Bảo-đảm-ANTT - nhập môn mạng máy tính

nhập môn mạng máy tính (Trường Đại học Công nghệ thông tin, Đại học Quốc gia Thành phố Hồ Chí Minh)

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC VÀ KĨ THUẬT THÔNG TIN

BÁO CÁO ĐỒ ÁN

CHỦ ĐỀ: Tìm hiểu học cộng tác trong phân tích mã độc Windows

MÔN: NHẬP MÔN BẢO ĐẢM VÀ AN NINH THÔNG TIN

LỚP: IE105.N11

Giảng viên: Nguyễn Tấn Cầm

Nhóm sinh viên:

20521571 – Nguyễn Thành Long

20521439 - Lưu Thượng Vỹ

20520586 - Nguyễn Đình Khoa

Tp. Hồ Chí Minh, 11/2022

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

……, ngày…….....tháng……năm 20…

Người nhận xét

(Ký tên và ghi rõ họ tên)

BẢNG PHÂN CÔNG, ĐÁNH GIÁ THÀNH VIÊN:

MSSV	Họ và tên	Phân công	Đánh giá
20521571	Nguyễn Thành Long (Nhóm trưởng)	Làm file word	Hoàn thành tốt, đóng góp 100%
20522179	Lưu Thượng Vỹ	Làm file word	Hoàn thành tốt, đóng góp 100%
20520586	Nguyễn Đình Khoa	Làm Powerpoint Thuyết trình	Hoàn thành tốt, đóng góp 100%

Bảng phân công, đánh giá thành viên

MỤC LỤC

MỤC LỤC HÌNH ẢNH

TÌM HIỂU HỌC CỘNG TÁC TRONG PHÂN TÍCH MÃ ĐỘC WINDOWS

Valerian Rey[a] , Pedro Miguel Sánchez Sánchez[b,] , Alberto Huertas Celdrán[c] , Gérôme Bovet[d] and Martin Jaggi[a]*

[a]École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland
[b]Department of Information and Communications Engineering, University of Murcia, Murcia 30100, Spain

[c]Communication Systems Group (CSG), Department of Informatics (IfI), University of Zurich UZH, 8050 Zürich, Switzerland

[d]Cyber-Defence Campus, armasuisse Science & Technology, 3602 Thun, Switzerland

Article info.

Keywords

IoT Security

Federated Learning

IoT Device

Botnet Detection

Adversarial Attack

ABSTRACT

Hàng tỷ thiết bị IoT thiếu cơ chế bảo mật phù hợp đã được sản xuất và triển khai trong những năm qua, và nhiều hơn nữa sẽ đến với sự phát triển của công nghệ Beyond 5G. Các lỗ hổng của chúng đối với phần mềm độc hại đã thúc đẩy nhu cầu về các kỹ thuật hiệu quả để phát hiện các thiết bị IoT bị nhiễm bên trong mạng. Với sự riêng tư và tính toàn vẹn của dữ liệu đang trở thành mối quan tâm lớn trong những năm gần đây, ngày càng gia tăng khi có sự xuất hiện của mạng 5G và mạng Beyond, các công nghệ mới như học tập liên kết và blockchain đã xuất hiện. Chúng cho phép đào

tạo các mô hình machine learning với dữ liệu phi tập trung trong khi vẫn bảo toàn tính riêng tư theo thiết kế. Công việc này nghiên cứu các khả năng được kích hoạt bởi học tập liên kết liên quan đến phát hiện phần mềm độc hại IoT và nghiên cứu các vấn đề bảo mật vốn có trong mô hình học tập mới này. Trong bối cảnh này, một khuôn khổ sử dụng tính năng học tập liên kết để phát hiện phần mềm độc hại ảnh hưởng đến các thiết bị IoT được giới thiệu.

1. Introduction

Đến năm 2025, các dự báo ước tính rằng sẽ có khoảng 64 tỷ thiết bị IoT online[1]. Việc triển khai ồ ạt các thiết bị này chắc chắn đang biến thế giới thành một môi trường siêu kết nối. Mô hình IoT, cùng với các công nghệ mạng 5G và Beyond 5G (B5G) mới, đang cho phép các ứng dụng và doanh nghiệp mới chưa từng thấy trước đây, chẳng hạn như Công nghiệp 4.0 và Smart Cities, trong số nhiều công nghệ khác [2]. Tuy nhiên, đồng thời với những tiến bộ của công nghệ mới, số lượng và sự đa dạng của các cuộc tấn công mạng đã tăng lên trong những năm gần đây, khiến các phương pháp bảo mật hiện tại trở nên lạc hậu trong một thời gian ngắn [3]. Vì lý do này, việc kiểm soát an ninh của các môi trường mạng trong tương lai được kích hoạt bởi các công nghệ B5G đưa ra những thách thức lớn phải được giải quyết bằng các kỹ thuật hiện đại.

Một chiến lược đã trở nên phù hợp khi phát hiện các thiết bị đã bị phần mềm độc hại làm hỏng là giám sát các hoạt động của thiết bị để tạo ra các dấu vân tay hoặc hồ sơ.

Dấu vân tay có thể được sử dụng để phát hiện các sai lệch gây ra bởi các cuộc tấn công mạng hoặc các sửa đổi phần mềm độc hại [4]. Trong các thiết bị IoT , các nguồn hành vi không đồng nhất có thể được giám sát, chẳng hạn như truyền thông mạng, tiêu thụ tài nguyên, các hành động và sự kiện phần mềm hoặc tương tác của người dùng. Do đó, tùy

thuộc vào mục tiêu cần đạt được mà có thể sử dụng cái này hay cái khác. Cụ thể, khi nói đến việc phát hiện các cuộc tấn công mạng, khía cạnh được sử dụng rộng rãi nhất trong tài liệu là truyền thông mạng [5].

Sau khi các nguồn hành vi được chọn và giám sát, bước tiếp theo để phát hiện phần mềm độc hại thành công là xử lý dữ liệu và tạo dấu vân tay hành vi của thiết bị. Trong bối cảnh 5G và B5G, công nghệ Trí tuệ nhân tạo (AI) , chủ yếu là Machine Learning (ML) và Deep Learning (DL), đã đạt được mức độ liên quan to lớn trong những năm gần đây [6].

Ngày nay, hầu hết các giải pháp hiện có sử dụng ML / DL để phát hiện phần mềm độc hại đều dựa vào một thực thể trung tâm chịu trách nhiệm thu thập dữ liệu từ các thiết bị khác nhau và đào tạo các mô hình toàn cầu. Sau đó, các mô hình này được phân phối giữa các máy khách riêng lẻ hoặc các máy khách này gửi dữ liệu thử nghiệm trực tiếp của họ đến máy chủ để đánh giá hành vi và phát hiện phần mềm độc hại. Tuy nhiên, cách tiếp cận này không phù hợp với các tình huống mà hành vi của thiết bị chứa dữ liệu nhạy cảm hoặc bí mật sẽ ảnh hưởng đáng kể đến an ninh môi trường và sự riêng tư trong trường hợp rơi vào ma-licious. Tình huống tương tự cũng xảy ra trong các tình huống mà các nguồn dữ liệu được giám sát có liên quan đến con người và các hành động riêng tư có liên quan.

Trong bối cảnh mà quyền riêng tư và tính toàn vẹn của dữ liệu là rất quan trọng, Federated Learning (FL) [7] và Blockchain đang đạt được mức độ liên quan rất lớn trong những năm qua như một mô hình ML hợp tác. Trong FL, việc đào tạo thuật toán được thực hiện theo cách phi tập trung bởi các node hoặc client khác nhau, sử dụng dữ liệu cục bộ. Trong trường hợp này, mỗi nút phi tập trung đào tạo một mô hình riêng lẻ bằng cách sử dụng dữ liệu của chính nó và chia sẻ các tham số của mô hình (thay vì dữ liệu) với phần còn lại. Việc trao đổi các tham số mô hình và tổng hợp của chúng để tạo ra một mô hình toàn cầu và duy nhất có thể được thực hiện thông qua một thực thể trung tâm, được gọi là máy chủ, hoặc theo cách tiếp cận ngang hàng [8]. Sau nhiều lần lặp lại, mỗi máy khách có một mô hình toàn cục thu được dưới dạng tổng hợp của mô hình riêng lẻ của

mỗi máy khách. Cách tiếp cận này cho phép bảo mật dữ liệu theo thiết kế, vì dữ liệu không được chia sẻ với bất kỳ danh tính bên ngoài nào.

