are taking only taking 1,00,000 samples. There exists 20 distinct type of threats in the dataset. The attacks fall into 4 main categories.

- DOS: denial-of-service like syn flood.
- R2L: unauthorized access from a remote machine, like guessing password.
- U2R: unauthorized access to local superuser (root) privileges like various buffer overflow attacks.
- probing: surveillance and other probing like port scanning.

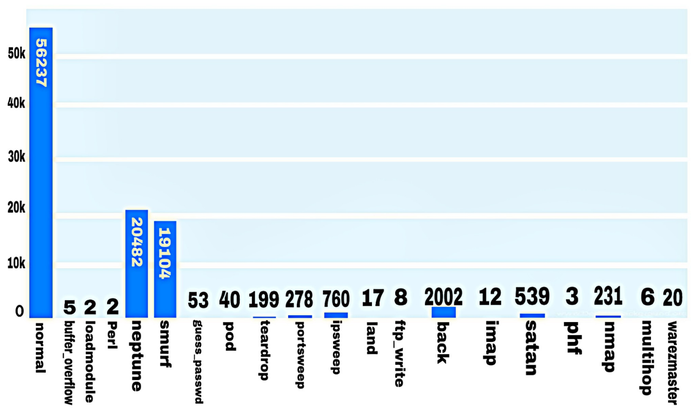

Figure 6: Distribution of attacks in the dataset[5]

Implementation

Steps involved in the Federated Learning approach:

- The data is divided equally into two and are given to two IoT devices. A local model is trained using corresponding training dataset in both the devices.
- A global model is developed aggregating the local models. The data is never seen as it is, only the pointers to the actual data are used with the help of specific functions in pysyft.
- Once the global model is ready. It is fetched back to the local devices. This is similar to that of modeling using the whole dataset. But here none of the data in raw format are provided. Only the information required for the modeling are passed with each other.
- Model build using the whole datasets, model build at edge gateways with alloted splitted dataset and the model build using federated learning are compared to deduce conclusions.

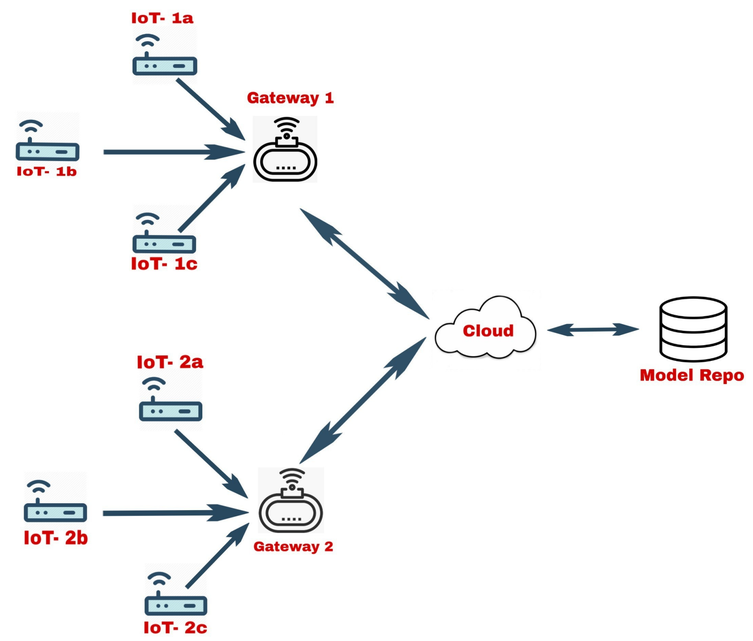

Figure 7: Proposed System Design

Tools Used

Pytorch

Torch is an open-source machine learning library, a scientific computing framework, and a script language.

Pysyft

PySyft is a Python library for secure and private Deep Learning. PySyft decouples private data from model training, using Federated Learning, Differential Privacy, and Encrypted Computation (like Multi-Party Computation (MPC) and Homomorphic Encryption (HE)) within the main Deep Learning frameworks like PyTorch and TensorFlow.

Model Comparison

.

Model	Training Set	Testing Set	True Positive	Accuracy
Edge 1	30000	20000	11235/20000	56%
Edge 2	30000	20000	15347/20000	77%
Full Dataset	60000	40000	38310/40000	96%
FL Model	60000	40000	38304/40000	96%

Result Analysis of the Experiment.

Inferences

- The model performs better with more data.
- The merged model and the federated model have similar accuracies.
- The federated model does not compromise accuracy or performance for the enhanced security.
- The load on central server can be reduced to a reasonable rate as the training data doesnt requires to be at the server.
- As the federated learning gives back the global model to the edges, the better performance of classification reflects alike in all the edge devices.
- This secures the data of the edge devices as they are only sending the weights of their local model.

Conclusions

- Federated Learning Model provides privacy without compromising accuracy of the model.
- Federated Learning is a privacy preserving approach that allows model to be trained on each edge device and is based on the data parallelism model.
- This distributed approach to anomaly detection can produce similar results to a non-distributed model. The distribution allows most of the computation to occur in parallel on the edge devices.
- Additionally, it can reduce the amount of data required to be transferred to the central server making it viable for situations when the edge devices have limited or inconsistent connections.

Future Works

- Secure aggregation can be done using a trusted third party worker which will enhance the privacy provided.
- Differential privacy can be introduced in the system to defend reverse engineering or reconstruction attacks.
- Federated Learning is relatively a new technology which is still in its early stages. Researches and experiments are taking place to make the best out of this approach.

References

- M. Hasan, Md. M. Islam and Md. I.I. Zarif et al, "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches," Internet of Things 7, 2019.
- H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson and Blaise Agera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," Google, 2016.
- Joseph Schneible and Alex Lu, Anomaly Detection on the Edge, Cyber Security and Trusted Computing, 2017.
- Tuhin Sharma and Bargava Subramanian, Anomaly Detection in Smart Buildings using Federated Learning, Oreilly AI London 2019, 17 Oct 2019.
- KDD CUP 1999 data [Online]. Available: https://kdd.ics.uci.edu/databases/kddcup99. [Accessed 24 February 2020]