ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

BÁO CÁO TỔNG KẾT
ĐỀ TÀI KHOA HỌC VÀ CÔNG NGHỆ SINH VIÊN NĂM 2021

Tên đề tài tiếng Việt:

HỆ THỐNG PHÁT HIỆN XÂM NHẬP CHO KIẾN TRÚC MẠNG EOT DỰA TRÊN MÔ HÌNH FEDERATED LEARNING

Tên đề tài tiếng Anh:

A FEDERATED LEARNING-BASED INTRUSION DETECTION SYSTEM FOR EDGE-OF-THINGS NETWORK

Khoa/ Bộ môn: Mạng máy tính và Truyền thông

Thời gian thực hiện: 06 tháng

Cán bộ hướng dẫn: ThS. Nguyễn Thanh Hoà

Tham gia thực hiện

TT	Họ và tên, MSSV	Chịu trách nhiệm	Điện thoại	Email
1.	Phạm Ngọc Tâm	Chủ nhiệm	0795541213	18521371@gm.uit.edu.vn
2.	Nguyễn Lê Thái Hoàng	Tham gia	0834971575	18520058@gm.uit.edu.vn
3.	Trần Quốc Khánh	Tham gia	0353178737	18520907@gm.uit.edu.vn

Thành phố Hồ Chí Minh – Tháng 12 /2021

ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Ngày nhận hồ sơ	
Mã số đề tài	
(Do CQ quản lý ghi)	

BÁO CÁO TỔNG KẾT

Tên đề tài tiếng Việt:

HỆ THỐNG PHÁT HIỆN XÂM NHẬP CHO KIẾN TRÚC MẠNG EOT DỰA TRÊN MÔ HÌNH FEDERATED LEARNING

Tên đề tài tiếng Anh:

A FEDERATED LEARNING-BASED INTRUSION DETECTION SYSTEM FOR EDGE-OF-THINGS NETWORK

Ngày ... tháng ..... năm ....
Cán bộ hướng dẫn
(Họ tên và chữ ký)

Nguyễn Thanh Hoà

Ngày ... tháng ..... năm ....
Sinh viên chủ nhiệm đề tài
(Họ tên và chữ ký)

Phạm Ngọc Tâm

THÔNG TIN KẾT QUẢ NGHIÊN CỨU

1. Thông tin chung:

- Tên đề tài: **Hệ thống phát hiện xâm nhập cho kiến trúc mạng EoT dựa trên mô hình Federated Learning**
- Chủ nhiệm: **Phạm Ngọc Tâm**
- Thành viên tham gia: **Nguyễn Lê Thái Hoàng, Trần Quốc Khánh**
- Cơ quan chủ trì: Trường Đại học Công nghệ Thông tin.
- Thời gian thực hiện: 06 tháng

2. Mục tiêu:

Trong thời đại công nghệ 4.0, các hệ thống Internet of Things (IoT) đang đóng vai trò thiết yếu trong việc cải thiện đời sống con người [1][2]. Bằng cách kết nối các thiết bị với nhau qua mạng Internet, IoT đã cung cấp nền tảng để triển khai các ứng dụng hữu ích như nhà thông minh, hệ thống giao thông thông minh, nông nghiệp thông minh [3]. Tuy nhiên, các hệ thống IoT truyền thống kết hợp cùng mô hình điện toán đám mây (Cloud computing) để gửi dữ liệu về các máy chủ trung tâm từ xa dẫn đến khó đáp ứng các nhu cầu trong tình hình mới. Một trong các thách thức quan trọng là độ trễ cao do khoảng cách địa lý giữa các thiết bị IoT đến các máy chủ trung tâm tại cloud rất lớn [4]. Đồng thời, các cơ sở hạ tầng mạng của các nhà cung cấp cloud computing hiện tại sẽ không thể đáp ứng được việc truyền tải lượng rất lớn dữ liệu do các thiết bị IoT liên tục tạo ra mỗi ngày [5]. Vì thế, kiến trúc Edge-of-things (EoT) (hay Edge computing – điện toán cận biên) đã được đề xuất với ý tưởng chính là đưa các máy chủ tính toán tại đám mây gần với các thiết bị đầu cuối hơn, nơi dữ liệu được tạo ra từ các thiết bị IoT để xử lí dễ dàng và nhanh chóng [6].

Mặc dù vậy, kiến trúc mạng EoT vẫn còn nhiều rủi ro bảo mật tương tự như mô hình mạng IoT thông thường [6]. Kẻ tấn công có thể kiểm soát các thiết bị IoT hoặc máy chủ cận biên (edge server, gateway) mà chúng muốn qua các

kết nối mạng trong quá trình trao đổi dữ liệu [4]. Mặt khác, nhiều nhà sản xuất phát hành các thiết bị IoT ngày càng nhanh với tâm lý "vội vàng tung ra thị trường" cũng như không cập nhật các phần mềm thường xuyên dẫn đến các thiết bị này không được trang bị đầy đủ các tính năng bảo mật và chứa nhiều lỗ hổng có thể bị khai thác. Những hạn chế trên tạo ra nhiều cơ hội cho những kẻ xâm nhập tấn công vào hệ thống mạng IoT theo nhiều phương thức khác nhau [5]. Thực tế, nhiều cuộc tấn công quy mô lớn nhắm vào các thiết bị IoT đã xảy ra, tiêu biểu như Mirai botnet vào năm 2016 [5], [7]. Theo báo cáo của Gemalto - một công ty dẫn đầu thế giới trong lĩnh vực bảo mật, hiện tại có đến 52% doanh nghiệp vẫn không thể phát hiện các thiết bị của họ có bị tấn công hay không [4]. Với những lý do trên, việc phát triển các giải pháp phát hiện xâm nhập (IDS – Intrusion Detection System) là điều kiện tiên quyết để giảm thiểu các rủi ro bảo mật và chủ động triển khai các biện pháp ngăn chặn kịp thời [6].

Nhìn chung, hệ thống IDS có thể được phân thành 2 loại chính là Signature-based và Anomaly-based [8][9]. Signature-based IDS so sánh các dấu hiệu đặc trưng (signature) của các loại tấn công đã biết với các sự kiện đang diễn ra trên hệ thống để xác định và đưa ra cảnh báo. Nhược điểm của hệ thống này là không thể phát hiện các loại tấn công mới hoặc biến thể của các loại tấn công đã có cho đến khi các nhà cung cấp IDS cập nhật signature mới [8]. Ngược lại, anomaly-based IDS có khả năng phát hiện các loại tấn công mới bằng cách so sánh các hoạt động hiện tại với các trạng thái bình thường đã được theo dõi trong một khoảng thời gian nhất định [9]. Đây cũng là cơ chế hoạt động của các IDS sử dụng các thuật toán Machine Learning (ML) [4]. Tuy nhiên, cách tiếp cận này gặp nhiều khó khăn khi triển khai trong thực tế, cụ thể là trong bối cảnh môi trường mạng IoT quy mô lớn. Những thách thức cụ thể như: 1) sự đa dạng về đặc điểm của các loại thiết bị IoT và khả năng xử lý của chúng thường rất hạn chế; 2) với mô hình ML truyền thống, lượng dữ liệu khổng lồ được tạo ra bởi các thiết bị đầu cuối cần được thu thập và truyền tải đến trung tâm dữ liệu có khoảng cách địa lí xa; 3) việc gửi tất cả dữ liệu qua mạng gây ra chi phí truyền tải cao và chi phí truyền thông lớn, 4) người dùng ngày càng đặc biệt quan tâm đến tính riêng tư của các dữ liệu được gửi đi từ các thiết bị IoT [4]. Chính vì vậy, việc

xây dựng được một hệ thống ML phát hiện xâm nhập được huấn luyện dựa trên dữ liệu từ tất cả các loại thiết bị trong hệ thống mạng, phát hiện chính xác và nhanh chóng các loại tấn công, cũng như chi phí truyền tải thấp và đặc biệt hơn hết là đảm bảo quyền riêng tư dữ liệu đang là một thách thức lớn.

Để giải quyết những thách thức đó, mô hình học máy cộng tác (FL - Federated Learning) đã được đề xuất và trở thành ứng viên tiềm năng. Với ý tưởng áp dụng mô hình FL cho hệ thống IDS, quá trình huấn luyện và cải thiện mô hình cục bộ có thể được thực hiện thu thập từ các thiết bị IoT trên các máy chủ cận biên (edge server). Dữ liệu thô (raw data) sẽ không bao giờ được chia sẻ với các bên tham gia mà chỉ có các thông số cập nhật mô hình (model update) được gửi đến máy chủ trung tâm, nơi chịu trách nhiệm điều phối, tổng hợp các kết quả nhận được nhằm cải thiện mô hình toàn cục. Sau đó, máy chủ trung tâm sẽ gửi các mô hình cập nhật mới đến các máy chủ cận biên, giúp cho hệ thống IDS được cải thiện chính xác hơn [5][8][9].

Với những tiềm năng của việc áp dụng mô hình FL cho hệ thống IDS, nhóm tác giả quyết định thực hiện đề tài *"Hệ thống phát hiện xâm nhập cho kiến trúc mạng EoT dựa trên mô hình Federated Learning".* Cụ thể, nhóm tác giả sẽ hướng đến triển khai mô phỏng hệ thống mạng EoT và nghiên cứu, xây dựng giải pháp IDS dựa trên mô hình FL. Từ đó, nhóm tác giả sẽ chứng minh tính khả thi, đánh giá độ hiệu quả của hệ thống này so với các hệ thống IDS truyền thống khác.

3. Tính mới và sáng tạo:

Tính mới và sáng tạo ở đây đó là việc ứng dụng mô hình học máy mới – Học hợp tác (Federated Learning) trong bài toán xây dựng hệ thống phát hiện và ngăn ngừa xâm nhập IDS trong ngữ cảnh IoT. Bốn thách thức đặt ra cho bài toán trên gồm: 1) sự đa dạng về đặc điểm của các loại thiết bị IoT và khả năng xử lý của chúng thường rất hạn chế; 2) với mô hình ML truyền thống, lượng dữ liệu khổng lồ được tạo ra bởi các thiết bị đầu cuối cần được thu thập và truyền tải đến trung tâm dữ liệu có khoảng cách địa lí xa; 3) việc gửi tất cả dữ liệu qua mạng gây ra chi phí truyền tải cao và chi phí truyền thông lớn, 4) người dùng ngày càng đặc

biệt quan tâm đến tính riêng tư của các dữ liệu được gửi đi từ các thiết bị IoT [4]. Nhóm tác giả tận dụng các ưu điểm và đặc điểm của mô hình học hợp tác để giải quyết bốn thách thức đặt ra: 1) việc học hợp tác từ nhiều nguồn giúp cho mẫu dữ liệu huấn luyện rất đa dạng, 2) trong học hợp tác các thiết bị đầu cuối chỉ gửi các kết quả đến máy chủ giúp cải thiện tốc độ và, 3) và băng thông đường truyền thấp, tiết kiệm, 4) bởi vì dữ liệu người dùng sẽ không được truyền trực tiếp qua mạng nên đảm bảo về bảo mật thông tin, tính riêng tư.

4. Tóm tắt kết quả nghiên cứu:

Sau quá trình thực hiện, nhóm tác giả đã thực hiện thành công những mục tiêu đã đưa ra là: nghiên cứu các lý thuyết liên quan và cũng đã thiết kế xây dựng một hệ thống hoàn chỉnh đầy đủ các chức năng đã đặt ra. Đồng thời nhóm tác giả cũng đã chứng minh được tính khả thi và đánh giá hiệu năng của hệ thống.

4.1. Nghiên cứu cơ sở lý thuyết và khảo sát các nghiên cứu liên quan

4.1.1. Máy học phân tán (Distribution Machine Learning)

Máy học phân tán đề cập đến các thuật toán và hệ thống máy học đa nốt được thiết kế để cải thiện hiệu suất, tăng độ chính xác và chia tỷ lệ thành kích thước dữ liệu đầu vào lớn hơn. Việc tăng kích thước dữ liệu đầu vào cho nhiều thuật toán có thể làm giảm đáng kể lỗi học tập (learning error) và thường có thể hiệu quả hơn so với với các phương pháp phức tạp hơn [10].

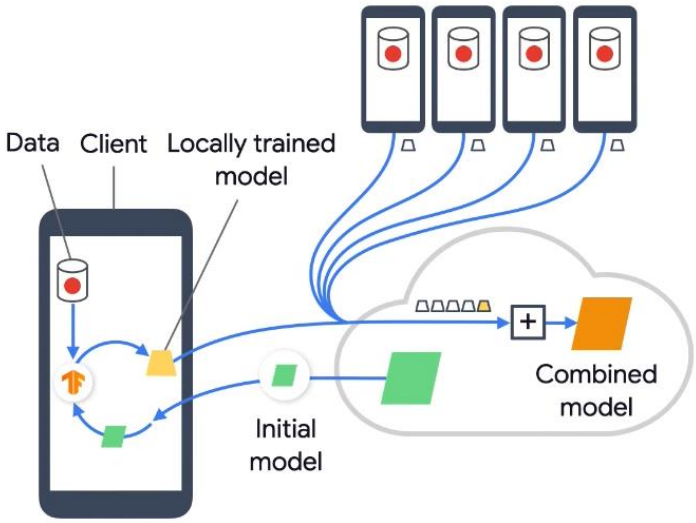
Học máy phân tán cho phép các công ty, nhà nghiên cứu và một cá nhân nào đó đưa ra đưa ra kết luận có nghĩa và hợp lý từ lượng lớn dữ liệu. Hệ thống thực hiện các tác vụ học máy trong môi trường phân tán được chia thành ba loại chính: cơ sở dữ liệu, hệ thống chung và hệ thống được xây dựng theo mục đích sử dụng.

Mỗi loại hệ thống có những ưu điểm và nhược điểm khác nhau, tùy theo mục đích sử dụng để có những yêu cầu hiệu suất, kích thước dữ liệu đầu vào và quá trình triển khai khác nhau.

4.1.2. Máy học cộng tác (Federated Learning)

Thuật ngữ học cộng tác (Federated Learning) được giới thiệu vào năm 2016 bởi McMahan và cộng sự. “Chúng tôi gọi cách tiếp cận của mình là federated learning vì nhiệm vụ học tập được thực trên các thiết bị tham gia (mà chúng tôi gọi là clients) được điều phố bởi một máy chủ trung tâm.”