Bất chấp tính mới mẻ và lợi ích của các phương pháp FL, ứng dụng của nó trong các kịch bản thế giới thực vẫn đưa ra một số câu hỏi mở cần được phân tích và giải quyết (hoặc ít nhất là phải được chứng minh) [9]. Các công trình trước đây xử lý FL để phát hiện xâm nhập [10, 11, 12] thiếu việc sử dụng các bộ dữ liệu thực tế trong bối cảnh FL, phân tích về tác động bất lợi, hoặc phân tích việc triển khai của chúng trong các kịch bản B5G. Theo nghĩa này, một số thách thức mở có liên quan nhất có thể được tóm tắt như: 1) FL có thể được sử dụng trong bối cảnh IoT như thế nào để xây dựng các mô hình chung mà không cần chia sẻ dữ liệu nhạy cảm ?; 2) Các phương pháp tiếp cận FL ảnh hưởng như thế nào đến hiệu suất của máy dò an-omaly và phân loại đa phương truyền thống trong các tình huống IoT ?; 3) Tác động của các cuộc tấn công adversarial khác nhau ảnh hưởng đến các mô hình liên hợp được thiết kế để phát hiện các cuộc tấn công mạng trên các kịch bản IoT là gì ?; và 4) Các cơ chế đối phó hiện tại có thể giảm thiểu tác động của các cuộc tấn công adversarial không ?; và nếu có, 5) Các biện pháp đối phó phù hợp nhất cho các tình huống IoT là gì ?; 6) Làm thế nào những giải pháp này có thể được kết hợp trong các mạng tương lai như B5G?

- Với mục tiêu vượt qua các đòn bẩy mở trước đây , bài báo này trình bày những đóng góp chính sau:

  • Một ca sử dụng trình bày kịch bản B5G trong đó cần phát hiện các cuộc tấn công mạng ảnh hưởng đến các thiết bị IoT, quản lý dữ liệu nhạy cảm, có dữ liệu Non-IID (Phụ thuộc và được phân phối giống nhau) và với các bên liên quan hoặc khách hàng không đáng tin cậy.

  • Một khuôn khổ bảo mật sử dụng FL để phát hiện các cuộc tấn công mạng ảnh hưởng đến các thiết bị IoT, trong một thời điểm sơ khai, các cuộc tấn công mạng ảnh hưởng đến các thiết bị IoT. Khung được đề xuất bao gồm cả các phương pháp tiếp cận phát hiện và phân loại anom anom bằng cách sử dụng các chi bảo vệ ar mạng thần kinh nhiều lớp perceptron và autoencoder .

• Một nhóm các thử nghiệm đo lường hiệu suất của khuôn khổ được đề xuất khi phát hiện phần mềm độc hại trong các thiết bị IoT. Để đạt được điều đó, các nội dung tiếp theo đã được so sánh: i) Một cách tiếp cận tập trung, nơi tất cả dữ liệu được chia sẻ, ii) Phương pháp tiếp cận phân tán trong đó mỗi phương thức đào tạo một mô hình độc lập với dữ liệu cục bộ của nó, và iii) Cách tiếp cận liên hợp trong đó một mô hình chung được tạo ra chia sẻ các bản cập nhật của mô hình cục bộ. Hai thuật toán học tập liên kết khác nhau khác nhau về số lượng liên lạc (cập nhật chia sẻ mô hình) với sever đã được xem xét trong so sánh trước đó.
• Đánh giá tác động của một số cuộc tấn công đối nghịch ảnh hưởng đến giải pháp FL của chúng tôi. Mục tiêu là để đo lường hiệu suất của các mô hình liên kết suy giảm như thế nào khi một số ứng dụng client độc hại và gửi các bản cập nhật mô hình bị giả mạo. Bên cạnh đó, người ta đã đánh giá xem các chức năng tổng hợp khác nhau hoạt động như các cơ chế đối phó cải thiện khả năng phục hồi của mô hình chống lại các cuộc tấn công của đối thủ như thế nào.
• Thảo luận về kết quả đối đầu, chi phí tính toán và giao tiếp cũng như thiết kế khuôn khổ, mô tả các vấn đề có thể xảy ra và hạn chế trong các tình huống B5G, cùng với giải pháp khả thi của chúng.

Phần còn lại của bài viết này có nội dung như sau. Phần 2 mô tả công việc liên quan đến AI cho an ninh mạng IoT, thuật toán FL, lỗ hổng bảo mật và các biện pháp đối phó cũng như bộ dữ liệu chứa dữ liệu tấn công mạng IoT. Phần 3 mô tả một kịch bản IoT với các yêu cầu về quyền riêng tư được thực hiện bởi bộ dữ liệu N-BaIoT, được dùng làm trường hợp sử dụng cho công việc này. Phần 4 trình bày chi tiết về thiết kế và triển khai khuôn khổ được đề xuất, sử dụng FL để phát hiện phần mềm độc hại sau các thiết bị IoT. Phần 5 xác định các cuộc tấn công đối thủ và các biện pháp đối phó được thử nghiệm dựa trên khuôn khổ đề xuất. Phần 6 cho thấy kết quả của các thử nghiệm được thực hiện trong công việc này, so sánh các phương pháp tiếp cận liên hợp với các phương pháp truyền thống và nêu chi tiết kết quả của các thiết lập đối thủ. Phần 7 phân tích những bài học

kinh nghiệm cũng như những mặt hạn chế có thể có của kiến trúc. Cuối cùng, phần 8 cho thấy các kết luận của nghiên cứu này và các hướng đi trong tương lai.

2. Công việc liên quan:

Phần này trình bày chi tiết về tình trạng hiện đại trong các chủ đề khác nhau được đề cập trong tác phẩm hiện tại. Đầu tiên, nó đánh giá việc sử dụng AI cho an ninh mạng IoT, đặc biệt xem xét FL.

Sau đó, nó mô tả tài liệu chính về các cuộc tấn công của đối thủ chống lại quá trình FL và các biện pháp giảm thiểu có thể có của chúng. Cuối cùng, nó xem xét các bộ dữ liệu có sẵn mô hình hóa các cuộc tấn công mạng trên các thiết bị IoT.

2.1. Trí tuệ nhân tạo cho bảo mật mạng IoT

Các kỹ thuật AI truyền thống đã được áp dụng rộng rãi trong tài liệu để phát hiện các vấn đề an ninh mạng trong các tình huống IoT. Trong [4], các công trình hiện có về dấu vân tay hành vi của thiết bị đã được khảo sát, bao gồm cả những công trình nhắm mục tiêu đến bảo mật IoT. Công việc này cho thấy các giải pháp bảo mật IoT ngày nay đang chuyển sang áp dụng các kỹ thuật ML và DL như thế nào. Theo hướng này, các tác giả của đã sử dụng kỹ thuật ML để phát hiện sớm phần mềm độc hại không đồng nhất ảnh hưởng đến các thiết bị IoT.

Một công trình khác đã được trình bày trong [5], trong đó nhiều hệ thống phát hiện xâm nhập cho các thiết bị IoT đã được xem xét, đưa ra các khuyến nghị để thiết kế các giải pháp phát hiện intru sion mạnh mẽ và nhẹ cho IoT.

Trong những năm gần đây, FL đang trở nên quan trọng trong lĩnh vực an ninh mạng, với một số công trình đã sử dụng mô hình này cho bảo mật IoT. Trong bối cảnh này, nghiên cứu được đề xuất trong [14] đã nêu rõ vấn đề bảo mật dữ liệu của các giải pháp dựa trên AI truyền thống , nhưng việc đánh giá diễn ra trên một tập dữ liệu riêng tư.

Ngoài ra, dữ liệu được phân chia ngẫu nhiên giữa các khách hàng, điều này có thể không xảy ra trong các tình huống thực tế, như trường hợp được xem xét trong tác phẩm này, trong đó mỗi dữ liệu khách hàng đến từ một phân phối ent khác nhau nói chung. Các công trình được trình bày trong [10, 11] cũng có các mục tiêu rất giống nhau, nhưng các nghiên cứu này được thực hiện đặc biệt cho các thiết bị IoT công nghiệp và chúng phân tích các mẫu ứng dụng và cảm biến đọc dữ liệu tương ứng chứ không phải dữ liệu mạng, như chúng tôi thực hiện trong công trình này. Trong [12], FL đã được nghiên cứu thông qua trường hợp sử dụng của các hệ thống phát hiện xâm nhập. Công việc này cũng bao gồm công nghệ blockchain để giảm thiểu các vấn đề gặp phải trong FL đối nghịch. Tuy nhiên, nó tập trung vào các bước phát hiện xâm nhập ban đầu hơn là phát hiện phần mềm độc hại đã chạy và nó không tập trung đặc biệt vào các thiết bị IoT.