Federated learning là mô hình machine learning mà trong đó nhiều thiết bị (clients) cùng cộng tác để giải quyết một vấn đề machine learning, dưới sự điều phối của một server trung tâm hoặc nhà cung cấp dịch vụ. Dữ liệu thô (raw data) của mỗi thiết bị tham gia chỉ được lưu trữ cục bộ trên thiết bị đó mà không được chia sẻ hay chuyển đi, thay vào đó, chúng sẽ chia sẻ các cập nhật chứa các dữ liệu đã được tổng hợp nhằm phục vụ cho quá trình học của cả hệ thống. Mô hình này sẽ đảm bảo được tính riêng tư cho dữ liệu cá nhân của mỗi thiết bị tham gia vào hệ thống.

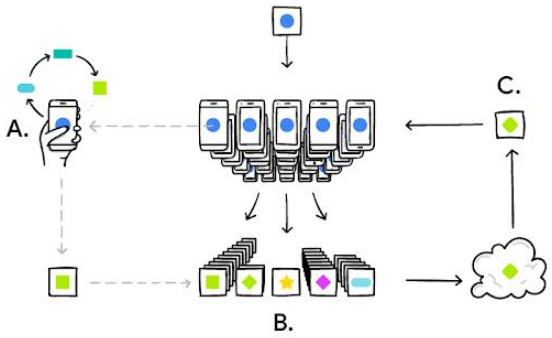


Hình 4.1.1. Federated Learning[1]

Quy trình hoạt động như sau: thiết bị tải xuống mô hình hiện tại, mô hình được cải thiện bằng cách học hỏi từ dữ liệu trên thiết bị, sau đó một bản cập nhật nhỏ được tóm tắt lại bởi các thay đổi. Bản cập nhật của mô hình được gửi đến server trung tâm, sử dụng giao thức mã hóa và được tính trung bình ngay lập tức với các bản cập nhật của người dùng khác để cải thiện mô hình được chia sẻ. Tất cả dữ liệu thô (draw data) dùng để huấn luyện vẫn còn

[1] https://ml.berkeley.edu/blog/posts/federated/

trên thiết bị và không có bản cập nhật riêng lẻ nào được lưu trữ tại server trung tâm.



Hình 4.1.2. Quy trình hoạt động của federated learning[2]

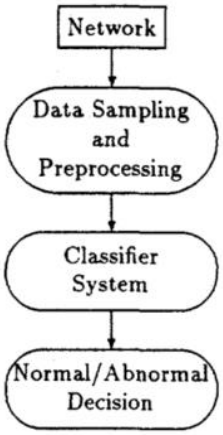
Federated leanring nhanh chóng tìm được chỗ đứng của mình, hòa nhập vào trong cuộc sống hiện đại của chúng ta, nhằm mục đích cải thiện bảo mật, xử lý dữ liệu để mang lại lợi ích cho nhiều lĩnh vực. Ngoài các lợi ích trên, FL cũng đem đến một số thách thức như hạn chế về phần cứng, phần mềm của các clients hay lo ngại về tính công bằng trong học máy [11].

4.1.3. Hệ thống tìm kiếm phát hiện xâm nhập

a. Khái niệm

Một hệ thống phát hiện (IDS) là một hệ thống quản lý bảo mật được sử dụng để phát hiện các xâm nhập, bất thường trong mạng và là một thiết yếu trong hệ thống bảo mật mạng ngày nay. IDS thường xuyên kiểm tra các lưu lượng gói tin ra và vào một mạng cụ thể để xác định xem mỗi gói tin có những đặc điểm bất thường hay xâm nhập hay không. Một IDS hiệu quả là khi có thể nhận biết được hầu hết các hoạt động xâm nhập, bất thường và tự động cảnh báo bằng cách ghi lại logs.

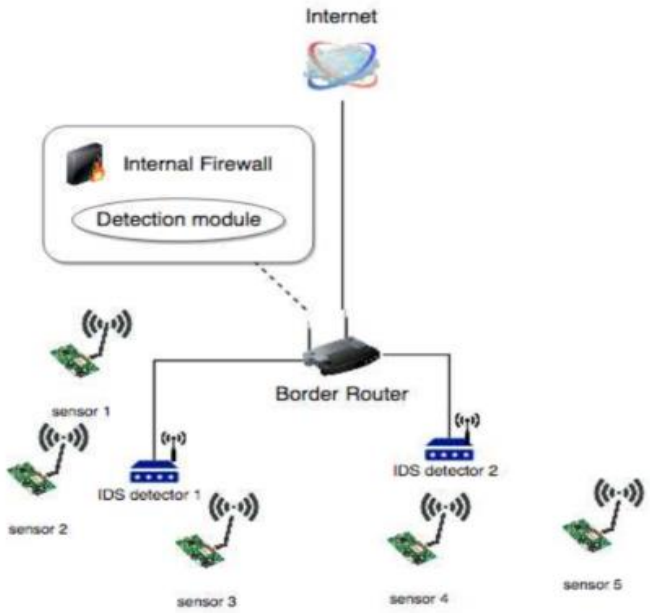
[2] https://ai.googleblog.com/2017/04/federated-learning-collaborative.html



Hình 4.1.3. Intrusion Detection System [12]

b. Phân loại

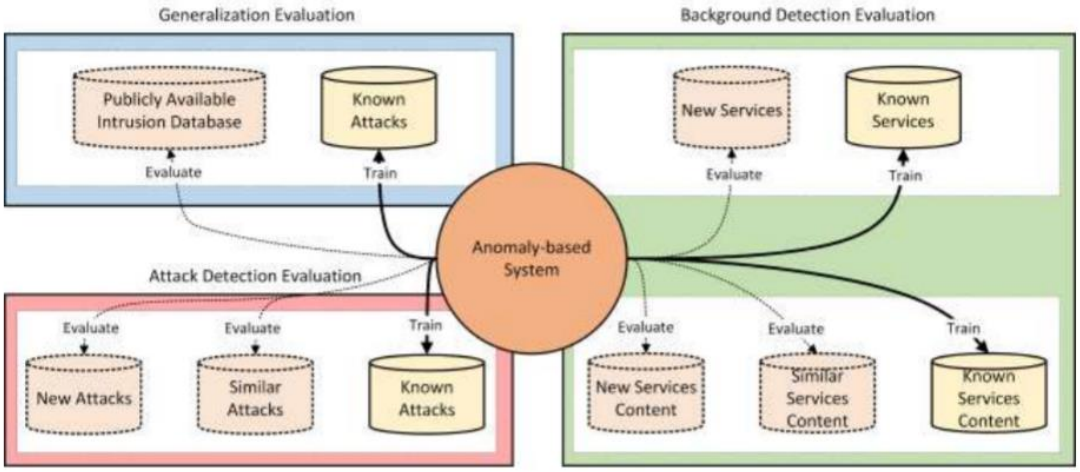
IDS có thể được chia làm 2 loại tùy thuộc vào kỹ thuật phát hiện chính: signature-based và anomaly-based. Một hệ thống IDS signature-based cần chỉ rõ đặc điểm nhận dạng của gói tin bất thường như thế nào, sau đó phát hiện chúng dựa trên các quy tắt đặt ra. Signature-based IDS có thể đạt được độ chính xác cao và có tỷ lệ phát hiện được các cuộc tấn công mới.



Hình 4.1.4. Signature-based IDS

Ngược lại, anomaly-based IDS sẽ được "học" các lưu lượng mạng bình thường, sau đó sẽ kiểm tra các hoạt động về sau có bất thường

hay không bằng cách so sánh với các hoạt động bình thường trước đó. Anomaly-based IDS không cần biết gì về đặc điểm của tấn công vì thế nó có thể phát hiện được những loại tấn công mới.



Hình 4.1.5. Anomaly-based IDS.

Hiện nay, kiến trúc mạng ngày càng trở nên phức tạp mang lại nhiều thách thức cho việc xây dựng các giải pháp IDS. Nhiều nghiên cứu các giải pháp gần đây cho IDS là sử dụng machine learning hay deep learning để cải thiện và tối ưu các kỹ thuật phát hiện. Các ví dụ điển hình như: support vector machine (SVM). Artificial neural networks (ANNs), genetic algorithms (Gas) đã có nhiều kết quả tốt trong lĩnh vực IDS.

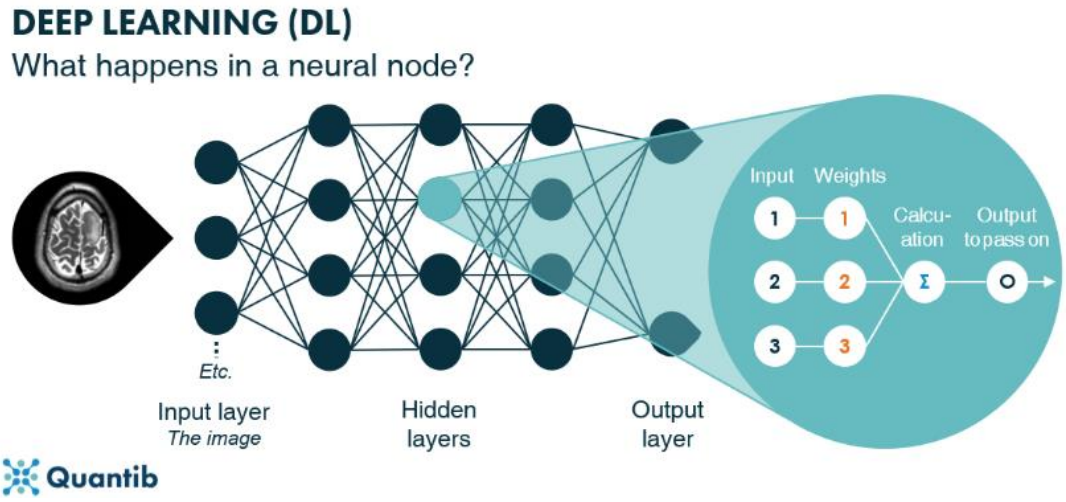
4.1.4. Học sâu (Deep Learning)

Học máy (Machine Learning) gây nên cơn sốt công nghệ trên toàn thế giới trong vài năm nay. Trong giới học thuật, mỗi năm có hàng ngàn bài báo khoa học về đề tài này. Trong công nghiệp từ các công ty lớn như Google, Facebook, Microsoft đến các công ty khởi nghiệp đều đầu tư vào machine learning. Hàng loạt các ứng dụng sử dụng machine learning ra đời trên mọi lĩnh vực của cuộc sống, từ khoa học máy tính đến những ít liên quan hơn như vật lý, hóa học, y học, chính trị. Tuy nhiên, khi dữ liệu càng khổng lồ thì hầu hết các phương pháp học máy truyền thống không thể giải quyết hiệu quả vấn đề phân loại dữ liệu khi đối mặt với môi trường ứng dụng mạng thực. Với sự phát triển năng động của các tập dữ liệu, nhiều nhiệm vụ phân

loại sẽ dẫn đến giảm độ chính xác của mô hình. Vào năm 2006, giáo sư Hinton đề xuất một lý thuyết mới là học sâu (Deep Learning), lý thuyết và công nghệ học sâu đã tạo ra một sự phát triển vượt bậc trong lĩnh vực máy học [13].

Deep Learning là một tập hợp con của machine learning, về cơ bản deep learning là một mạng nơ-ron với ba hoặc nhiều lớp (layer). Các mạng nơ-ron này cố gắng mô phỏng hành vi của bộ não con người, cho phép nó "học hỏi" từ một lượng lớn dữ liệu. Mặc dù một mạng nơ-ron với một lớp duy nhất vẫn có thể đưa ra dự đoán, tuy nhiên deep learning với nhiều nơ-ron chúng có thể giúp tối ưu hóa và nâng cao độ chính xác.

Với weights là hệ số và bias là độ lệch giá trị trung bình, mà mô hình dự đoán và giá trị thực tế của dữ liệu, mạng nơ-ron deep learning cố gắng bắt chước bộ não con người thông qua sự kết hợp của dữ liệu đầu vào, weights và bias. Các yếu tố này hoạt động cùng nhau để nhận dạng, phân loại và gắn nhãn các đối tượng trong dữ liệu. Mạng nơ-ron bao gồm nhiều lớp các nút được kết nối với nhau, mỗi lớp xây dựng trên kết quả của lớp trước để học hỏi và tối ưu hóa các dự đoán, phân loại. Quá trình này được gọi là quá trình truyền xuôi (forward propagation). Lớp đầu vào (input layer) là nơi mô hình deep learning thu thập dữ liệu để xử lý và lớp đầu ra (output layer) là nơi dự đoán hoặc phân loại cuối cùng được thực hiện. Một quá trình khác được gọi là truyền ngược (backpropagation) sử dụng thuật toán như gradient descent, để tính toán lỗi trong dự đoán và sau đó điều chỉnh weights và bias của hàm bằng cách di chuyển ngược qua các lớp trong quá trình đào tạo mô hình. Sự kết hợp của hai quá trình truyền xuôi và truyền ngược cho phép một mạng nơ-ron đưa ra dự đoán và thay đổi các thông số phù hợp. Theo thời gian, thuật toán trở nên chính xác hơn dần.



Hình 4.1.6. Quá trình học sâu[3]

4.1.5. Artificial Neural network (ANN)

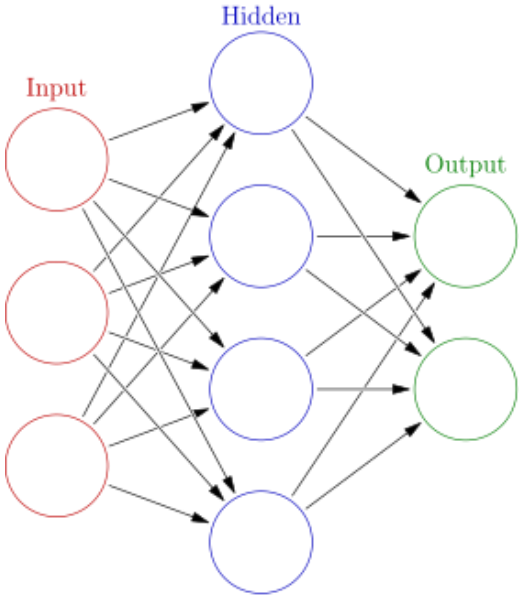
Mạng nơ-ron nhân tạo hay thường gọi ngắn gọn là mạng nơ-ron (Artificial Neural networl – ANN) là một mô hình toàn học hay mô hình tính toán được xây dựng dựa trên các mạng nơ-ron sinh học. Nó gồm có một nhóm các nơ-ron nhân tạo nối với nhau, và xử lý thông tin bằng cách truyền theo các kết nối và tính giá trị mới tại các nút. Trong nhiều trường hợp , mạng neural nhân tạo là một hệ thống tự thay đổi cấu trúc của mình dựa trên các thông tin bên ngoài hay bên trong trong quá trình học.