Tóm lại, phần này đã chỉ ra sự thiếu các giải pháp đối phó với các phương pháp tiếp cận FL xem xét dữ liệu được tạo bởi các nguồn phi tập trung để phát hiện phần mềm độc hại ảnh hưởng đến các thiết bị và kịch bản IoT .

2.2. Thuật toán học liên kết, các lỗ hổng và biện pháp đối phó

Tập trung vào các thuật toán FL và các đặc điểm riêng của chúng, công việc của [7] định nghĩa thuật ngữ học liên hợp bằng ký tự giải quyết vấn đề tối ưu hóa phi IID phi tập trung. Họ đề xuất thuật toán Trung bình Liên kết (FedAVG) hiện đóng vai trò như một cơ sở mạnh mẽ cho nhiều nghiên cứu của chúng tôi trong FL. Trong thuật toán này, một số client sử dụng bộ dữ liệu riêng lẻ của họ để cộng tác đào tạo một mô hình toàn cầu, nhờ vào sự điều phối được cung cấp bởi một máy chủ trung tâm. Vai trò của máy chủ là tính trung bình các tham số của các mô hình do máy khách gửi và trả về mô hình toàn cục. FL đã trưởng thành rất nhiều kể từ đó và một số cuộc khảo sát ([8, 15]) xem xét những tiến bộ mới nhất trong lĩnh vực đó.

Do tính chất phi tập trung của nó, FL chia sẻ mối đe dọa giữa nhiều thực thể, cụ thể là các máy khách và máy chủ. Công việc của [16] xem xét nhiều vấn đề có thể phát

sinh khi xem xét thiết lập đối thủ, cũng như hầu hết các biện pháp phòng thủ nổi tiếng để bảo vệ hệ thống chống lại điều đó. Trong [17], một số cuộc tấn công nhiễm độc dữ liệu chống lại Máy hỗ trợ Vec tor đã được xác định. Thử nghiệm cơ bản của họ sử dụng ý tưởng lật nhãn, trong đó nhãn nhị phân của một số điểm dữ liệu trong tập huấn luyện được đảo ngược để cản trở việc đào tạo mô hình. Trong [18], các tác giả đã nghiên cứu khả năng phục hồi của việc triển khai phân tán Stochastic Gradient Des cent chống lại các đối thủ có hành vi tùy tiện (Byzantine). Cuối cùng, một cuộc tấn công đầu độc mô hình từ quan điểm của một ứng dụng khách độc hại, có khả năng ước tính gradient, đã được thử nghiệm. Đầu tiên, nó chứng minh rằng mô hình thông thường tính trung bình bước được thực thi bởi máy chủ trong hầu hết các nhịp điệu FL không xử lý ngay cả một ứng dụng khách độc hại duy nhất trong liên kết. Nói một cách tổng thể hơn, họ đã chứng minh rằng không có hàm tổng hợp mô hình nào, tuyến tính trong các mô hình do khách hàng gửi, mạnh chống lại các đối thủ Byzantine. Trong [19], hai hàm tổng hợp mô hình mạnh mẽ theo từng bậc đã được đề xuất. Đặc biệt, chúng dựa trên đường trung bình theo tọa độ và giá trị trung bình được cắt theo tọa độ của các mô hình do khách hàng gửi đến máy chủ. Các tác giả của [20] đề xuất lấy lại mẫu để giảm sự không đồng nhất trong việc phân phối các mô hình do khách hàng gửi. Nó có nghĩa là được áp dụng trước khi sử dụng một hàm tổng hợp mạnh mẽ, và nó nhằm mục đích giảm các tác dụng phụ mà hàm như vậy có khi áp dụng cho các mô hình được đào tạo với bộ dữ liệu không phải IID.

Cuối cùng, trong tài liệu cũng có các phân quyền để đảm bảo các tác vụ tính toán phân tán sử dụng các phương pháp tiếp cận học tập củng cố trong các tình huống không có sự tin tưởng trực tiếp vào khách hàng nhưng vẫn mong muốn có được kết quả chính xác ngay cả khi có một số khách hàng có hành vi độc hại [21] .

2.3. Tập dữ liệu mô hình hóa các cuộc tấn công mạng ảnh hưởng đến các thiết bị IoT

Bộ dữ liệu là chìa khóa cho AI nói chung và FL nói riêng. Theo nghĩa này, một số bộ dữ liệu mạng công cộng về bảo mật IoT có thể được tìm thấy trong tài liệu. Bảng 1 đánh giá một số trong những bảng thú vị nhất. Tất cả các bộ dữ liệu đó đều được tạo ở một vị trí trung tâm, nhưng đối với một số chúng, một chiến lược phân tách thực tế có thể thực hiện được để sử dụng chúng trong các phương pháp tiếp cận FL. Trong bối cảnh này, cột Tách trình bày đề xuất của chúng tôi về các chiến lược khả thi để tách tập dữ liệu giữa các thực thể khác nhau. Trong chiến lược Tách thiết bị , tập dữ liệu đã có lưu lượng truy cập từ mỗi thiết bị được đặt vào một tệp khác nhau. Chiến lược IP sẽ bao gồm việc nhóm các mẫu tập dữ liệu theo địa chỉ IP để cách ly lưu lượng của từng thiết bị theo cách thủ công. Chiến lược phân tách kịch bản sẽ tận dụng lợi thế của thực tế là tập dữ liệu được tạo trong một số tình huống khác nhau và có thể coi mỗi kịch bản là đến từ một khách hàng khác nhau.

Trong [22], một tập dữ liệu có tên N-BaIoT được tạo ra bằng cách xử lý trước lưu lượng được tạo ra bởi 9 thiết bị IoT thương mại thuộc nhiều loại khác nhau, hoặc bị nhiễm bởi Mirai hoặc BASHLITE(hai cuộc tấn công bằng phần mềm độc hại botnet) hoặc không bị gián đoạn. Trong [23], một mạng quy mô trung bình gồm 83 thiết bị IoT thực hoặc giả lập được coi là để tạo ra tập dữ liệu MedBIoT. Nó sử dụng cùng một quy trình xử lý gói trước như trong N-BaIoT, nhưng ở đây các giai đoạn khác của lưu lượng phần mềm độc hại được xem xét (lây nhiễm, lan truyền và giao tiếp với máy chủ chỉ huy và điều khiển). Trong [24], tập dữ liệu đánh giá bao gồm một mạng gồm 8 camera an ninh bị một số cuộc tấn công. Ngoài ra, họ bao gồm một mạng khác bao gồm 9 thiết bị IoT thương mại, trong đó có một thiết bị đã bị nhiễm Mirai. [25] đề xuất một tập dữ liệu được gọi là Bot_IoT, chứa lưu lượng mạng IoT hợp pháp và mô phỏng, bao gồm các cuộc tấn công khác nhau. Tập dữ liệu TON_IoT [26] bao gồm dữ liệu không đồng nhất nguồn (dữ liệu mạng nhưng cũng có thể đọc cảm biến, nhật ký hệ điều hành và dữ liệu đo từ xa) về mạng chứa một số thiết bị IoT / IIoT. Trong [27], các tác giả đề xuất một tập dữ liệu thu thập các dấu vết lưu lượng tấn công lành tính và thể tích cho 27 thiết bị IoT. Mục đích chính của tập dữ liệu này là để

đánh giá các cuộc tấn công theo khối lượng được thực hiện chống lại một mạng chứa các thiết bị IoT thương mại thực. Tập dữ liệu được đề xuất trong [28] được tạo với lưu lượng của 2 thiết bị IoT gia đình trong nhiều tình huống tấn công. Nó cũng bao gồm lưu lượng Mirai mô phỏng dường như đến từ các thiết bị IoT. Cuối cùng, IoT 23 [29] là một tập dữ liệu bao gồm 20 bản ghi bao gồm hoạt động của phần mềm độc hại cũng như 3 bản ghi lưu lượng truy cập IoT lành tính. Một số tác phẩm sử dụng nhiều mẫu để phát hiện phần mềm độc hại đáng kể và những tác phẩm khác tập trung vào nhiệm vụ chi tiết hơn là phân loại đơn mẫu. Trong phương pháp luận của chúng tôi, cả hai Để kết luận, cần phải đề cập đến việc thiếu bộ dữ liệu phù hợp cho các phương pháp FL phát hiện phần mềm độc hại trong các thiết bị IoT. Các giải pháp dựa trên FL hiện tại phải xem xét các bộ dữ liệu tập trung được chia nhỏ để áp dụng các kỹ thuật liên kết.

-

Ref .	**Name**	**Year**	**Splitting**
[22]	N-BaIoT	2018	Device
[23]	MedBIoT	2020	IP
[24]	Kitsune	2019	Unrealistic
[25]	Bot_IoT	2018	IP, scenario
[26]	TON_IoT	2019	IP, scenario
[27]	IoT benign & attack traces	2019	IP, scenario
[28]	IoT network intrusion	2019	Unrealistic
[29]	IoT-23	2020	Unrealistic

Bảng 1. Bộ dữ liệu mạng IoT công cộng.

3. Trường hợp sử dụng: Tình huống IoT bị phần mềm độc hại ảnh hưởng

Phần này trình bày các đặc điểm của kịch bản được xác định trong công việc này và giải thích chi tiết về tập dữ liệu được sử dụng để đánh giá hiệu suất của khuôn khổ đề xuất.

Các thành phố của chúng ta có hàng triệu thiết bị IoT được kết nối với Internet và cảm nhận các phần dữ liệu không đồng nhất. Số lượng và tính không đồng nhất của các thiết bị sẽ tăng lên theo cấp số nhân với sự ra đời của mạng B5G, vì chúng cho phép các ngành dọc và kịch bản mới dựa trên hiệu suất mạng được nâng cao về độ trễ và thông lượng [30]. Một số ví dụ là Dịch vụ phương tiện bay không người lái, Dịch chuyển ba chiều hoặc Thực tế mở rộng. Trong bối cảnh như vậy, các vấn đề về quyền riêng tư thường xuyên xuất hiện khi các phần dữ liệu được cảm nhận thuộc về các khía cạnh nhạy cảm trong cuộc sống hàng ngày hoặc tổ chức của chúng ta [31]. Khi bị quỷ ám , các thiết bị IoT bị hạn chế về tài nguyên và không được thiết kế với tính bảo mật, khiến chúng dễ bị tấn công bởi nhiều loại phần mềm độc hại. Trong các tình huống này, các phương pháp tiếp cận phát hiện dựa trên AI truyền thống không phù hợp do không thể đào tạo các mô hình tập trung với dữ liệu nhạy cảm thuộc các tổ chức hoặc đối tượng khác nhau.

Do đó, FL đang phát triển như một cơ chế chính để phát hiện các hành vi bất thường và kích hoạt các cơ chế giảm thiểu trong các tình huống nhạy cảm về quyền riêng tư được kích hoạt bởi mạng 5G và B5G. Tuy nhiên, FL cũng gặp phải các vấn đề cố hữu là giao dịch với các bên không xác định và do đó, không đáng tin cậy. Các máy khách độc hại thực hiện các cuộc tấn công đầu độc đối với dữ liệu và mô hình là một trong những ví dụ tốt nhất theo hướng này. Tiếp theo các đặc điểm trước đó, công việc này xem xét các khía cạnh chính sau đây cho kịch bản đã xác định: i) dữ liệu được phân phối không đồng nhất trên các thiết bị IoT (thuộc sở hữu của khách hàng), ii) cần thiết để phát hiện các bất thường do không nhìn thấy hoặc không phần mềm độc hại ngày ảnh hưởng đến các thiết bị IoT, iii) cần phải phân loại phần mềm độc hại nổi tiếng ảnh hưởng đến các thiết bị IoT khác nhau, iv) đối thủ có thể xuất hiện giữa các máy khách liên kết, vì vậy một số biện pháp đối phó cần được áp dụng.

Một số bộ dữ liệu công khai phù hợp với các kịch bản ứng dụng B5G và phần mềm độc hại IoT tồn tại trong tài liệu. Trong số đó, N-BaIoT là phù hợp nhất để đánh giá đào tạo hợp tác bảo vệ quyền riêng tư . Cụ thể, tập dữ liệu này đã phân tách dữ liệu lưu lượng truy cập của các thiết bị IoT thành các tệp khác nhau, giúp dễ dàng chia nó thành nhiều

phần không được phân phối giống nhau cho một cài đặt liên kết thực tế. Vì lý do đó, chúng tôi đã chọn N-BaIoT để đánh giá cách tiếp cận của mình. Lưu ý rằng một hạn chế của tập dữ liệu này là nó chỉ chứa dữ liệu từ 9 thiết bị IoT, đây là một hạn chế đối với các thử nghiệm vì nó giới hạn số lượng khách hàng tối đa có thể được xem xét.

N-BaIoT chứa các gói được xử lý trước từ lưu lượng của 9 thiết bị IoT. Tất cả các thiết bị đã tạo ra một số lưu lượng truy cập trong khi không bị hỏng (mẫu lành tính) và trong khi được Mirai và BASHLITE kiểm tra thông tin. Tất cả các thiết bị, ngoại trừ chuông cửa Ennio và webcam, cũng đã tạo ra một số lưu lượng truy cập khi bị lây nhiễm bởi Mirai. Bảng 2 cho thấy số lượng các mẫu lành tính và tấn công cho mỗi thiết bị, cũng như tổng số.

Mỗi mẫu trong tập dữ liệu tương ứng với một gói mạng được Wireshark phát hiện. Đối với mỗi, 115 fea tures số đặc trưng cho ngữ cảnh của gói tin đã được trích xuất. Các tính năng có sẵn là thống kê về kích thước, số lượng và độ chập chờn của các gói mạng tổng hợp, trong 100 mili giây, 500 mili giây, 1,5 giây, 10 giây và 1 phút. Ví dụ, một tính năng là kích thước gói trung bình trong 10 giây qua trong lưu lượng giữa IP nguồn gói hiện tại và IP đích. Rõ ràng là các tính năng của các gói được bắt trong một khoảng thời gian rất ngắn có mối tương quan rất cao. Điều này có nghĩa là tập dữ liệu này cần phải được xử lý cẩn thận để giảm càng nhiều càng tốt việc rò rỉ dữ liệu giữa tàu và các tập kiểm tra khi tách một tệp nhất định thành hai phần đó. Để đạt được mục tiêu đó, chúng tôi luôn sử dụng phân tách theo thứ tự thời gian để tạo các phần đào tạo và kiểm tra , và chúng tôi để lại một tập hợp nhỏ mẫu không được sử dụng giữa phần đào tạo và phần kiểm tra cho mỗi tệp trong tập dữ liệu.

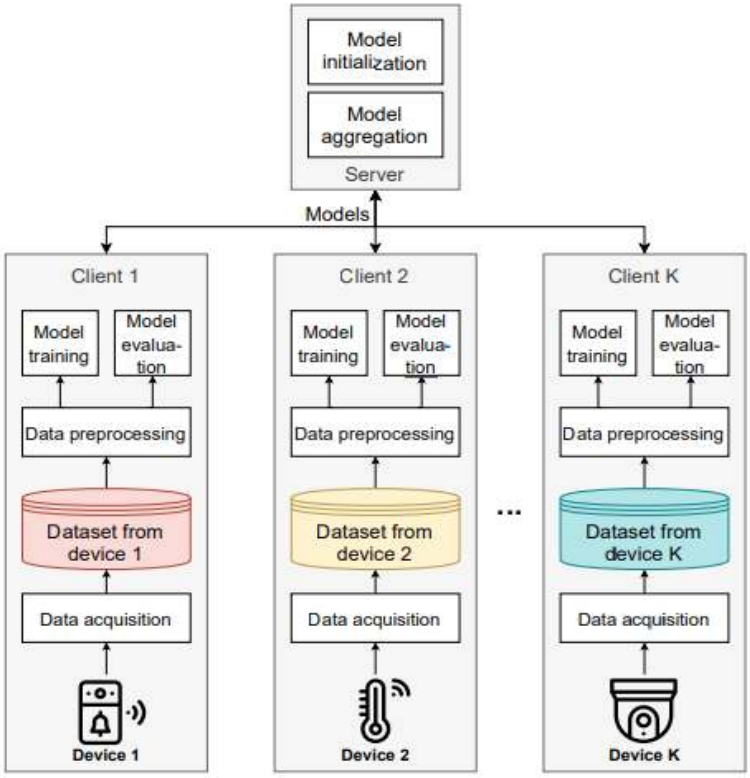
Sau khi phân tích các đặc điểm phù hợp nhất của tập dữ liệu, chúng tôi cũng đã xem xét một số công trình đáng chú ý nhất hiện có trên NBaIoT [22, 32, 33, 34, 35, 36]. Hầu hết những giải pháp đó tập trung vào các giải pháp phát hiện bất thường không được giám sát, chỉ sử dụng phần lành tính của tập dữ liệu để đào tạo. Tuy nhiên, trong [36], các siêu tham số cũng được điều chỉnh bằng cách sử dụng một số dữ liệu tấn công, và trong [34],

thay vào đó, một phân loại có giám sát được xem xét. Một số tác phẩm sử dụng nhiều mẫu để phát hiện phần mềm độc hại đáng kể và những tác phẩm khác tập trung vào nhiệm vụ chi tiết hơn là phân loại đơn mẫu. Trong phương pháp luận của chúng tôi, cả hai được giám sát và các tình huống không được giám sát được xem xét, và sự chú ý được đặt vào phân tích đơn mẫu. bên trong tình huống được giám sát, vì N-BaIoT chứa 10 cuộc tấn công khác nhau được thực hiện bằng Mirai và BASHLITE, chúng tôi sử dụng tất cả các cuộc tấn công có thể có sẵn được gắn nhãn bằng cách sử dụng cùng một lớp (tấn công) theo thứ tự để phát hiện càng nhiều cuộc tấn công càng tốt. Lưu ý rằng trong [35], a phương pháp học tập hợp tác được đề xuất. Tuy nhiên, kịch bản giả định và mục tiêu khác với của chúng tôi, vì chúng tập trung vào việc xây dựng một mô hình cho mỗi thiết bị với giả định rằng dữ liệu của một thiết bị đến từ 2 hoặc 3 thiết bị khác nhau các nguồn.