Mạng nơ-ron ANN được sử dụng rộng rãi [14] và chúng có thể cung cấp các dự đoán chính xác cao. Mạng nơ-ron ANN có những điểm vượt trội sau đây

- Học tập thích ứng (Adaptive learning): ANN tái tạo bộ não con người theo cách nó học cách thực hiện các nhiệm vụ trong khi học. Một chương trình bình thường không thể thích ứng với các loại đầu vào khác nhau
- Tư tổ chức (Self organization): ANN có thể tạo tổ chức của riêng mình trong khi học. Một chương trình bình thường được cố định cho nhiệm vụ của nó và sẽ không làm bất cứ điều gì khác ngoài những gì nó dự định sẽ làm

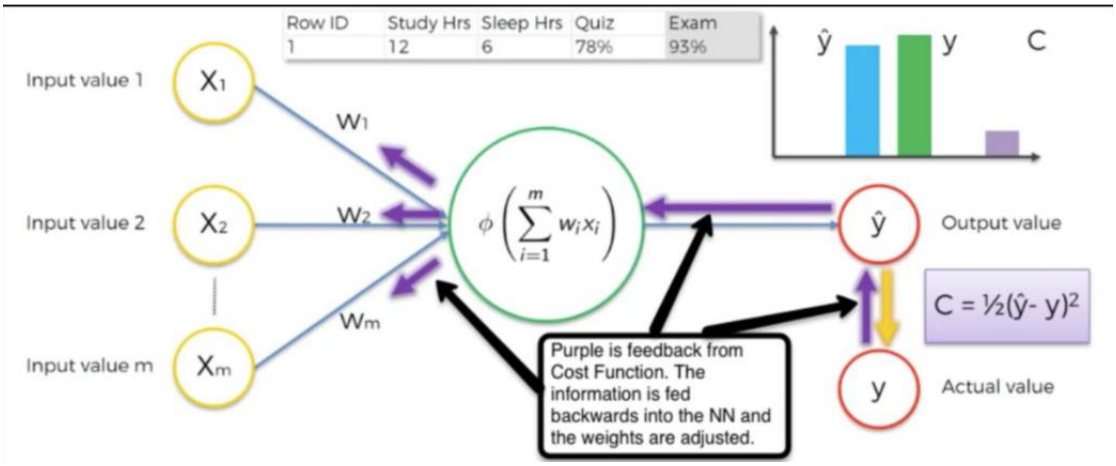
[3] https://www.quantib.com/blog/how-does-deep-learning-work-in-radiology

- Hoạt động song song (Parallel operation): ANN hoạt động song song giống như bộ não con người. Điều này khác với một chương trình máy tính hoạt động nối tiếp
- Khả năng chịu lỗi (Fault tolerance): Một trong những đặc tính thú vị nhất của mạng nơ-ron là khả năng hoạt động của chúng ngay cả trên cơ sở dữ liệu không đầy đủ (incomplete), nhiễu (noisy) và mờ (fuzzy). Một chương trình bình thường không thể xử lý dữ liệu không đầy đủ, không rõ ràng và sẽ ngừng hoạt động khi nó gặp phải dữ liệu sai nhỏ nhất.



Hình 4.1.7. Mô hình mạng nơ-ron nhân tạo[5]

[5]https://en.wikipedia.org/wiki/Artificial_neural_network



Hình 4.1.8. Cách hoạt động của mạng nơ-ron ANN.

Quá trình training của ANN có thể được tóm tắt như sau: hình trên mô tả chi tiết cách thức ANN hoạt động. Giả sử dựa vào các thông tin của mẫu dữ liệu bao gồm: thời gian học tập (study hours), thời gian ngủ (sleep hours), số điểm khi làm test (Quiz), mô hình sẽ phân tích cho ra dự đoán số điểm cuối cùng sau khi kiểm tra của học sinh này là bao nhiêu. Đầu vào của mô hình sẽ là các thông tin dữ liệu x_1, x_2,…, x_m, được đưa vào neuron lưu trữ các trọng số Weight w_1, w_2,… w_m nơ-ron này sẽ thực hiện tính biểu thức $w_1x_1 + w_2x_2$,… + w_mx_m sau đó giá trị được đưa vào hàm khích hoạt để đưa ra khoảng dữ liệu mong muốn và so sánh với output mong muốn (thực tế). Hình trên mô tả kết quả dự đoán và kết quả mong muốn sẽ được tính độ chênh lệch bằng hàm loss function sau đó mô hình sẽ điều chỉnh các trọng số sao cho đạt được giá trị loss nhỏ nhất. Thuật toán điều chỉnh Weights của mô hình cụ thể sẽ tính đạo hàm của Weights so với hàm loss function (độ dốc) và giảm Weights theo độ dốc đó với bước nhảy (learning rate). Một số thuật toán phổ biến được áp dụng như Gradient descent (GD), Stochastic Gradient Descent (SGD).

Hình 4.1.9. Mô tả quá trình học của mạng nơ-ron ANN

4.1.6. Các thư viện và nền tảng sử dụng

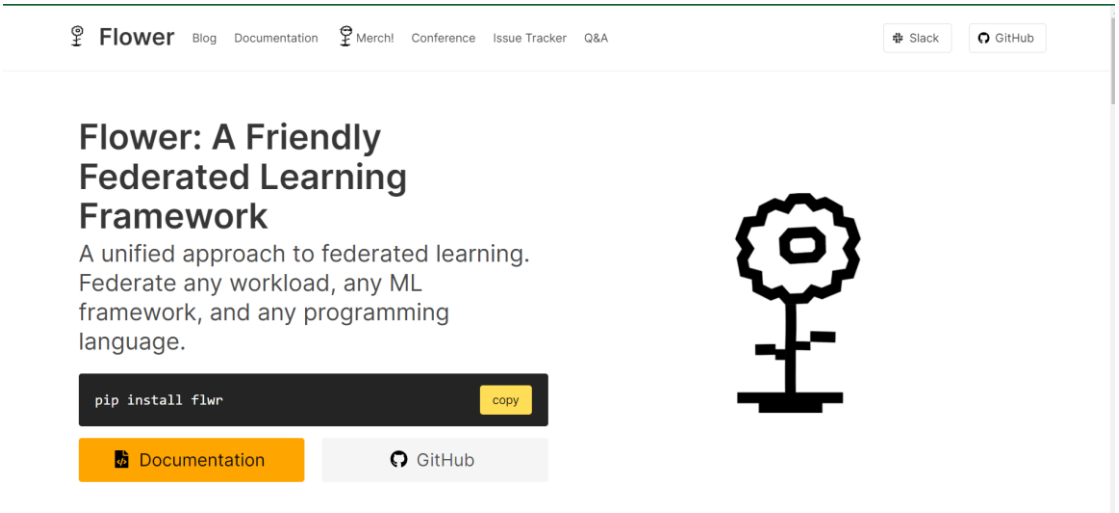
4.1.6.1. Flower

Flower là một dự án mã nguồn mở được xây dựng và duy trì liên tục bởi bởi các cộng tác viên của dự án. Flower là một framework Federated Learning (FL) mới được hợp nhất giữa hai quan điểm. Một là hỗ trợ các môi trường không đồng nhất bao gồm thiết bị di động và thiết bị cạnh (edge

device). Hai là mở rộng quy mô tới một số lượng lớn các client phân tán. Flower cung cấp các giải pháp để chuyển đổi khối lượng công việc với chi phí thấp, bất kể framework nào của Machine Learning được sử dụng, đồng thời cho phép các nghiên cứu thử nghiệm các phương pháp tiếp cận mới để nâng cao đến trình độ tiên tiến nhất [15]. Mục tiêu thiết kế và kiến trúc của Flower là sử dụng nó để đánh giá tác động của quy mô và tính không đồng nhất đối với các phương pháp FL phổ biến trong các thử nghiệm lên đến 1000 clients.

Một số ưu điểm của Flower như sau:

- Tương thích với nhiều framework ML như Keras và PyTorch.
- Hỗ trợ nhiều loại thiết bị và máy chủ bao gồm Android, Nvidia Jetson, iOS và Raspberry Pi.
- Platform độc lập.
- Framework dễ sử dụng
- Được thiết kết để làm việc với lượng lớn Client.



Hình 4.1.10. Giao diện phần mềm Flower

4.1.6.2. Tensorflow

TensorFlow là một nền tảng mã nguồn mở end-to-end dành cho học máy. Nó có một hệ sinh thái toàn diện, linh hoạt gồm các công cụ, thư viện và tài nguyên cộng đồng cho phép các nhà nghiên cứu đẩy

mạnh tính năng hiện đại trong ML và các nhà phát triển dễ dàng xây dựng và triển khai các ứng dụng hỗ trợ ML.

TensorFlow là một hệ thống học máy hoạt động ở quy mô lớn và trong môi trường không đồng nhất [16]. Tensor-Flow sử dụng đồ thị luồng dữ liệu (dataflow) để biểu diễn tính toán, trạng thái chia sẻ (shared state) và các hoạt động thay đổi trạng thái đó. TensorFlow cung cấp các API Python và C ++ một cách ổn định, cũng như API đảm bảo cho các ngôn ngữ khác.

4.1.7. Các nghiên cứu liên quan

Việc ứng dụng Federated Learning cho hệ thống IDS đã và đang được phát triển rộng rãi trên nhiều khía cạnh, ngữ cảnh trong thực tế như là một giải pháp tối ưu hơn các hệ thống IDS truyền thống. Federated Learning đảm bảo được tính riêng tư dữ liệu khi truyền qua Internet, bên cạnh đó cũng giải quyết được nhiều vấn đề khác.

4.1.7.1. Federated learning-based Intrusion detection for IoT networks

Trình bày một hệ thống phân tán tự học tự động để phát hiện các thiết bị IoT bị xâm phạm, xây dựng hiệu quả dựa trên các cấu hình giao tiếp dành riêng cho loại thiết bị mà không có sự can thiệp của con người cũng như dữ liệu được gắn nhãn sau đó được sử dụng để phát hiện các sai lệch bất thường trong hành vi giao tiếp của thiết bị, có khả năng gây ra bởi những kẻ thù độc hại. Hệ thống này sử dụng phương pháp học tập cộng tác để tổng hợp các cấu hình hành vi một cách hiệu quả và có thể đối phó với các cuộc tấn công mới xuất hiện và chưa được biết đến [5].

- Kết quả:

TABLE III: DL-based classifier accuracy results.

Label	Accuracy	Precision	Recall	FalseAlarm	F-Score
Normal	98.6	98	100	2.86	99
DoS	98.89	99	98	0.54	98
Probe	99.8	100	98	0.03	99
R2L	99.26	0	0	0	0
U2R	99.95	0	0	0	0

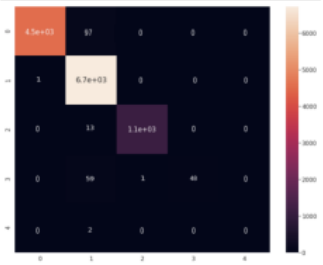
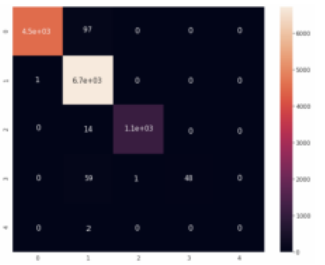


TABLE IV: FL-based classifier accuracy results.

Label	Accuracy	Precision	Recall	FalseAlarm	F-Score
Normal	98.62	98	100	2.93	99
DoS	99.13	100	98	0.006	99
Probe	99.9	100	99	0.02	100
R2L	99.59	100	0.45	0	0.62
U2R	99.95	0	0	0	0



Hình 4.1.11. Kết quả so sánh giữa centralized và federated learning [5].

Kết quả được thực nghiệm trên bộ dataset NSL-KDD trong 2 mô hình sử dụng ANN, bên trái là kết quả khi thực nghiệm với mô hình centralized, bên phải là kết quả thực nghiệm trong mô hình federated learning. Nhóm tác giả thấy rằng kết quả đạt được ở federated learning ngang bằng với kết quả khi thực nghiệm centralized xấp xỉ 98.6% [5].

Bên cạnh đó, một nghiên cứu khác với ý tưởng sử dụng mô hình deep learning khác là GRU cũng đạt hiệu quả khá tốt với False Positive Rate hầu như không có và True Positive Rate đạt 95.6% [5].

Type	Centralized learning	Federated learning		
		5 clients	9 clients	15 clients
FPR	0.00%	0.00%	0.00%	0.00%
TPR	95.60%	95.43%	95.01%	94.07%

Hình 4.1.12: Kết quả so sánh giữa centralized và federated learning [5].

Đối với giải pháp mô hình áp dụng Auto Encoder kết quả được với Federated learning trong datasets KDD cup 99 đạt 97.52% tối ưu hơn so với Centralized. NSL-KDD datasets độ chính xác trong federated learning cao hơn 3%. Cuối cùng với UNSW-NB15 độ chính xác tương đương với centralized xấp xỉ 95.6% [5].

	KDD		NSL-KDD		UNSW	
	ACC	TPR	ACC	TPR	ACC	TPR
Centrilized DL	97	92.52	90.86	77.15	95.67	78.33
Federated Learning (AE)	97.52	93.79	93.99	84.98	95.6	78.01

Hình 4.1.13. Kết quả so sánh giữa ccentralized và federated learning [5].

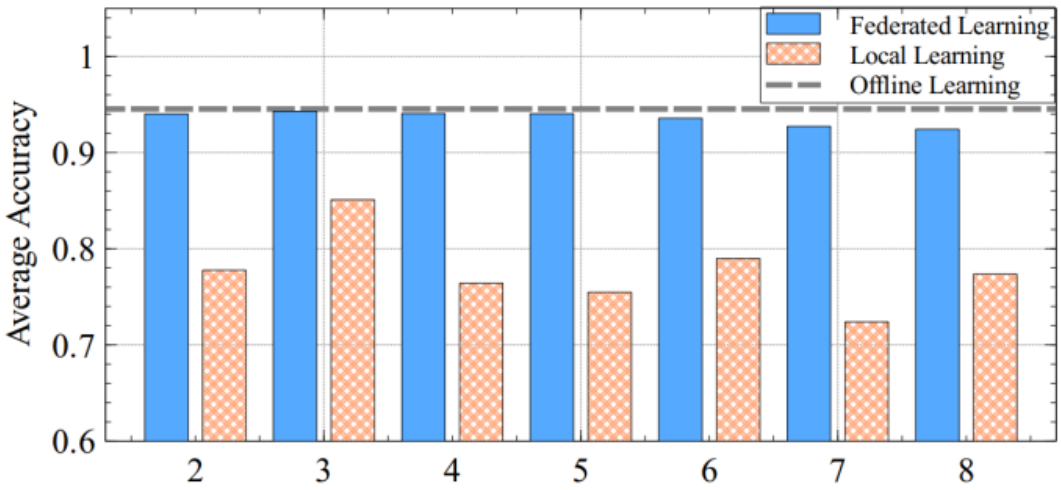
4.1.7.2. Federated learning-based Intrusion detection for Edge networks.