4. Khung dựa trên Học tập liên kết và triển khai

Phần này trình bày chi tiết về thiết kế kiến trúc của khung dựa trên FL được đặt ra chuyên nghiệp, mô tả các thành phần của nó và cách họ tương tác với nhau trong quá trình đào tạo người mẫu và các quy trình đánh giá. Bên cạnh đó, nó cũng mô tả cách khung được triển khai cho trường hợp sử dụng xác thực của chúng tôi, tận dụng tập dữ liệu N-BaIoT.

Kiến trúc khung, được mô tả trong Hình 1, bao gồm của khách hàng sở hữu dữ liệu từ một thiết bị duy nhất và máy chủ điều phối quá trình FL. Các phần sau cung cấp chi tiết thiết kế về từng thành phần tạo nên kiến trúc được đề xuất. Mã được sử dụng để triển khai toàn bộ đường ống hiện có tại [37] .



Hình 1: Kiến trúc khung và các thành phần. Các chia sẻ các giá trị chuẩn hóa và lựa chọn siêu tham số cộng tác được bỏ qua để đơn giản hóa.

4.1. Client

Xem xét rằng các thiết bị IoT thường có nguồn re hạn chế và độ tin cậy khiêm tốn, các khách hàng phụ trách đào tạo các mô hình không phải là thiết bị được bảo vệ, mà là các quyền khác có khả năng thu thập lưu lượng truy cập của các thiết bị IoT hiện có trong cùng một mạng, chẳng hạn như các trạm gốc B5G hoặc truy cập khác điểm. Theo nghĩa này, trong kiến trúc B5G [38], hiện tại hệ thống sẽ được tích hợp trong RAN SLICING Edge Các nút hoặc trong các nút sương mù TRƯỢT ĐÁM MÂY. Hệ thống này thuộc loại FL-silo chéo, như được định nghĩa trong [15], trong đó khách hàng được liên kết rất ít nhưng mạnh mẽ và đáng tin cậy.Lưu ý rằng mỗi khách hàng có thể sở hữu một số thiết bị IoT, nhưng đối với vì sự đơn giản, kiến trúc và các thử nghiệm

được mô tả với một cái duy nhất cho mỗi khách hàng. Hình 2 mô tả chi tiết kiến trúc của khách hàng sau khi thu thập dữ liệu, cũng như tương tác với máy chủ. Tập dữ liệu và các thành phần được mô tả trong hình được đề cập ở trên được giải thích chi tiết trong phần còn lại của phần này.

4.1.1. Thu thập dữ liệu

4.1.2. Dataset

4.1.3. Xử lý dữ liệu

4.1.4. Model training

4.1.5. Đánh giá model

4.2. Server

Trong kiến trúc khung được đề xuất, máy chủ chịu trách nhiệm điều phối các nỗ lực đào tạo của các máy khách liên kết. Cụ thể, nó khởi tạo mô hình ngay từ đầu và nó tổng hợp các mô hình do khách hàng gửi vào một mô hình được gọi là toàn cầu. Nó cũng phải phối hợp các bước bổ sung được mô tả trong Phần 4.3, tức là chuẩn hóa cộng tác, tìm kiếm lưới cộng tác và lựa chọn ngưỡng cộng tác (đối với phương pháp phát hiện bất thường). Trong bối cảnh kiến trúc B5G [38], thành phần máy chủ sẽ được đặt trong lớp CLOUD SLICING, trên Fog Nodes hoặc trong Cloud Data Center, đang chờ xử lý trên phạm vi của các máy khách được bao phủ.

4.2.1. Khởi tạo model

4.2.2.Tổng hợp model

4.3. Mối quan tâm bổ sung của Khung đề xuất

Phần này tóm tắt một số mối quan tâm bổ sung phát sinh khi thực hiện đường ống dẫn đầy đủ thông thường của ML theo cách liên kết. Cụ thể, các bước chuẩn hóa và lựa chọn siêu tham số phải được chú ý. Bên cạnh đó, đối với giải pháp không có giám sát, bước lựa chọn ngưỡng cũng cần phải xem xét đặc biệt.

4.3.1. Chuẩn hóa cộng tác vì tính năng chia tỷ lệ tối

4.3.2. Tìm kiếm theo lưới cộng tác Cần

4.3.3. Lựa chọn ngưỡng cộng tác

5. Các cuộc tấn công và biện pháp đối phó:

Phần này cung cấp nền tảng lý thuyết liên quan đến một số vụ tấn công đầu độc nổi tiếng nhất, có ý định làm giảm hiệu suất của mô hình. Bên cạnh đó, nó còn mô tả các chức năng tổng hợp mô hình khác nhau có thể cải thiện khả năng phục hồi của mô hình chống lại các cuộc tấn công.

5.1. Các cuộc tấn công

Một máy chủ đáng tin cậy, phần lớn khách hàng đáng tin cậy và một số ít khách hàng nguy hiểm có khả năng thông đồng với nhau được giả định qua phần giải thích sau đây.

Các máy khách độc hại như vậy thường được gọi là Byzantine Workers. Máy chủ và những người tham gia đáng tin cậy có thể cũng được coi là tin cậy nhưng tò mò (cố gắng suy luận càng nhiều thông tin càng tốt mà không làm lệch giao thức), nhưng các vấn đề về quyền riêng tư nằm ngoài phạm vi của công việc này. Tiếp theo, một số cuộc tấn công đầu độc dữ liệu và mô hình được mô tả, thực hiện và đánh giá sau này trong Phần 6.

Attack name	Poisoning	Need data	Attacker's objective
Benign label flipping	Data	Yes	TNR = 0
Attack label flipping	Data	Yes	TPR = 0
All labels flipping	Data	Yes	Acc. = 0
Gradient factor	Model	Yes	Acc. = 0
Model cancelling	Model	No	$w^{(i)} = 0$, $\forall i \in [d]$

Hình 1: Đặc điểm tấn công đối phương. Các chỉ số được hiển thị ở đây được định nghĩa trong Phần 6.

Các đặc điểm của các cuộc tấn công được mô tả được tóm tắt trong hình 1. Các cuộc tấn công này đã được lựa chọn vì tính đơn giản và đa dạng của chúng, nhưng các cuộc tấn công lén lút và tinh vi hơn khác vẫn tồn tại.

Các cuộc tấn công nhiễm độc dữ liệu hoạt động thông qua phương tiện của tập dữ liệu máy khách. Máy khách có thể độc hại và cố tình sửa đổi dữ liệu của chính nó với mục đích làm cho nó gây hiểu lầm. Ngay cả khi khách hàng đáng tin cậy, cuộc tấn công có thể đến từ bất kỳ phần nào trong đường dẫn dữ liệu máy khách mà thực thể độc hại bên ngoài có quyền kiểm soát. Do đó, loại tấn công này là loại tấn công giả định ít hơn từ các khách hàng và điều đó có khả năng xảy ra cao nhất. Ba cuộc tấn công nhiễm độc dữ liệu, tất cả đều dựa trên việc lật nhãn , được mô tả cho tình huống được giám sát.

- Lật nhãn lành tính. Ở đây, các nhãn 0 (lành tính) được lật thành 1s (tấn công). Mục tiêu của kẻ tấn công làm điều này là làm cho mô hình luôn phân loại lưu lượng truy cập là tấn công và làm cho nó có TNR là 0%. Một mô hình như vậy sẽ liên tục đưa ra các cảnh báo sai và có thể rất đáng lo ngại cho người dùng của nó.