Trong ngữ cảnh Edge networks [17] đã chứng minh độ hiệu quả của federated learning so với giải pháp IDPS truyền thống với độ chính xác 99,84% với mô hình GRU kết hợp SVM (FedAVG) và 84,5% với mô hình ANN phân biệt nhị phân.

TABLE III
PERFORMANCE OF DIFFERENT METHODS

Method	IID			non-IID		
	Accuracy (↑)	FAR (↓)	F1score (↑)	Accuracy (↑)	FAR (↓)	F1score (↑)
GRU-SVM(local)	99.84%	0.42%	98.25%	98.84%	1.32%	97.65%
GRU-softmax(local)	98.94%	1.12%	97.96%	96.84%	3.15%	95.46%
ICNN(local)	95.84%	5.16%	94.55%	90.74%	9.89%	90.55%
VAE(local)	97.84%	3.28%	96.23%	95.84%	4.54%	94.23%
FedAVG(GRU-SVM)	99.84%	1.82%	97.44%	97.84%	2.08%	97.02%

Hình 4.1.14. Kết quả thuật toán FedAVG so sánh với một số mô hình khác [17].



Hình 4.1.15. Đồ thị kết quả so sánh độ chính xác giữa federated learning và local learning [17].

Nghiên cứu cho thấy sự hiệu quả của các phương pháp federated learning so với local learning (Distributed) nó tổng hợp được các đặc trưng của các node khác hau do đó độ chính xác được cải thiện đáng kể, cao hơn so với local learning [17].

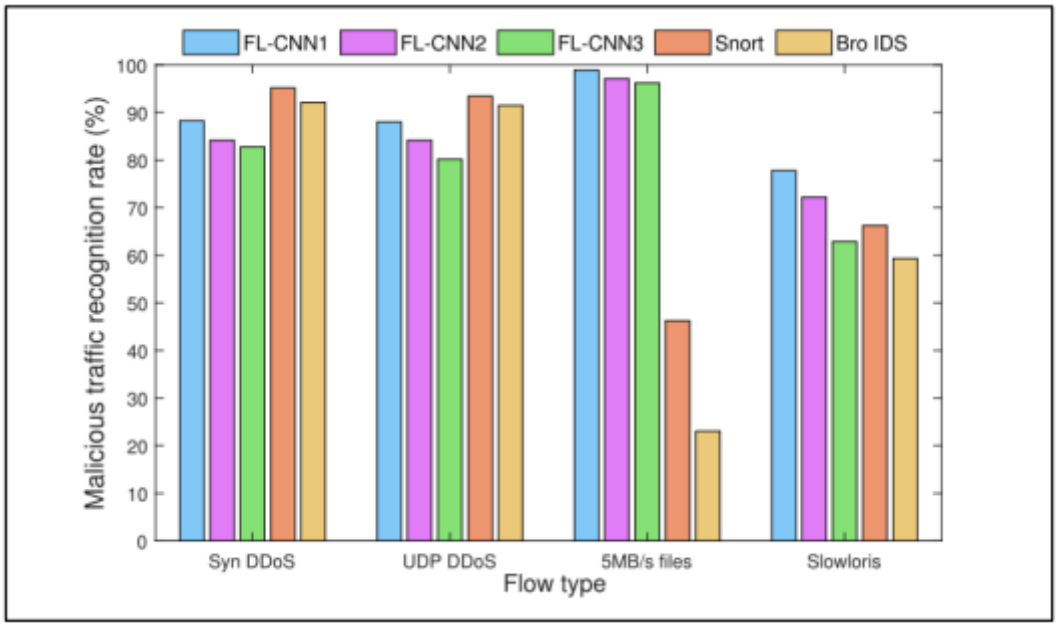
4.1.7.3. Distributed Network Intrusion Detection System in Satellite-Terrestrial Integrated Networks using Federated learning

Trong ngữ cảnh mạng tích hợp vệ tinh mặt đất [14], những thông tin gửi đến mạng vệ tinh hầu như là rất quan trọng vì thế việc đảm bảo tính riêng tư, bí mật dữ liệu là vô cùng cần thiết. Bên cạnh đó, mạng vệ tinh không tránh khỏi những cuộc tấn công mạng gây ra nhiều thiệt hại về tài sản. Đề tài đề xuất giải pháp sử dụng hệ thống deep learning-based NIDS để phát hiện và ngăn chặn các cuộc tấn công đồng thời áp dụng federated learning để đảm bảo tính riêng tư của dữ liệu [18].

Giải pháp này giải quyết được những vấn đề trong mạng vệ tinh như sau:

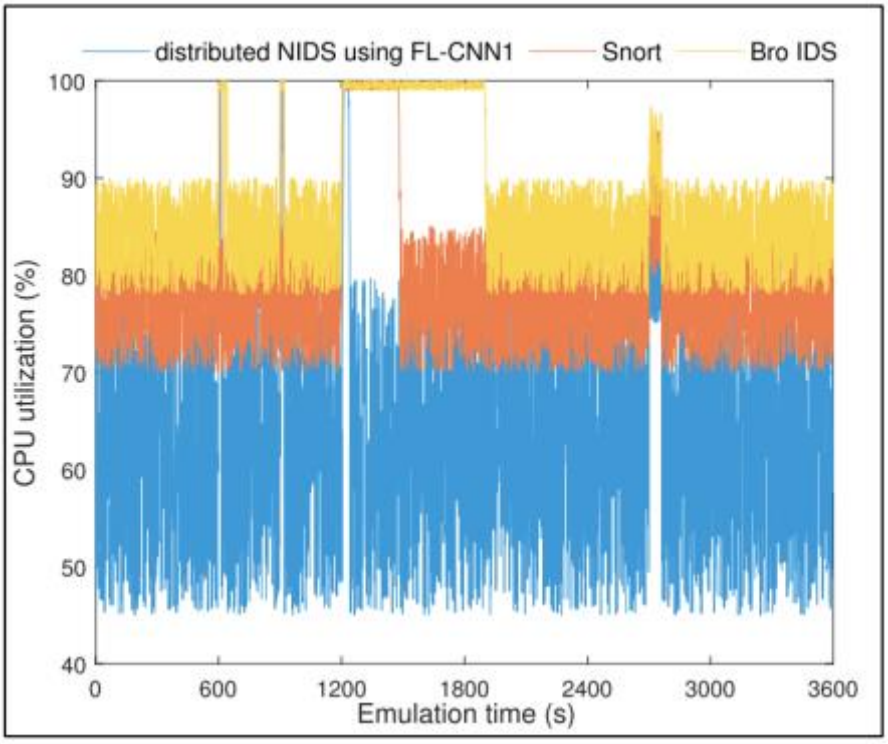
- Tài nguyên của mạng vệ tinh có giới hạn nên việc tính toán hạn chế
- Các cuộc tấn công DdoS gây ảnh hưởng rất lớn
- Thông tin truyền tải lên mạng vệ tinh rất quan trọng
- Trong federated learning, mỗi một hệ thống NIDS chỉ training một số ít dữ liệu -> việc tính toán ít. Sử dụng NIDS ứng dụng deep learning phát hiện được các cuộc tấn công mới, tinh vi hơn. Dữ liệu gốc không truyền qua internet nên đảm bảo được tính riêng tư

Kết quả đạt được:



Hình 4.1.16. Tỉ lệ nhận dạng lưu lượng truy cập độc hại của năm NIDSs

➔ **Hiệu quả hơn so với các IDS truyền thống snort [14].**



Hình 4.1.17: Biểu đồ sử dụng CPU của Edge Router 1.

4.1.7.4. Federated Deep learning for Intrusion Detection in Industrial Cyber-Physical Systems

Industrial Cyber-Physical Systems (CPSs) hay còn gọi là các hệ thống vật lí mạng công nghiệp bao gồm trong đó là các mạng thông minh và các công nghệ tính toán hiện đại như: 5G, Software-defined Network (SDN), cloud computing, Artificial intelligent (AI),… Việc kết hợp các công nghệ hiện đại lại với nhau sẽ thừa hưởng các lỗ hổng hay các cuộc tính công trên chính công nghệ đó dẫn đến các loại tấn công sẽ đa dạng và tinh vi hơn rất nhiều. Các giải pháp IDS dựa trên deep learning được áp dụng để có thể phát hiện các loại tấn công mới nhưng đòi hỏi phải cần 1 lượng các mẫu tấn công chất lượng trong ngữ cảnh Industrial CPSs. Tuy nhiên là một intrudial CPS có rất ít mẫu dữ liệu tấn công nên không thể triển khai deep learning IDS. Hơn nữa, những người sở hữu Indutrial CPS sẽ không sẵn sàng chia sẽ dữ liệu cho một bên thứ ba bởi vì tính nhạy cảm, riêng tư của dữ liệu [19]. Trong nghiên cứu, tác giả đã đề xuất sử dụng federated learning để có thể tận dụng được dữ liệu của các chủ sở hữu CPS để triển khai training mô hình deep learning mà không cần quan tâm đến việc dữ liệu nhạy cảm được chia sẻ. Mô hình deep learning sử dụng CNN kết hợp GRU.

Kết quả đạt được:

Type of cyber threats	Local model			The proposed DeepFed			Ideal model		
	Precision	*Recall*	*F-score*	*Precision*	*Recall*	*F-score*	*Precision*	*Recall*	*F-score*
Naive malicious response injection attack	0.9909	0.9009	0.9438	0.9562	0.9476	0.9519	1.0000	0.9024	0.9487
Complex malicious response injection attack	0.9550	0.9838	0.9691	0.9904	0.9997	0.9950	0.9917	0.9997	0.9957
Malicious state command injection attack	0.9932	0.9359	0.9637	0.9932	0.9359	0.9637	0.9932	0.9359	0.9637
Malicious parameter command injection attack	0.9792	0.9856	0.9824	0.9792	0.9856	0.9824	0.9792	0.9856	0.9824
Malicious function command injection attack	1.0000	0.9478	0.9732	1.0000	0.9478	0.9732	1.0000	0.9478	0.9732
Denial-of-service attack	0.9955	0.9771	0.9862	0.9945	0.9864	0.9904	0.9945	0.9864	0.9904
Reconnaissance attack	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

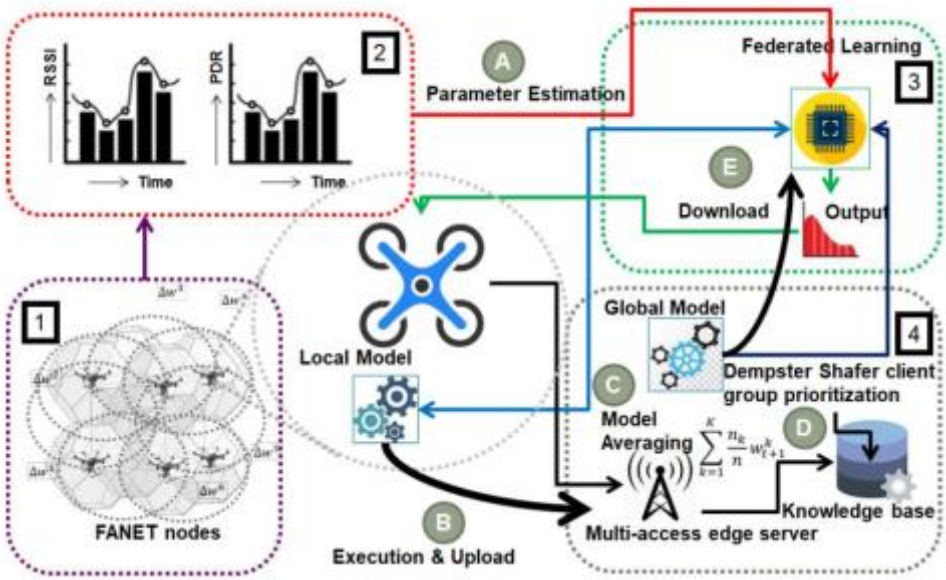
Hình 4.1.18. Kết quả số của các mô hình phát hiện tương đối với các vòng truyền thông có vòng truyền thông theo ba kịch bản khác nhau [13].

➔ Hiệu quả hơn khi sử dụng local model (centralized)

4.1.7.5. Federated learning-Based Cognitive Detection of Jamming Attack in Flying Ad-Hoc Network

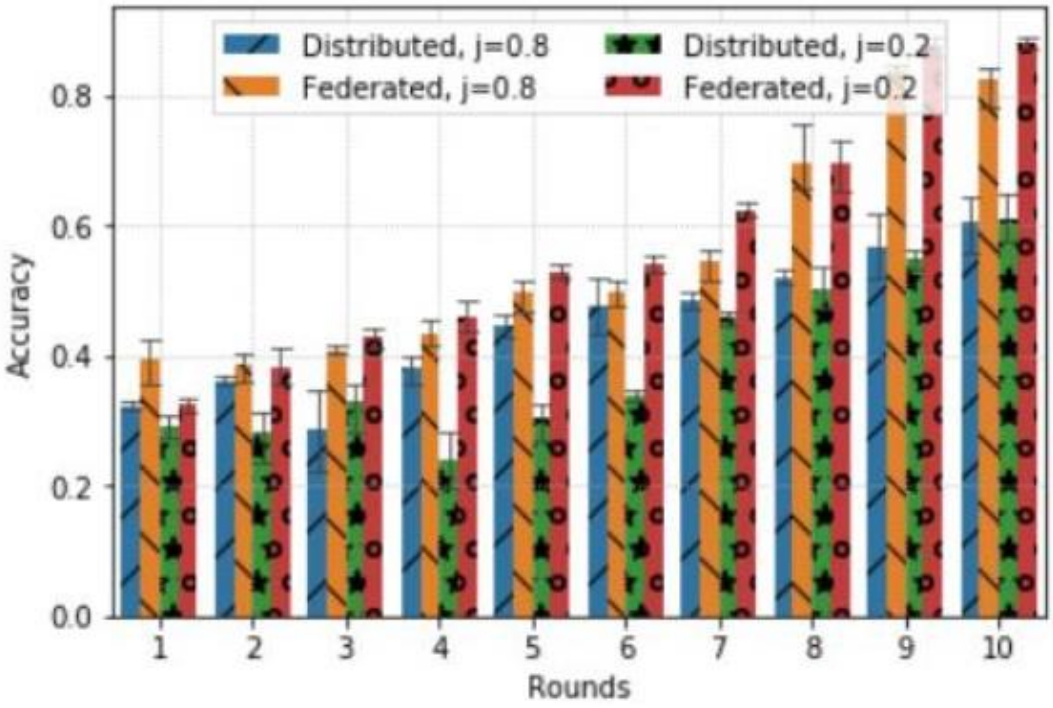
Flying Ad-Hoc Network (FANET) là hệ thống liên lạc phi tập trung hình thành bởi hệ thống máy bay không người lái (UAV) [19]. Trong FANET, các thiết bị UAV dễ bị tấn công bởi các cuộc tấn công độc hại khác nhau, trong đó có cuộc tấn công gây nhiễu. Các kẻ tấn công trong cuộc tấn công gây nhiễu làm gián đoạn liên lạc của mạng nạn nhân. Đầu tiên, do phạm vi giao tiếp khác nhau và các hạn chế về tiêu thụ điện năng, bất kỳ hệ thống phát hiện tập trung (centralized) nào cũng trở nên khó khăn trong FANET. Thứ hai, các giải pháp phi tập trung (decentralized) hiện có, không khả quan trong FANET vì dữ liệu của chúng đa dạng từ các môi trường mới, không gian mới, không phù hợp với các UAV có tính di động cao và không đồng nhất về mặt không gian trong FANET. Thứ ba, với một số lượng lớn UAV, mô hình trung tâm có thể cần phải chọn một nhóm nhỏ các khách hàng UAV để cung cấp các cập nhật mô hình kịp thời.

Trong nghiên cứu này federated learning được áp dụng vì nó giải quyết được vấn đề về sự khác nhau trong thộc tính dữ liệu bên cạnh việc cung cấp giải pháp giao tiếp hiệu quả, do đó làm cho nó trở thành một lựa chọ phù hợp cho FANET.



Hình 4.1.19. Mô hình federated learning phát hiện tấn công gây nhiễu trong FANET[20]

Trong nghiên cứu nhóm tác giả đã thực nghiệm trên tập dataset ns-3 simulated FANET jamming attack. Hình dưới đây là kết quả về độ chính xác của mô hình qua các round rong Distributed learning và Federated learning cho thấy kết quả của Federated learning tốt hơn.



Hình 4.1.20. Biểu đồ so sánh kết quả giữa Distributed learning và Federated learning [20].

4.2. Phân tích thiết kế và triển khai hệ thống IDS nghiên cứu

4.2.1. Giới thiệu mô hình phát hiện xâm nhập IDS (Khả năng, mục tiêu)

Mục tiêu: Phát hiện và ngăn ngừa các hành động phá hoại bảo mật hệ thống, hoặc những hành động trong tiến trình tấn công như dò tìm, quét các cổng như (ví dụ tên các cuộc tấn công…) trong ngữ cảnh IoT.

Khả năng: Hệ thống phát hiện xâm nhập nghiên cứu được đặt trên các thiết bị biên (Edge device) theo cấu hình học hợp tác (để nhận toàn bộ lưu lượng mạng trên mạng cần phát hiện.

4.2.1.1. Tiêu chí đánh giá hiệu quả của một hệ thống phát hiện xâm nhập

a. Accuracy

Accracy (độ chính xác) là phương pháp đánh giá mô hình machine learning đơn giản nhất. Cách đánh giá này đơn giản tính tỉ lệ giữa số điểm được dự đoán đúng và tổng số điểm trong tập dữ liệu kiểm thử.

Công thức tính accuracy:

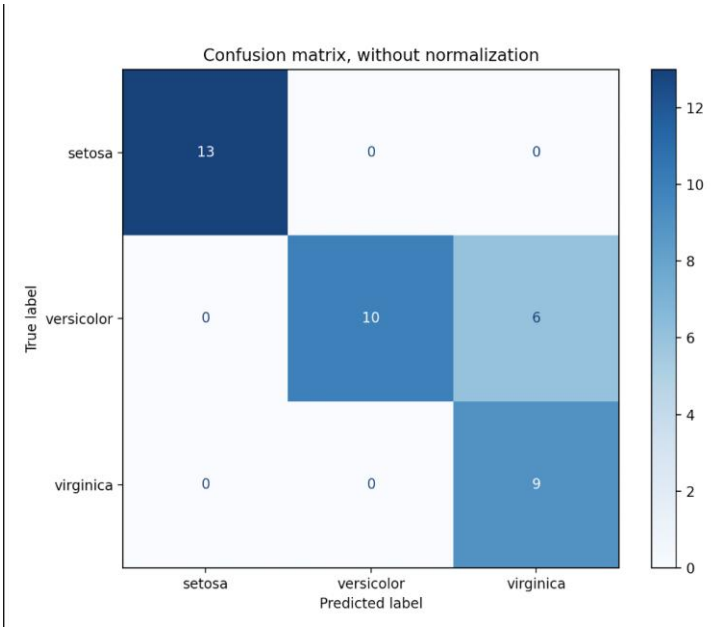
$$Accuracy = \frac{correct\ predictions}{all\ predictions}$$

Hình 4.2.1. Công thức tính Accuracy

b. Confusion Matrix

Cách tính sử dụng accuracy như ở trên chỉ cho biết được bao nhiêu phần trăm lượng dữ liệu được phân loại đúng mà không chỉ ra được cụ thể mỗi loại như thế nào, lớp nào được phân loại đúng nhiều nhất, và dữ liệu thuộc lớp nào thường bị phân loại nhầm vào lớp khác. Để có thể đánh giá được các giá trị này, sử dụng một ma trận được gọi là confusion matrix.

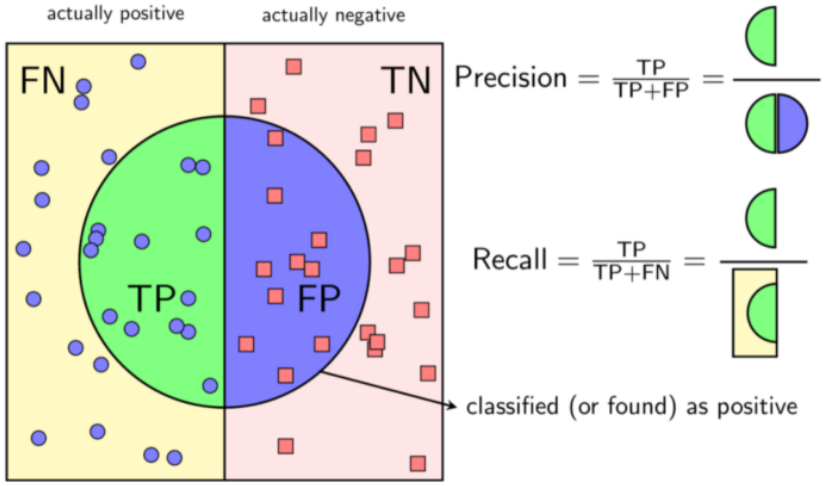
Về cơ bản, confusion matrix thể hiện có bao nhiêu điểm dữ liệu thực sự thuộc vào một class, và được dự đoán là rơi vào một class. Confusion matrix xem tỷ lệ báo động nhầm và báo động sai một cách rõ ràng hơn.



Hình 4.2.2. Confusion matrix

c. Precision và Recall

Với bài toán phân loại mà tập dữ liệu của các lớp là chênh lệch nhau rất nhiều, có một phép đo hiệu quả thường được sử dụng là Precision-Recall



Hình 4.2.3 Cách tính Precision và Recall

Với một cách xác định một lớp là positive, precision được định nghĩa là tỉ lệ số điểm true positive trong số những điểm được phân loại là positive (TP + FP). Recall được định nghĩa là tỉ lệ số điểm true positive trong số những điểm thực sự là positive (TP + FN). Precision và Recall đều là các số không âm nhỏ hơn hoặc bằng một.

Precision cao đồng nghĩa với việc độ chính xác của các điểm tìm được là cao. Recall đồng nghĩa với việc True Positive Rate cao, tức tỉ lệ bỏ sót các điểm thực sự positive là thấp.

d. F1 – score

F1-Score được định nghĩa như trung bình điều hòa (harmonic mean) giữa Precision và Recall.

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall}$$

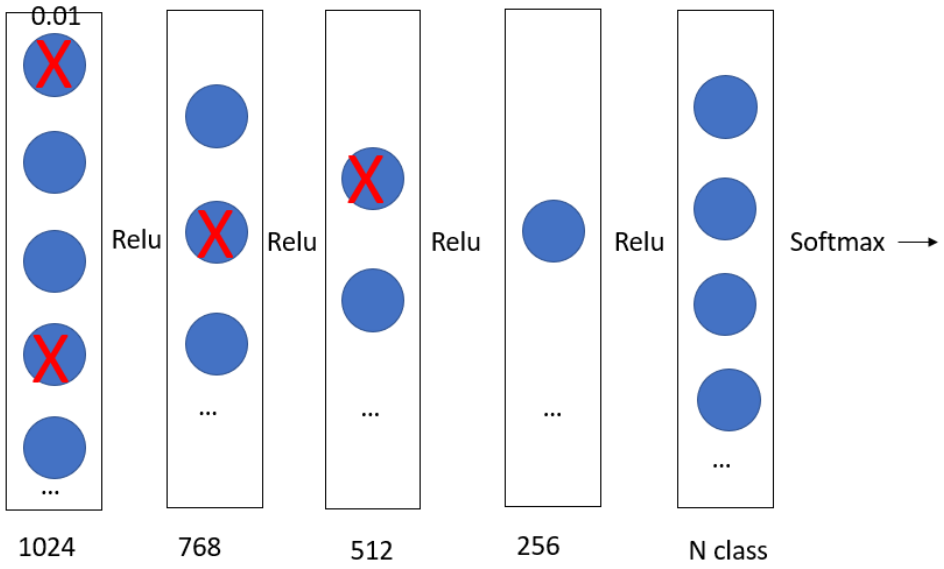
Hình 4.2.5: Công thức tính F1 – score.

4.2.2. Thiết kế và xây dựng mô hình phát hiện xâm nhập sử dụng phương pháp truyền thống

4.2.2.1. Xây dựng thuật toán và mô hình học sâu đề xuất

Thuật toán deep learning mà nhóm tác giả sử dụng là ANN (Artificial neural network). Lí do nhóm tác giả lựa chọn mô hình ANN bởi vì đây là dạng mô hình cơ bản nhất của deep learning, không yêu cầu thiết kế phức tạp và độ tin cậy cao, do đó mô hình ANN sẽ đảm bảo được sự ổn định trong huấn luyện để và dễ dàng so sánh hiệu suất giữa thiết kế truyền thống và học hợp tác.

Mô hình đề xuất có cấu hình như sau:



Hình 4.2.1 Mô hình ANN đề xuất

Mô hình ANN bao gồm 5 lớp: lớp đầu vào bao gồm 1024 nơ-ron, lớp 2 bao gồm 768 nơ-ron, lớp 3 bao gồm 512 nơ-ron, lớp 4 có 256 nơ-ron và lớp đầu ra có 1 nơ-ron. Giữa các lớp sử dụng hàm kích

hoạt (Activation function) là Relu, riêng lớp đầu ra sẽ có hàm kích hoạt là Softmax – chuyên dùng cho phân loại đa lớp (multi-class).

4.2.2.2. Kịch bản thực nghiệm

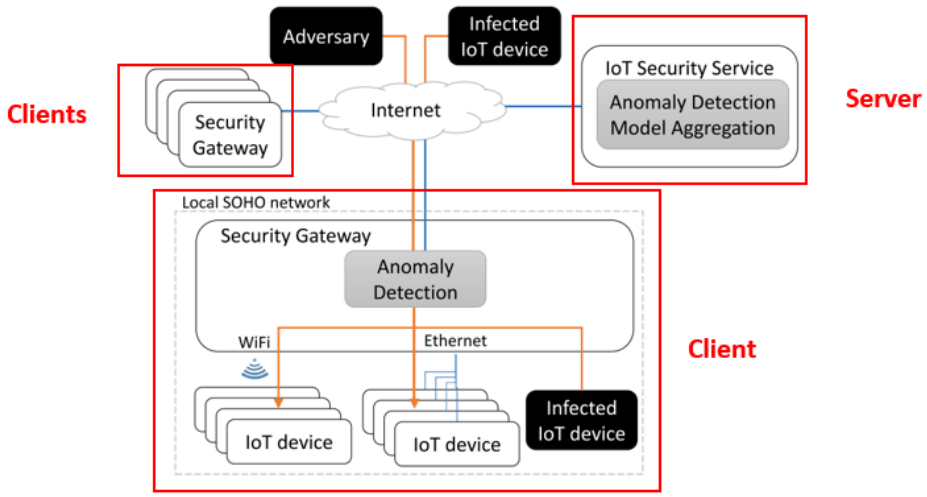
Trong thiết kế truyền thống, nhóm tác giả lựa chọn các thông số thực nghiệm:

- Số epochs: 50

4.2.3. Xây dựng mô hình phát hiện xâm nhập sử dụng phương pháp mô hình máy học hợp tác

4.2.3.1. Xây dựng thuật toán và mô hình học sâu đề xuất

Với mô hình IDS truyền thống, các thiết bị IoT khi muốn phát hiện tấn công trong thiết bị của mình hay không bắt buộc client phải gửi dữ liệu thuần lên phía server, server sẽ phân tích dự đoán và gửi kết quả về cho client. Tuy nhiên, việc gửi dữ liệu thô qua mạng không đảm bảo được tính riêng tư. Do đó, nhóm đề xuất giải pháp áp dụng federated learning cho IDS. Hình 3.1.2 là mô hình tổng quan nhóm đề xuất.



Hình 4.2.2 Mô hình federated learning cho IDS [5]

Trong mô hình này gồm 2 thành phần chính Security Gateway và IoT Security Service. Vai trò của Security Gateway là theo dõi các thiết bị

và được đặt một hệ thống IDS để phát hiện các bất thường trong mạng và các thiết bị có bị tấn công hay không. Bên cạnh đó, IoT Security Service được xem như là các nhà cung cấp dịch vụ đảm nhiệm việc tổng hợp các kết quả từ các hệ thống IDS ở các Security Gateway.