- Tấn công lật nhãn. Trong trường hợp này, các nhãn 1 được lật thành 0, với mục tiêu làm cho mô hình đạt TPR 0%. Một mô hình như vậy sẽ không bao giờ đưa ra cảnh báo về lưu lượng tấn công và sẽ cho phép phần mềm độc hại tiềm ẩn không bị phát hiện.

- Tất cả các nhãn đang lật. Trong cuộc tấn công này, tất cả các nhãn đều bị lật, tức là số 1 trở thành số 0 và số 0 trở thành số 1. Mục tiêu của một cuộc tấn công như vậy là đưa độ chính xác của mô hình về 0%, kết hợp cả hai cuộc tấn công trước đó.

  Lưu ý rằng hai cuộc tấn công đầu tiên được coi là có mục tiêu vì chúng tập trung vào một lớp cụ thể, trong khi cuộc tấn công thứ ba được coi là không có mục tiêu vì nó tập trung vào cả hai lớp. Tất cả các cuộc tấn công này được tham số hóa bằng tỷ lệ p_{poison} của các nhãn nhắm mục tiêu được lật.

Các cuộc tấn công đầu độc mô hình được tiến hành thông qua các bản cập nhật mô hình bị hỏng được gửi đến máy chủ. Chúng là một vấn đề rất lớn khi sử dụng học liên kết vì các máy khách có thể gửi các mô hình xấu tùy ý đến máy chủ, và do tính riêng tư mà học liên kết cung cấp, rất khó để kiểm tra xem các mô hình nhận được có thực sự tương ứng với dữ liệu đào tạo cục bộ hay không. Theo một nghĩa nào đó, các cuộc tấn công đầu độc dữ liệu có thể được coi là một tập hợp con của các cuộc tấn công đầu độc mô hình bởi vì việc huấn luyện từ dữ liệu sai tạo ra một mô hình sai. Tiếp theo, một số cuộc tấn công đầu độc mô hình cơ bản nhất được mô tả:

- Tấn công yếu tố Gradient. Trong trường hợp này, các máy khách độc hại nhân gradient của chúng với hệ số âm a_{grad} trước khi cập nhật mô hình cục bộ của chúng và chia sẻ nó với máy chủ. Trong cuộc tấn công này các máy khách độc hại chỉ đơn giản là có quyền truy cập vào dữ liệu từ một thiết bị. Do đó, họ chỉ có thể tính toán ước tính

của gradient trên phân phối dữ liệu của riêng họ. Với tổng số K khách hàng trong đó f là độc hại, hệ số a_{grad} được chọn để xác minh:

Cụ thể, các máy khách độc hại chọn hệ số cập nhật của họ để hệ số cập nhật trung bình bao gồm các máy khách đáng tin cậy là -1 (thay vì 1 trong trường hợp không đối nghịch). Lưu ý rằng việc chọn một giá trị a_{grad} giải phương trình 2 là không cần thiết để thực hiện cuộc tấn công này (bất kỳ giá trị âm nào cũng có thể được xem xét).

- Cuộc tấn công hủy bỏ mô hình. Trong cuộc tấn công này, các máy khách độc hại cố gắng đưa tất cả các tham số của mô hình toàn cục về giá trị 0. Họ chọn mô hình của mình theo cách mà khi được tính trung bình với các mô hình máy khách đáng tin cậy, mô hình toàn cầu ban đầu sẽ biến mất, tức là $w^{(i)} = 0,\ \forall i \in [d]$. Chỉ có bản cập nhật gần đây nhất từ khách hàng đáng tin cậy vẫn còn. Để đạt được điều đó, họ chỉ cần xuất ra các tham số mô hình toàn cục ban đầu, nhân với hệ số a_{param} phải thỏa mãn:

  Cụ thể, a_{param} phải được chọn sao cho trọng lượng của các máy khách độc hại ($a_{param} \cdot f$) hủy bỏ trọng lượng của các máy khách đáng tin cậy ($K - f$). Lưu ý rằng lần này, việc sử dụng đúng giá trị của a quan trọng hơn nhiều, vì vậy cần có sự thông đồng giữa các máy khách độc hại để chúng biết số chính xác (f) của chúng lúc đầu. Cuộc tấn công này rất mạnh mẽ, nhưng đồng thời, nó không phải là lén lút, vì các giá trị được đưa ra bởi các máy khách độc hại rất khác so với những giá trị thường được mong đợi về hướng và độ lớn.

5.2. Các chức năng tổng hợp mô hình mạnh mẽ

Có nhiều cách khác nhau để đảm bảo an toàn cho hệ thống trước các cuộc tấn công. Một trong những ý tưởng mở rộng nhất là sử dụng tổng hợp mô hình và cập nhật các giải pháp xử lý có tính đến khả năng khách hàng độc hại cố gắng chiếm đoạt mô hình. Tiếp theo, hai hàm tổng hợp khác nhau, ngoài tính trung bình (AVG), được xác định cũng như một bước trước để áp dụng cho các mô hình do khách hàng gửi. Hầu hết các bằng chứng hội tụ của các hàm tổng hợp này không được lưu giữ trong công việc này bởi vì các tập dữ liệu máy khách không phải từ cùng một bản phân phối. Tuy nhiên, trực giác đằng sau việc sử dụng các chức năng này vẫn giống nhau. Các biện pháp đối phó này đã được lựa chọn vì tính đơn giản tuyệt vời, vì chúng chỉ yêu cầu sửa đổi bước tổng hợp mô hình, điều này rất dễ thực hiện. Chúng cũng không yêu cầu bất kỳ kiến thức nào trước đây về việc phân phối dữ liệu của khách hàng, điều này khó có được trong cài đặt liên kết thực tế.

Coordinate-wise median. Hàm tổng hợp này áp dụng giá trị trung bình cho từng tham số riêng lẻ để loại trừ hoàn toàn bất kỳ giá trị ngoại lai tiềm ẩn nào. Tọa độ của i^{th} được cho bởi $w^{(i)} = med\ \{\}$. Lưu ý rằng định nghĩa thông thường về giá trị trung bình được sử dụng, tức là khi K lẻ, giá trị giữa được chọn và khi K chẵn, giá trị trung bình giữa hai giá trị giữa được lấy. Chúng tôi gọi hàm tổng hợp này là MED.

Coordinate-wise trimmed mean. Giá trị trung bình bị cắt xén, có thể được coi là sự thỏa hiệp giữa giá trị trung bình và giá trị trung vị. Đối với mỗi tọa độ $i \in [d]$, một phần của giá trị lớn nhất và nhỏ nhất bị loại bỏ trước khi tính giá trị trung bình. Do số lượng khách hàng thấp trong kịch bản mà chúng tôi xem xét, thuật toán trung bình được cắt bớt được xác định lại bằng cách sử dụng số nguyên c của các giá trị lớn nhất và thấp nhất bị loại trừ thay vì tỷ lệ, nhưng cả hai đều tương đương. Do đó, theo định nghĩa của chúng ta, tọa độ i^{th} của w được cho bởi

, trong đó $U^{(i)}$ là một tập con của {} nhận được bằng cách loại bỏ c lớn nhất và c nhỏ nhất trong số các phần tử của nó. Số phần tử bị loại là $2c$. Hàm tổng hợp này được gọi là TM (c).

s-Resampling. Thay vì là một hàm tổng hợp, s-Resampling là một bước bổ sung có thể được thực hiện trước khi tổng hợp. Trong trường hợp tập dữ liệu của mỗi khách hàng có bản phân phối riêng, tập dữ liệu này nhằm mục đích giảm sự không đồng nhất của các mô hình do mỗi khách hàng gửi. Do đó, s-Resampling có nghĩa là được kết hợp với một chức năng tổng hợp mạnh mẽ để giảm tác dụng phụ của việc sử dụng một chức năng như vậy trên các mô hình nonIID. Lưu ý rằng việc kết hợp s-Resampling với AVG là vô ích, vì kết quả luôn giống hệt như khi chỉ sử dụng AVG. Nó hoạt động bằng cách thay thế từng mô hình bằng giá trị trung bình giữa các mô hình s được lấy mẫu ngẫu nhiên từ mô hình K khách hàng. Mỗi mô hình có thể được lấy mẫu tổng cộng tối đa s lần. s-Resampling cũng có thể làm cho các mô hình độc hại bị pha loãng thành một số mô hình mà nó xuất ra, làm tăng phạm vi tiếp cận của các máy khách độc hại. Vì lý do đó, nó chỉ được kỳ vọng sẽ hoạt động hài lòng với một số lượng nhỏ máy khách độc hại, giá trị s nhỏ và với chức năng tổng hợp loại bỏ một số giá trị cực cao, chẳng hạn như MED hoặc TM (2).