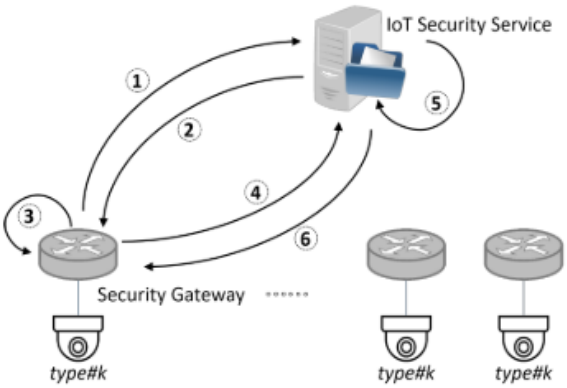
- **Security Gateway:**

Security Gateway có thể được xem như là cổng truy cập gateway trong hệ thống mạng thông thường, bất kì lưu lượng nào trong mạng ra internet đều đi qua nó. Do đó, Security Gateway có thể quan sát được mọi lưu lượng mạng của hệ thống cần bảo vệ giao tiếp với bên ngoài và nội bộ bên trong. Các lưu lượng mạng này sẽ được đưa vào một hệ thống phát hiện xâm nhập được đặt tại đây để phân tích và phát hiện bất thường trong mạng. Kết quả sau khi phân tích được sẽ được gửi cho IoT security service bên ngoài và đồng thời cũng nhận những dữ liệu cập nhật mới từ đó.

- **IoT Security Service:**

IoT Security Service đảm nhiệm việc hỗ trợ dịch vụ, cập nhật cho các Security Gateway. Khi hệ thống IDS ở Security Gateway học được những dữ liệu mới, IoT Security Service sẽ tổng hợp lại không chỉ 1 Security Gateway mà nhiều Security Gateway do đó có thể nói IoT Security Service có thể học được từ rất nhiều hệ thống IDS và tổng hợp được đa dạng các dữ liệu. Sau đó, IoT Security Service sẽ gửi các dữ liệu đã tổng hợp cho các Security Gateway. Từ đó, các Security Gate cũng được học từ nhiều Security Gateway khác. Đây là một ưu điểm rất lớn.

- **Cơ chế hoạt động:**



Hình 4.2.3 Tổng quan quá trình federated learning [4]

Cơ chế hoạt động của hệ thống được định nghĩa như sau:

- ➔ **Bước 1**: Khi Security Gateway muốn phân tích lưu lượng của thiết bị nào đó, nó sẽ yêu cầu IoT Security Service một mô hình GRU chung cho loại thiết bị này.
- ➔ **Bước 2:** Security Gateway nhận được mô hình GRU chung của IoT Security Service cung cấp
- ➔ **Bước 3:** Mô hình chung sẽ được train bằng dữ liệu lưu lượng được thu thập từ thiết bị đó
- ➔ **Bước 4:** Kết quả sau khi train bởi Security Gateway sẽ được gửi lên lại IoT Security Service
- ➔ **Bước 5:** IoT Security Service sẽ tổng hợp các kết quả từ Security Gateway và cải thiện mô hình chung.
- ➔ **Bước 6:** Cuối cùng mô hình chung sau khi được cập nhật sẽ được gửi lại cho Security Gateway để sử dụng.

4.2.3.2. Kịch bản thực nghiệm

Nhóm tác giả thiết kế một số cấu hình học hợp tác để làm thực nghiệm như sau:

- Số lượng client tham gia: 5
- Số lượng client được chọn: 2
- Số vòng: 100
- Epoch trong mỗi vòng tại client: 1

4.2.4. Môi trường và tập dữ liệu thực nghiệm

4.2.4.1. Môi trường triển khai thực nghiệm

Sử dụng hệ điều hành Ubuntu, RAM 16GB, ổ cứng HDD 60GB.

4.2.4.2. Tập dữ liệu thực nghiệm

- **BoT-IoT**[4]

Bộ dữ liệu BoT-IoT được tạo ra bằng cách thiết kế một môi trường mạng thực tế trong Phòng thí nghiệm phạm vi mạng của Trung tâm UNSW Canberra Cyber . Môi trường kết hợp giữa lưu lượng truy cập thông thường và botnet. Tệp nguồn của tập dữ liệu được cung cấp ở các định dạng khác nhau, bao gồm tệp pcap gốc, tệp argus đã tạo và tệp csv. Các tệp được tách biệt, dựa trên danh mục tấn công và danh mục phụ, để hỗ trợ tốt hơn trong quá trình gắn nhãn. Các tệp pcap được chụp có kích thước 69,3 GB, với hơn 72.000.000 bản ghi. Lưu lượng dòng đã trích xuất, ở định dạng csv có kích thước 16,7 GB. Bộ dữ liệu bao gồm các cuộc tấn công DDoS, DoS, OS và Service Scan, Keylogging…

Attack Type	Flow Count
BENIGN	9543
Service scanning	1463364
OS Fingerprinting	358275
DDoS TCP	19547603
DDoS UDP	18965106
DDoS HTTP	19771
DoS TCP	12315997
DoS UDP	20659491
DoS HTTP	29706
Keylogging	1469
Data theft	118
/	73370443

[4] https://research.unsw.edu.au/projects/bot-iot-dataset

Hình 4.2.4. Các loại tấn công trong datasets BoT-IoT

Tập dữ liệu bao gồm 2 tập tin:

- *UNSW_2018_IoT_Botnet_Final_10_best_Testing.csv (1)*
- *UNSW_2018_IoT_Botnet_Final_10_best_Training.csv (2)*

Tập huấn luyện (2) nhóm tác giả sử dụng chia thành 2 phần 20% và 80%, phần lớn sẽ dùng để huấn luyện mô hình và phần nhỏ dùng để đánh giá mô hình sau mỗi epoch. Trong học hợp tác, mỗi client chỉ tải lên ngẫu nhiên 25% từ tập huấn luyện để tránh trường hợp các client huấn luyện cùng một tập.

- **N-BaIoT**[5]

Bộ dữ liệu N-BaIoT là tập dữ liệu công khai được sử dụng rộng rãi để nghiên cứu về phát hiện bất thường của dữ liệu IoT. N-BaIoT nắm bắt luồng lưu lượng mạng từ 9 thiết bị IoT thương mại bị tấn công một cách chân thực bởi các hành vi xâm nhập mạng. Mỗi thiết bị IoT có 2 tập hợp con: một tập hợp lành tính chỉ chứa dữ liệu luồng mạng thông thường và một tập hợp con dữ liệu tấn công bao gồm hai cuộc tấn công phần mềm độc hại phổ biến, Mirai và BASHLITE, mỗi tập hợp chứa năm loại tấn công khác nhau như trong bảng …

Botnet	Attack	Explanation
Bashlite	Scan	Scans the network for vulnerable devices
	Junk	Sending spam data
	UDP	UDP flooding
	TCP	TCP flooding
	COMBO	Sends spam data and open connection of IP, port
Mirai	Scan	Automatic scanning for vulnerable devices
	Ack	ACK flooding
	Syn	SYN flooding
	UDP	UDP flooding
	Plain UDP	Less of an option of UDP flooding for higher packet per second

Hình 4.2.5. Các kiểu tấn công trong tập datasets N-BaIoT

[5] https://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_BaIoT

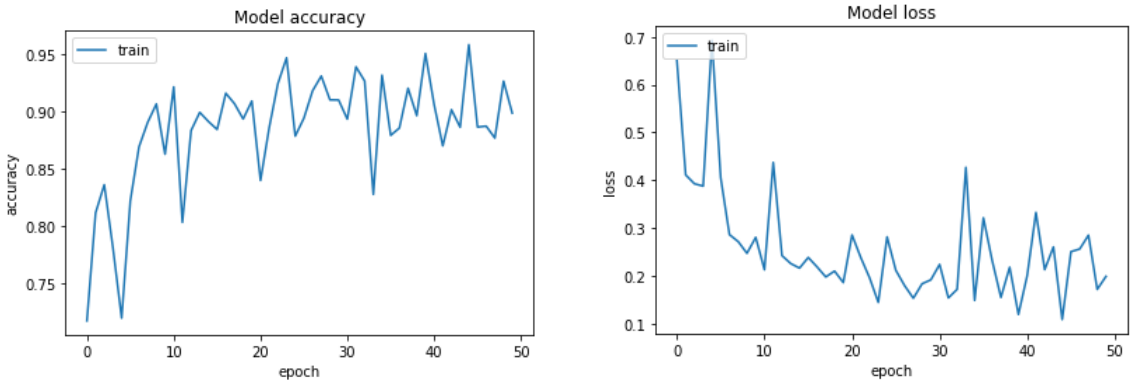
Tập dữ liệu này được chia làm 9 phần mỗi phần bao gồm các tập tin csv các loại tấn công. Trong thực nghiệm, nhóm tác giả chỉ sử dụng 1 phần để thực nghiệm bởi vì các phần còn lại đều tương tự như phần 1. Các thức chia tập dữ liệu cũng tương tự tập BoT-IoT.

4.3. Kết quả thực nghiệm và đánh giá

4.3.1. Kết quả trên mô hình phát hiện xâm nhập sử dụng phương pháp truyền thống

4.3.1.1. Tập dữ liệu BoT-IoT (UNSW-NB18)

- **Accuracy, Precison, Recall và F1-score:**



Hình 4.3.1 Biểu đồ loss và accuracy trên tập dữ liệu BoT-IoT trong ngữ cảnh truyền thống

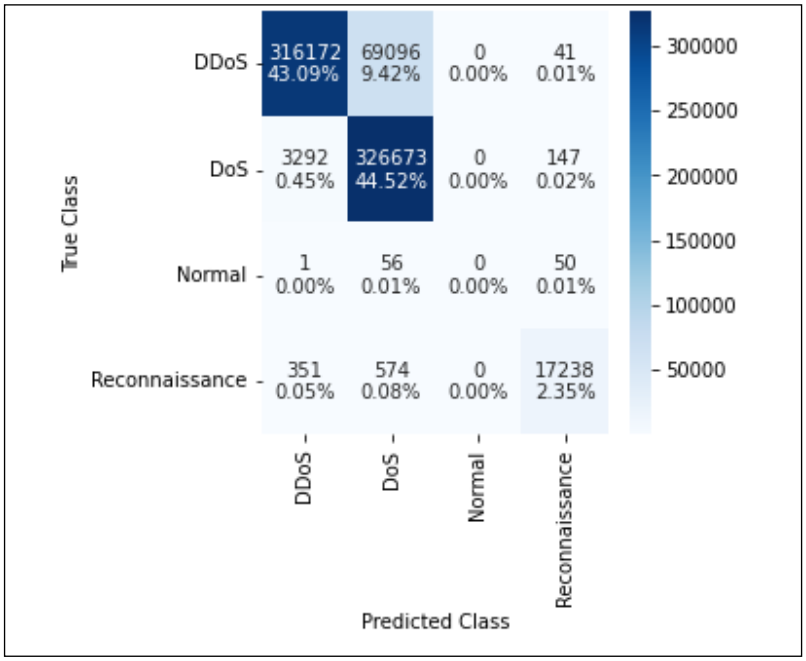
	precision	recall	f1-score	support
DDoS	0.99	0.82	0.90	385309
DoS	0.82	0.99	0.90	330112
Normal	0.00	0.00	0.00	107
Reconnaissance	0.99	0.95	0.97	18163
accuracy			0.90	733691
macro avg	0.70	0.69	0.69	733691
weighted avg	0.91	0.90	0.90	733691

Hình 4.3.2 Thống kê phân loại tập dữ liệu BoT-IoT trong ngữ cảnh truyền thống

Hình 4.3.1 cho thấy trong ngữ cảnh truyền thống, độ chính xác đạt được là 90-92%, loss cũng giảm dần qua các vòng.

Hình 4.3.2 cho thấy rằng Precision đạt được từ các cuộc tấn công lần lượt là 0.99, 0.82, 0.00, 0.99, điều này cho thấy tỷ lệ các mẫu dữ liệu tấn công được phân loại đúng là tấn công khá tốt, ít gặp trường hợp nguy hiểm là nhầm lẫn từ tấn công sang bình thường. Recall cũng đạt khoảng từ 0.82 – 0.95% . Tuy nhiên các mẫu dữ liệu nhãn 'normal' hầu như không phân loại được, nhóm tác giả cho rằng các mẫu dữ liệu nhãn 'normal' ít hơn nhiều so với các nhãn còn lại do đó các đặc điểm của những mẫu dữ liệu này bị nhầm lẫn sang các nhãn khác. Do Precision và Recall đạt kết quả ổn định ➔ F1-score có kết quả cao.

- **Confusion Matrix:**

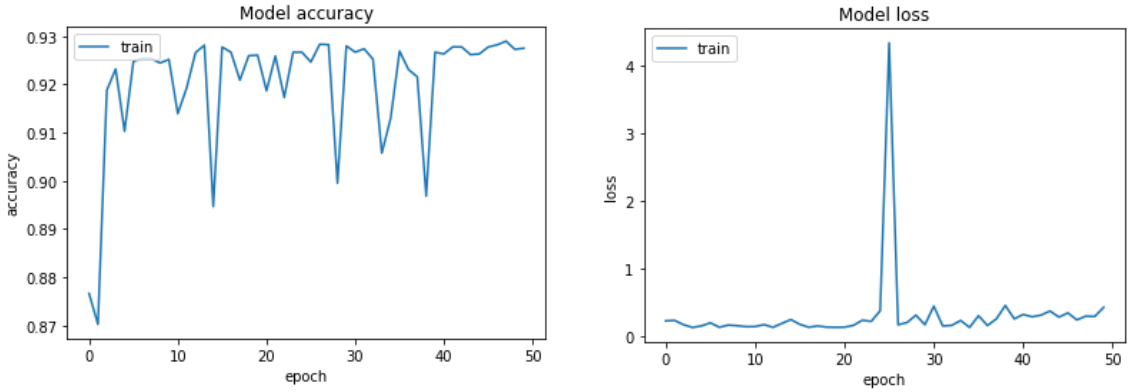


Hình 4.3.3 Confusion matrix trên tập dữ liệu BoT-IoT trong ngữ cảnh truyền thống

Hình 4.3.3 cho thấy tỷ lệ phân loại đúng đến các nhãn khá tốt, tỷ lệ phân loại sai các nhãn dao động từ 0 – 9.42%.

4.3.1.2. Tập dữ liệu NBaIoT

- **Accuracy, precision, recall và f1-score:**

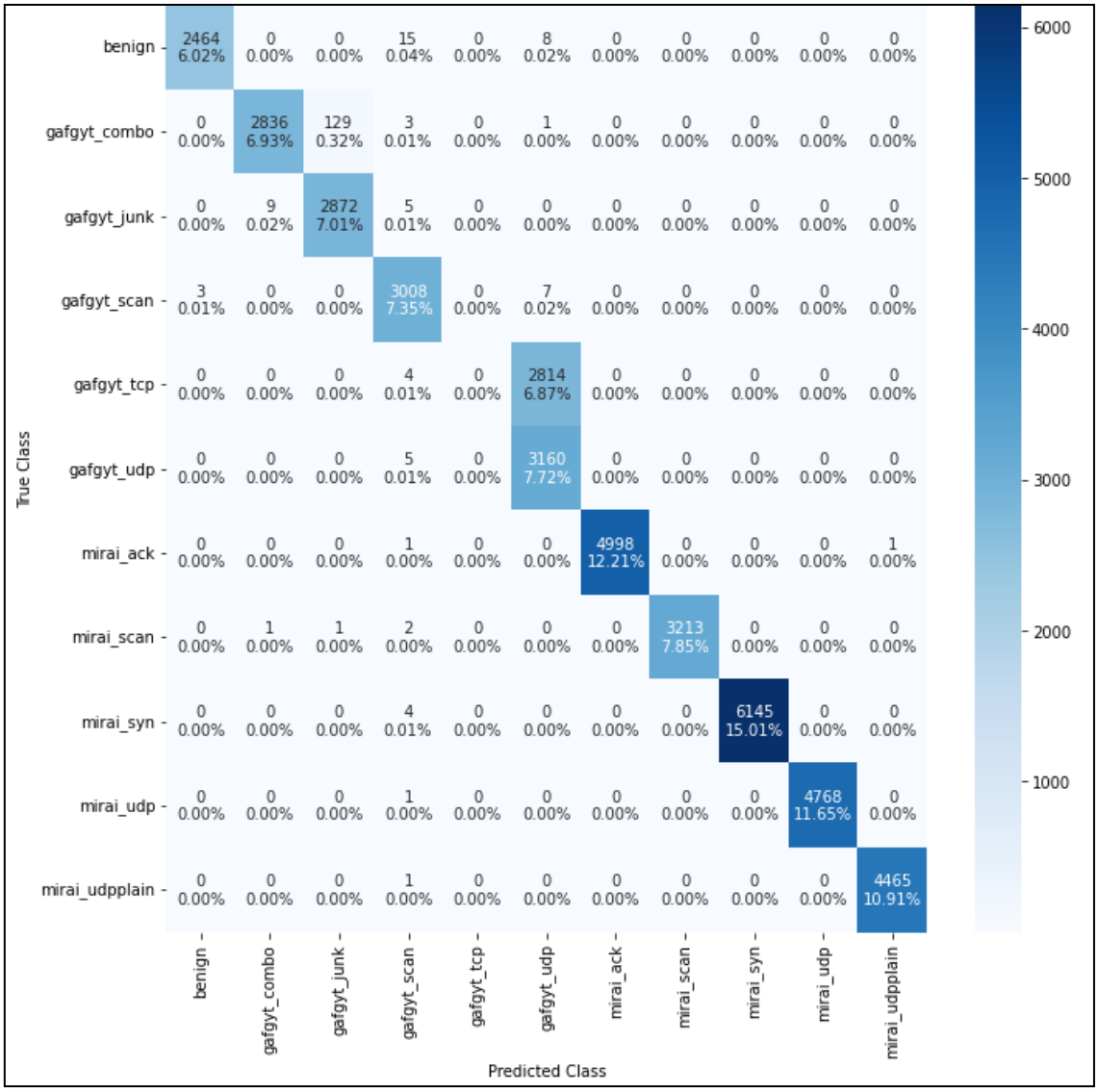


Hình 4.3.4. Biểu đồ loss và accuracy trên tập dữ liệu NBaIoT trong ngữ cảnh truyền thống

	precision	recall	f1-score	support
benign	1.00	0.99	0.99	2487
gafgyt_combo	1.00	0.96	0.98	2969
gafgyt_junk	0.96	1.00	0.98	2886
gafgyt_scan	0.99	1.00	0.99	3018
gafgyt_tcp	0.00	0.00	0.00	2818
gafgyt_udp	0.53	1.00	0.69	3165
mirai_ack	1.00	1.00	1.00	5000
mirai_scan	1.00	1.00	1.00	3217
mirai_syn	1.00	1.00	1.00	6149
mirai_udp	1.00	1.00	1.00	4769
mirai_udpplain	1.00	1.00	1.00	4466
accuracy			0.93	40944
macro avg	0.86	0.90	0.88	40944
weighted avg	0.89	0.93	0.90	40944

Hình 4.3.5. Thống kê phân loại tập dữ liệu NBaIoT trong ngữ cảnh truyền thống

Hình 4.3.4 và 4.3.5 cho thấy, trong ngữ cảnh truyền thống kết quả đạt được như sau: tỷ lệ chính xác (accuracy) đạt 0.93%. Precision trung bình đạt 0.86 , Recall trung bình đạt 0.9 ➔ f1-score trung bình đạt 0.88. Nhóm tác giả nhận xét với mô hình ANN với số lượng nơ-ron như thiết kế đạt kết quả khá chính xác, tuy nhiên nhãn ‘gafgyp_tcp’ và ‘gafgyp_udp’ vẫn còn nhầm lẫn với nhau dẫn đến phân loại trên 2 nhãn này không tốt.



- **Confusion matrix:**

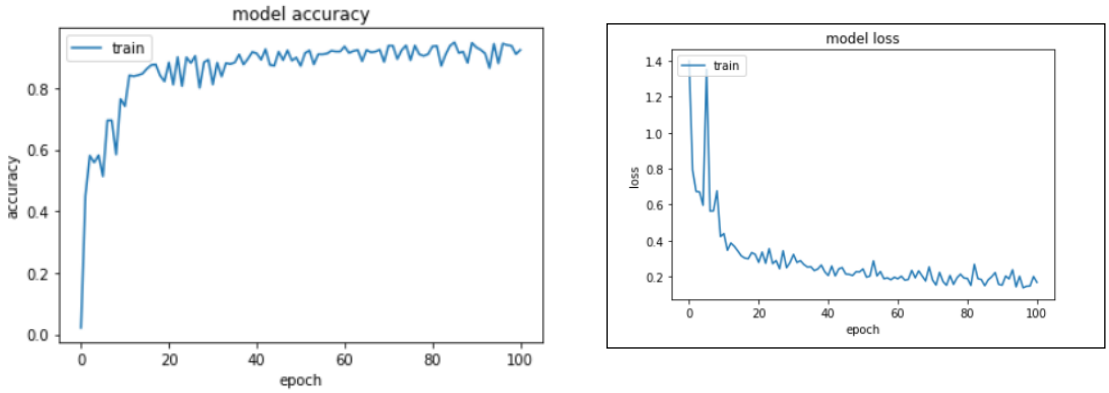
Hình 4.3.6. Confusion matrix trên tập dữ liệu NBaIoT trong ngữ cảnh truyền thống

Hình 4.3.6 cho thấy tỷ lệ phân loại đúng đến các nhãn khá tốt, tuy nhiên nhãn ‘gafgyp_tcp’ gần như bị nhầm lẫn hoàn toàn sang ‘gafgyp_udp’, còn lại các nhãn khác phân loại đúng khá cao, tỷ lệ nhầm lần giữa các nhãn chỉ dao động 0.00 – 0.01% ngoại trừ nhãn ‘gafgyp_tcp’ là 6.87%.

4.3.2. Kết quả trên mô hình phát hiện xâm nhập sử dụng phương pháp học hợp tác

4.3.2.1. Tập dữ liệu BoT-IoT (UNSW-NB18)

- **Accuracy, precision, recall và f1-score:**



Hình 4.3.7. Biểu đồ loss và accuracy trên tập dữ liệu BoT-IoT trong ngữ cảnh Federated learning

	precision	recall	f1-score	support
DDoS	0.91	0.94	0.93	385309
DoS	0.93	0.90	0.91	330112
Normal	1.00	0.11	0.20	107
Reconnaissance	1.00	0.97	0.98	18163
accuracy			0.92	733691
macro avg	0.96	0.73	0.76	733691
weighted avg	0.92	0.92	0.92	733691

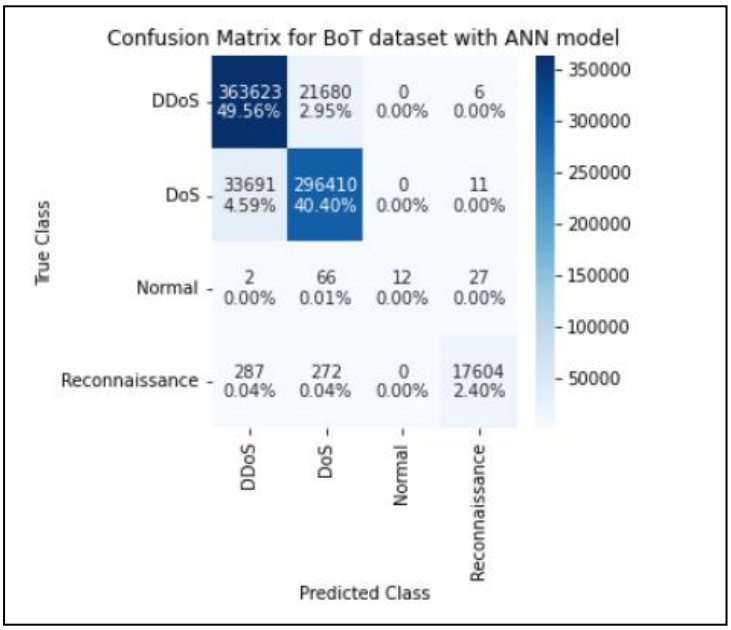
Hình 4.3.8. Thống kê phân loại tập dữ liệu BoT-IoT trong federated learning

Hình 4.3.7 cho thấy trong ngữ cảnh federated learning, độ chính xác đạt được là 92-93%, loss cũng giảm dần qua các vòng.

Hình 4.3.8 cho thấy rằng Precision đạt được từ các cuộc tấn công lần lượt là 0.91, 0.93, 1.00, 1.00, điều này cho thấy tỷ lệ các mẫu dữ liệu tấn công được phân loại đúng là tấn công khá tốt, ít gặp trường hợp nguy hiểm là nhầm lẫn từ tấn công sang bình thường. Recall cũng đạt khoảng từ 0.9 – 0.97% tuy nhiên trường hợp normal chỉ có 0.11. Nhóm tác giả cho rằng các mẫu dữ liệu nhãn ‘normal’ ít hơn nhiều so với các

nhãn còn lại do đó tỷ lệ phân loại đúng nhãn này không cao. Do Precision và Recall đạt kết quả ổn định ➔ F1-score có kết quả cao.

- **Confusion Matrix:**

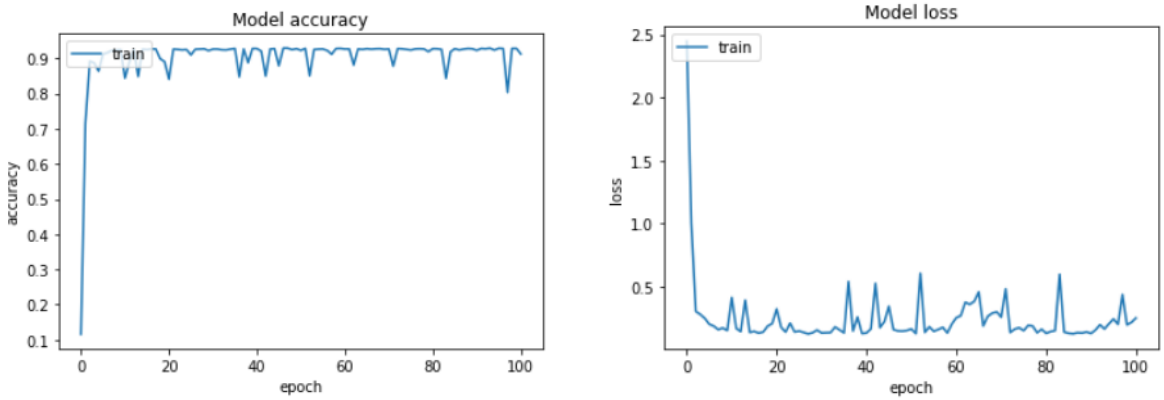


Hình 4.3.9 Confusion matrix trong ngữ cảnh federated learning

Hình 4.3.9 cho thấy tỷ lệ phân loại đúng đến các nhãn khá tốt, tỷ lệ phân loại sai các nhãn dao động từ 0.1 - 4.59%.

4.3.2.2. Tập dữ liệu NBaIoT

- **Accuracy:**



Hình 4.3.10 Biểu đồ loss và accuracy trên tập dữ liệu NBaIoT trong ngữ cảnh federated learning

	precision	recall	f1-score	support
benign	1.00	0.99	1.00	2487
gafgyt_combo	1.00	0.94	0.97	2969
gafgyt_junk	0.94	1.00	0.97	2886
gafgyt_scan	0.99	1.00	0.99	3018
gafgyt_tcp	0.00	0.00	0.00	2818
gafgyt_udp	0.53	1.00	0.69	3165
mirai_ack	1.00	0.89	0.94	5000
mirai_scan	1.00	1.00	1.00	3217
mirai_syn	1.00	1.00	1.00	6149
mirai_udp	0.90	1.00	0.95	4769
mirai_udpplain	1.00	1.00	1.00	4466
accuracy			0.91	40944
macro avg	0.85	0.89	0.86	40944
weighted avg	0.88	0.91	0.89	40944

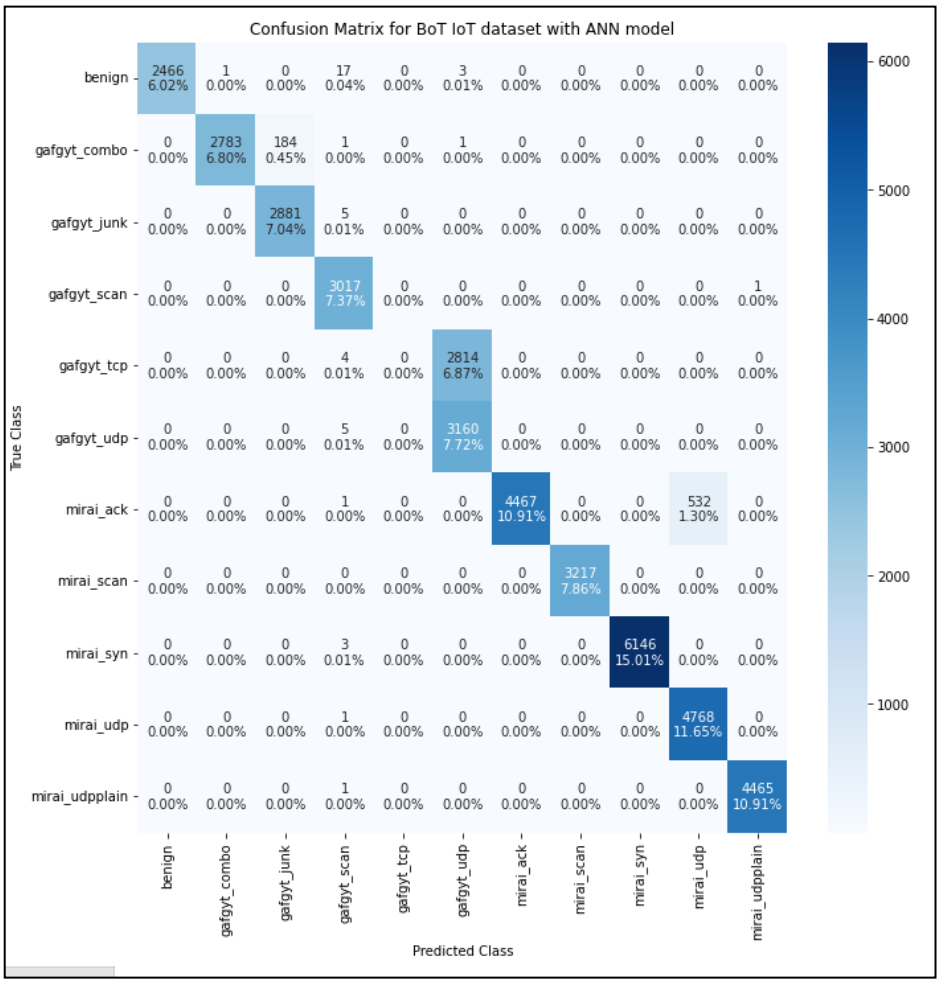
Hình 4.3.11. Thống kê phân loại tập dữ liệu NBaIoT trong ngữ cảnh federated learning

Hình 4.3.10 cho thấy trong ngữ cảnh federated learning, độ chính xác đạt được là 90 - 91%, loss cũng giảm dần qua các vòng.