6. Kết quả thực nghiệm

Phần này trình bày chi tiết các kết quả thu được trong các thí nghiệm khác nhau được thực hiện để xác nhận khung đề xuất. Đầu tiên, nó so sánh hiệu suất khi phát hiện phần mềm độc hại giữa các phương pháp tiếp cận được liên kết và truyền thống khi thực hiện theo cả giải pháp được giám sát hoặc không được giám sát. Sau đó, nó cho thấy tác động của các cuộc tấn công đối thủ được đề xuất trong Phần 5 và cách các chức năng tổng hợp khác nhau giảm thiểu các cuộc tấn công đó.

Các chỉ số được sử dụng để đánh giá và so sánh hiệu suất của từng cách tiếp cận như sau (TP: Khẳng định đúng, TN: Phủ định đúng, FP: Khẳng định sai, FN: Phủ định sai, TPR: Tỷ lệ khẳng định đúng, TNR: Tỷ lệ phủ định đúng):

- TPR =
- TNR =
- Độ chính xác =
- Điểm F1 =

Tất cả các thí nghiệm được thực hiện trong phần này đều tuân theo một phương pháp tương tự. Theo nghĩa này, liên kết bao gồm $K = 8$ máy khách, mỗi máy sở hữu dữ liệu của một trong 9 thiết bị có sẵn trong tập dữ liệu N-BaIoT. Dữ liệu của một thiết bị không được sử dụng trong quá trình đào tạo, giữ nó như một thiết bị vô hình cho mục đích thử nghiệm. Trong bối cảnh này, chín tổ hợp thiết bị khác nhau (với một thiết bị không nhìn thấy) đã được sử dụng trong tất cả các thử nghiệm. Hơn nữa, các thí nghiệm được lặp lại 5 lần để cải thiện tính nhất quán của kết quả. Cuối cùng, kết quả của mỗi thử nghiệm cho thấy tổng số trung bình trên 45 lần chạy (9 thiết bị không nhìn thấy có thể có và 5 lần thực thi).

6.1. Hiệu suất của Học tập liên kết và truyền thống để phát hiện phần mềm độc hại trong các thiết bị IoT

Thử nghiệm này nhằm đo lường hiệu suất của giải pháp của chúng tôi khi phát hiện phần mềm độc hại IoT bằng cách sử dụng tập dữ liệu N-BaIoT. Để xác minh rằng phương pháp học tập liên kết phù hợp với kịch bản phần mềm độc hại IoT của chúng tôi một cách chính xác, cần phải so sánh nó với các giải pháp truyền thống. Cụ thể hơn, các lựa chọn thay thế được so sánh là:

- Naive decentralized approach: Mỗi khách hàng sử dụng tập dữ liệu cục bộ của mình để đào tạo và thử nghiệm. Vì mỗi khách hàng sản

xuất mô hình riêng của mình, kết quả được so sánh bằng cách lấy trung bình hiệu suất của mỗi khách hàng.

- Centralized approach. Tất cả dữ liệu đào tạo được chia sẻ với một máy chủ phụ trách đào tạo và thử nghiệm một mô hình với nó. Nó không bảo vệ sự riêng tư.
- Liên kết với MINI-BATCH AVG. Các máy khách khác nhau cộng tác để tạo ra một mô hình toàn cục bằng cách sử dụng thuật toán MINI-BATCH AGGREGATION với chức năng tổng hợp AVG.
- Liên kết với MULTI-EPOCH AVG. Tương tự như cách tiếp cận trước nhưng đào tạo với thuật toán MULTI-EPOCH AGGREGATION để giảm đáng kể chi phí truyền thông.

Các bước sau được thực hiện cho cả giải pháp được giám sát và không được giám sát. Đầu tiên, hai siêu tham số quan trọng (kiến trúc của mô hình và L2 - giá trị chính quy λ) được chọn cho mỗi thiết lập bằng cách sử dụng tìm kiếm lưới. Các kiến trúc MLP và tự động mã hóa được xem xét là những kiến trúc được mô tả trong Phần 4.1.4. Các giá trị được xét cho λ là 0, 10^{-5} và 10^{-4}. Lưu ý rằng đối với phương pháp ngây thơ, các siêu tham số được chọn cho mỗi máy khách vì các máy khách không cộng tác trong việc chọn siêu tham số. Đối với các phương pháp tiếp cận học liên kết, mỗi liên đoàn đã sử dụng các tìm kiếm lưới cộng tác để chọn các siêu tham số, như được định nghĩa trong Phần 4.3.2. Cuối cùng, trong phương pháp tập trung, tìm kiếm lưới được thực hiện trực tiếp bởi máy chủ nhận toàn bộ tập dữ liệu. Đối với tất cả các thử nghiệm, kích thước lô $B = 64$ đã được sử dụng khi đào tạo, ngoại trừ với MINI-BATCH AVG trong đó kích thước lô được chia cho số lượng khách hàng ($B = 8$), để mỗi bản cập nhật mô hình được thực hiện với tổng số 64 mẫu là tốt. Trong tất cả các thử nghiệm, các cập nhật mô hình được tính toán với Stochastic Gradient Descent (SGD). Đối với giải pháp có giám sát, quá

trình huấn luyện được thực hiện trong $E = 4$ kỷ nguyên; đối với giải pháp không giám sát, nó được tạo ra với $E = 120$ kỷ nguyên.

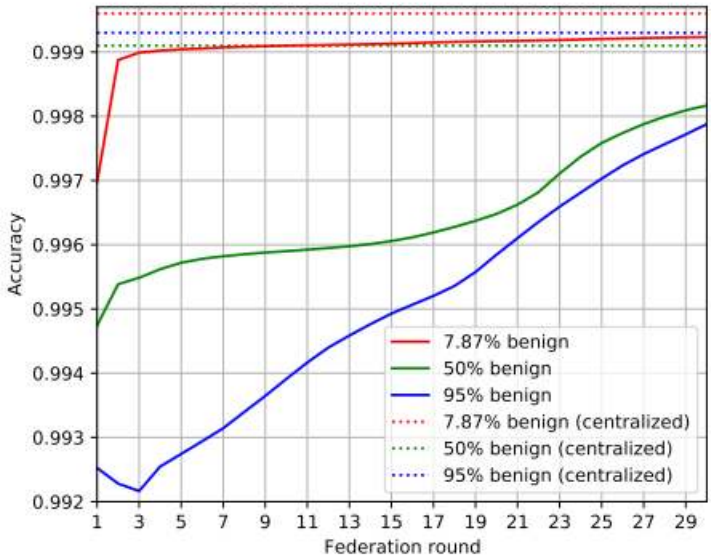
Tình huống được giám sát. Đầu tiên, giải pháp được giám sát được xác minh. Ở đây, ba tùy chọn tách tập dữ liệu khác nhau được giải thích trong Phần 4.1.2 (dữ liệu lành tính 7,87%, 50% và 95%) được sử dụng trong các thử nghiệm lặp lại, cũng để kiểm tra xem các số dư lớp khác nhau ảnh hưởng như thế nào đến kết quả. Hình 2 cho thấy kết quả đạt được trong các thí nghiệm này.

		Naive	Multi-epoch avg	Mini-batch avg	Central.
Known devices (7.87%)	Acc.	99.78	99.92	99.96	99.96
	TPR	99.98	99.98	99.98	99.98
	TNR	97.49	99.26	99.69	99.69
New device (7.87%)	Acc.	98.94	99.89	99.89	99.88
	TPR	99.58	99.97	99.98	99.97
	TNR	91.44	99.00	98.93	98.83
Known devices (50%)	Acc.	99.92	99.82	99.93	99.91
	TPR	99.97	99.97	99.97	99.97
	TNR	99.88	99.67	99.88	99.85
New device (50%)	Acc.	98.36	99.63	99.58	99.52
	TPR	98.79	99.90	99.95	99.93
	TNR	97.93	99.35	99.21	99.10
Known devices (95%)	Acc.	99.92	99.79	99.93	99.93
	TPR	99.89	99.93	99.98	99.98
	TNR	99.92	99.78	99.92	99.93
New device (95%)	Acc.	98.59	99.43	99.38	99.42
	TPR	97.79	99.55	99.82	99.81
	TNR	98.63	99.42	99.36	99.40