Hình 4.3.11 cho thấy rằng Precision đạt được từ các cuộc tấn công dao động từ 0.9 – 1.00 ngoại trừ nhãn ‘gafgyp_tcp’ và ‘gafgyp_udp’ có nhẫm lần khá nhiều, nhóm tác giả cho rằng lí do có thể các mẫu dữ liệu của hai nhãn này khá giống nhau. Tỷ lệ các mẫu dữ liệu tấn công được phân loại đúng là tấn công khá tốt, ít gặp trường hợp nguy hiểm là nhầm lẫn từ tấn công sang bình thường. Recall cũng đạt khoảng từ 0.89 – 1.00. Do Precision và Recall đạt kết quả ổn định ➔ F1-score có kết quả ổn định trung bình là 0.89.

- **Confusion matrix:**

Hình 4.3.12 cho thấy tỷ lệ phân loại đúng đến các nhãn khá tốt, tuy nhiên nhãn ‘gafgyp_tcp’ gần như bị nhầm lẫn hoàn toàn sang ‘gafgyp_udp’, còn lại các nhãn khác phân loại đúng khá cao, tỷ lệ nhẫm lần giữa các nhãn chỉ dao động 0.00 – 0.01% ngoại trừ nhãn ‘gafgyp_tcp’ là 6.87%.



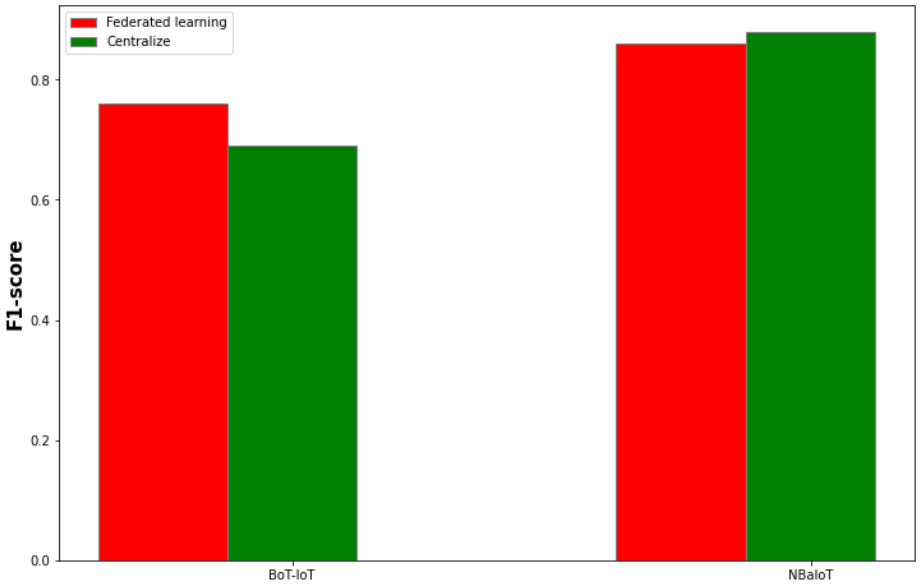
Hình 4.3.12 Confusion matrix tập dữ liệu NBaIoT trong ngữ cảnh federated learning

5. Tên sản phẩm: FedDIOT – Deep learning-based IDS áp dụng federated learning

6. Hiệu quả, phương thức chuyển giao kết quả nghiên cứu và khả năng áp dụng:

6.1. Đánh giá hiệu suất hệ thống

Nhóm tác giả dựa vào kết quả đạt được từ các thực nghiệm và đưa ra một số giả định so sánh IDS sử dụng deep learning trong ngữ cảnh truyền thống và học hợp tác như sau:



Hình 6.1.1. Biểu đồ so sánh kết quả thực nghiệm f1-score trên tập BoT-IoT, NBaIoT trong ngữ cảnh truyền thống và federated learning

Hình 6.1.1 nhóm tác giả sử dụng thông số f1-score để thực hiện so sánh giữa federated learning và truyền thống (centralize) bởi vì f1-score là trung bình giữa Precision và Recall, mặt khác f1-score đánh giá tốt hơn về phân loại đa nhãn so với accuracy. Kết quả biểu đồ cho thấy, giữa 2 ngữ cảnh đều đạt f1-score tương đồng với nhau, trong BoT-IoT federated learning còn hiệu quả hơn so với truyền thống.

Nhóm tác giả kết luận rằng, việc áp dụng mô hình học hợp tác (federated learning) để xây dựng deep learning-based IDS là có khả thi và độ hiệu quả đạt được tương tự như học truyền thống.

6.2. Phương thức chuyển giao kết quả nghiên cứu

Quá trình chuyển giao kết quả nghiên cứu như sau:

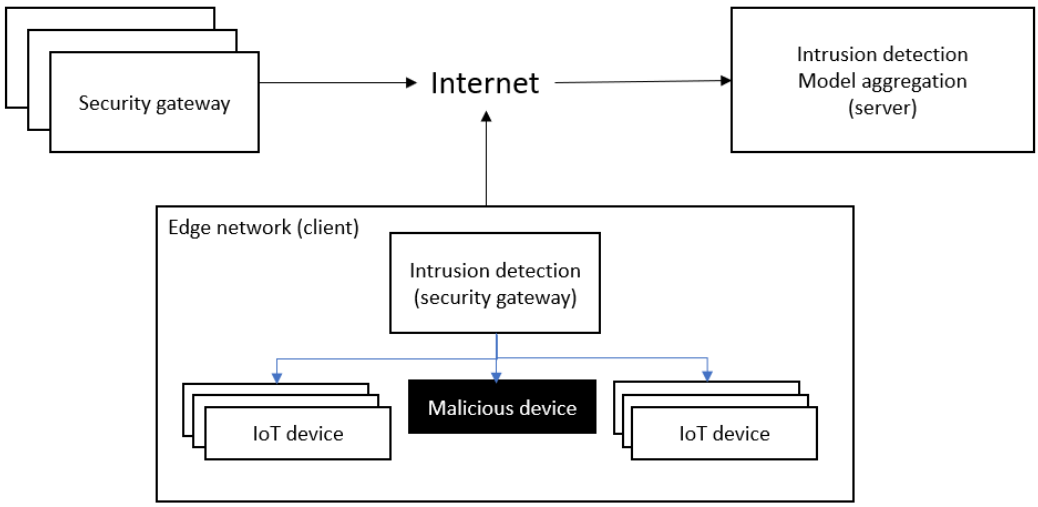
- Tìm hiểu và nghiên cứu các bài báo khoa học liên quan để nắm rõ các khái niệm.

- Xây dựng mô hình học hợp tác đơn giản bằng framework Flower để xem xét tính ổn định và hiệu quả.
- Tìm kiếm các tập dữ liệu trong ngữ cảnh IoT (N-BaIoT, UNSW-NB15, BoT-IoT)
- Thực nghiệm trên mô hình học truyền thống (centralize) trên Colab sử dụng mô hình ANN trên các tập dữ liệu đã chọn.
- Đưa các phần mã nguồn trên centralize vào mô hình federated learning đã xây dựng.
- Thực nghiệm tương tự trên các tập dữ liệu và so sánh, đánh giá kết quả giữa centralize và federated learning.

6.3. Khả năng áp dụng

Federated learning là một mô hình học máy mới mang lại nhiều lợi ích và khắc phục một số thách thức mà các mô hình truyền thống đang có, đặc biệt là đảm bảo được tính riêng tư của dữ liệu – một thách thức rất lớn trong ngành máy học. Trong đề tài nghiên cứu khoa học sinh viên này, nhóm tác giả sử dụng federated learning để thực hiện mô hình IDS dựa trên deep learning và từ thực nghiệm nhóm tác giả kết luận rằng áp dụng federated learning thay thế truyền thống là có khả thi. Vì vậy, federated learning cũng có thể áp dụng được cho nhiều bài toán khác phân loại khác như: hình ảnh, mã độc, spam,… Trong tương lai, Federated learning sẽ được áp dụng rộng rãi cho nhiều bài toán cụ thể và là bước đệm để phát triển lĩnh vực trí tuệ nhân tạo.

7. Hình ảnh, sơ đồ minh họa chính



8. Tài liệu tham khảo

[1] P. Sun *et al.*, "DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system," *Secur. Commun. Networks*, vol. 2020, 2020, doi: 10.1155/2020/8890306.

[2] T. V. Khoa *et al.*, "Collaborative Learning Model for Cyberattack Detection Systems in IoT Industry 4.0," *IEEE Wirel. Commun. Netw. Conf. WCNC*, vol. 2020-May, 2020, doi: 10.1109/WCNC45663.2020.9120761.

[3] and M. M. A. Noor Ali Al-Athba Al-Marri, Bekir S. Ciftler, "Federated Mimic Learning for Privacy Preserving Intrusion Detection Noor.pdf." Doha, Qatar, p. 6, 2020.

[4] 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.

[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," *Proc. - Int. Conf. Distrib. Comput. Syst.*, vol. 2019-July, no. Icdcs, pp. 756–767, 2019, doi: 10.1109/ICDCS.2019.00080.

[6] A. S. Almogren, "Intrusion detection in Edge-of-Things computing," *J.*

Parallel Distrib. Comput., vol. 137, pp. 259–265, 2020, doi: 10.1016/j.jpdc.2019.12.008.

[7] T. Duc Nguyen, P. Rieger, M. Miettinen, and A.-R. Sadeghi, "Poisoning Attacks on Federated Learning-based IoT Intrusion Detection System," *Work. Decentralized IoT Syst. Secur. 2020*, no. February, 2020, [Online]. Available: https://dx.doi.org/10.14722/diss.2020.23003.

[8] J. Kizza, F. Migga Kizza, J. Kizza, and F. Migga Kizza, "Intrusion Detection and Prevention Systems," *Securing the Information Infrastructure*. pp. 239–258, 2011, doi: 10.4018/978-1-59904-379-1.ch012.

[9] G. Fernandes, J. J. P. C. Rodrigues, L. F. Carvalho, J. F. Al-Muhtadi, and M. L. Proença, "A comprehensive survey on network anomaly detection," *Telecommun. Syst.*, vol. 70, no. 3, pp. 447–489, 2019, doi: 10.1007/s11235-018-0475-8.

[10] A. Galakatos, A. Crotty, and T. Kraska, "Distributed Machine Learning," in *Encyclopedia of Database Systems*, Springer New York, 2017, pp. 1–6.

[11] M. Aledhari, R. Razzak, R. M. Parizi, and F. Saeed, "Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Access*, vol. 8, pp. 140699–140725, 2020, doi: 10.1109/ACCESS.2020.3013541.

[12] M. S. R Heady, G Luger, A Maccabe, "The architecture of a network level intrusion detection system." 1990.

[13] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," *IEEE Access*, vol. 7. pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[14] H. Kukreja, "An introduction to artificial neural networks," *Hardw. Archit. Deep Learn.*, no. 5, pp. 3–26, 2020, doi: 10.1049/pbcs055e_ch1.

[15] D. J. Beutel *et al.*, "Flower: A Friendly Federated Learning Research Framework," 2020.

[16] M. Abadi *et al.*, "TensorFlow : A System for Large-Scale Machine Learning This paper is included in the Proceedings of the TensorFlow : A

system for large-scale machine learning," 2016.

[17] Z. Chen, N. Lv, P. Liu, Y. Fang, K. Chen, and W. Pan, "Intrusion Detection for Wireless Edge Networks Based on Federated Learning," *IEEE Access*, vol. 8, pp. 217463–217472, 2020, doi: 10.1109/ACCESS.2020.3041793.

[18] K. Li, H. Zhou, Z. Tu, W. Wang, and H. Zhang, "Distributed Network Intrusion Detection System in Satellite-Terrestrial Integrated Networks Using Federated Learning," *IEEE Access*, vol. 8, pp. 214852–214865, 2020, doi: 10.1109/ACCESS.2020.3041641.

[19] B. Li, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated Deep Learning for Intrusion Detection in Industrial Cyber-Physical Systems," *IEEE Trans. Ind. Informatics*, vol. 17, no. 8, pp. 5615–5624, Aug. 2021, doi: 10.1109/TII.2020.3023430.

[20] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, "Federated Learning-Based Cognitive Detection of Jamming Attack in Flying Ad-Hoc Network," *IEEE Access*, vol. 8, pp. 4338–4350, 2020, doi: 10.1109/ACCESS.2019.2962873.

Cơ quan Chủ trì	Chủ nhiệm đề tài
(ký, họ và tên, đóng dấu)	*(ký, họ và tên)*
	Phạm Ngọc Tâm