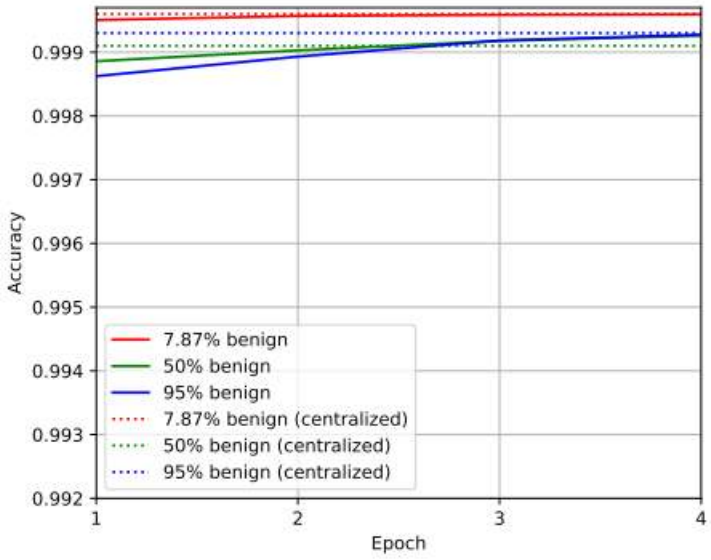
Hình 2: Kết quả được giám sát so sánh cả hai phương pháp tiếp cận FL (trung bình Multiepoch và trung bình theo lô nhỏ) với phương pháp đơn giản và phương pháp tập trung. Phần trăm dữ liệu lành tính của các bộ dữ liệu được sử dụng được biểu thị bằng màu ở bên trái.

Kết quả đáng chú ý đầu tiên là hiệu suất của phương pháp tập trung cao hơn phương pháp ngây thơ phân tán, đặc biệt khi được đánh giá trên một thiết bị không

nhìn thấy. Hơn nữa, trên cả ba cài đặt tập dữ liệu, kết quả MINI-BATCH AVG rất gần với kết quả tập trung, thậm chí đôi khi vượt quá chúng. Mặc dù việc thu được kết quả tốt hơn so với phương pháp tập trung có thể gây ngạc nhiên, nhưng điều này có thể được giải thích bởi một số yếu tố, chẳng hạn như tính ngẫu nhiên của các thí nghiệm hoặc thực tế là các siêu tham số được tính toán khác nhau. Hình 3 cho thấy tốc độ hội tụ của các mô hình gần hiệu suất tập trung.



(a) Over the federation round (MULTI-EPOCH AVG)



(b) Over the training epoch (MINI-BATCH AVG)

Hình 3: Sự phát triển của độ chính xác của các thiết bị đã biết. Độ chính xác thu được bằng các phương pháp tập trung được hiển thị bằng các đường chấm chấm để so sánh.

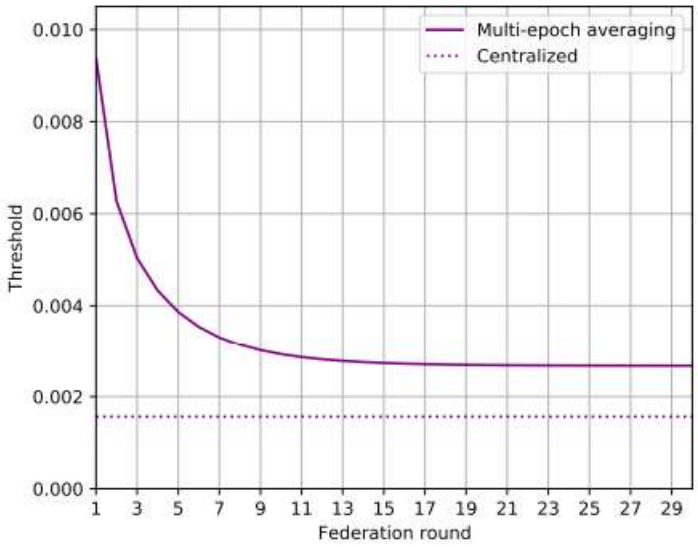
MULTI-EPOCH AVG cũng tạo ra kết quả khá hài lòng, với độ chính xác giảm không đáng kể trên các thiết bị đã biết, bù lại là độ chính xác luôn vượt quá mức tập trung trên thiết bị mới. Điều này có thể được giải thích là do tính trung bình của các tham số mô hình với hàm mất mát không lồi có thể có tác động làm hỏng mô hình một cách tùy ý. Tuy nhiên, nó cũng có thể được xem như một dạng cơ chế hoạt động chống lại việc trang bị quá mức, do đó cải thiện tính khái quát trên một thiết bị mới. Hơn nữa, những kết quả này có thể được cải thiện một chút bằng cách sử dụng số vòng liên đoàn T cao hơn. Hình 4 cho thấy rằng đối với các bộ dữ liệu có 50% và 95% dữ liệu lành tính, độ chính xác dường như chưa hội tụ chính xác sau 30 vòng.

Tình huống không được giám sát. Khi hiệu suất được giám sát được xác minh, bước tiếp theo là đánh giá hoạt động không được giám sát. Ở đây, chỉ lưu lượng lành tính được sử dụng để đào tạo, vì vậy mô hình cuối cùng không phụ thuộc vào số dư lớp trong tập dữ liệu. Để làm cho kết quả của chúng tôi độc lập với số dư lớp được sử dụng, chúng tôi chỉ hiển thị TPR và TNR cho giải pháp này (chứ không phải độ chính xác). Phương trình được sử dụng để xác định ngưỡng được mô tả trong Phần 4.1.4. Trong số các kiến trúc có thể có, kiến trúc đầu tiên (Autoencoder A) luôn là kiến trúc gây mất xác thực tốt nhất trong tất cả các lựa chọn siêu tham số. Tất cả các kết quả từ tình huống không được giám sát sẽ được tạo thêm bằng Autoencoder A. Hình 4 cho thấy các kết quả không được giám sát của hệ thống.

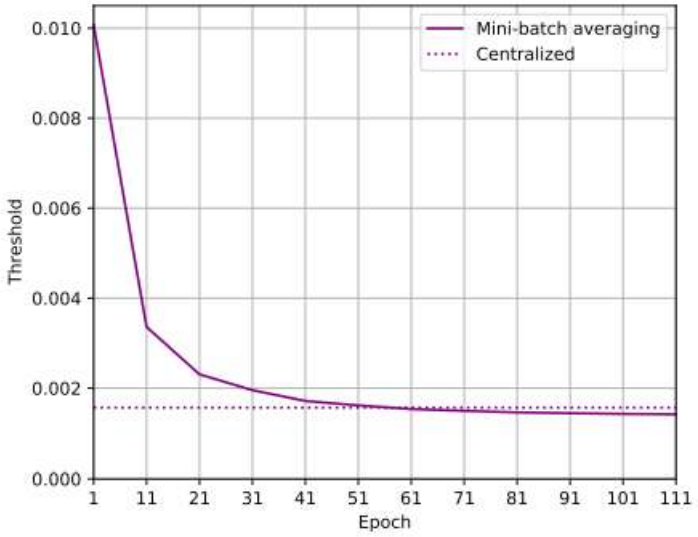
		Naive	MULTI-EPOCH AVG	MINI-BATCH AVG	Central.
Known devices	TPR	88.00	99.98	99.98	99.98
	TNR	97.38	94.84	95.12	95.56
New device	TPR	87.77	99.98	99.98	99.98
	TNR	59.66	92.61	91.78	92.76

Hình 4 Kết quả không giám sát so sánh cả hai phương pháp tiếp cận FL ((Multiepoch avg và Mini-batch avg) với phương pháp tiếp cận đơn giản và phương pháp tiếp cận tập trung.

Ở đây, tập trung dữ liệu cho thấy tổng thể một sự cải thiện hiệu suất cao so với phương pháp ngây thơ. Hơn nữa, các thuật toán FL cũng giải quyết rất thành công nhiệm vụ lấy dấu vân tay không được giám sát. Cụ thể, hiệu suất tập trung gần như đạt được bởi cả hai phương pháp MULTIEPOCH AVG và MINI-BATCH AVG. Một lần nữa, MULTIEPOCH AVG dường như giúp mô hình tổng quát hóa tốt hơn, vì nó thể hiện trên thiết bị mới TNR tốt hơn một chút so với MINI-BATCH AVG. Điều thú vị là ngưỡng, như được hiển thị trong Hình 5, hội tụ thành một giá trị lớn hơn, trong trường hợp MULTIEPOCH AVG so với những gì mà phương pháp tập trung đạt được. Như đã giải thích trước đó, lựa chọn ngưỡng cộng tác không tương đương với việc chọn ngưỡng trực tiếp trên toàn bộ tập hợp ngưỡng lựa chọn (như trong phương pháp tập trung), vì vậy kết quả này không có gì đáng ngạc nhiên.



(a) Over the federation round (MULTI-EPOCH AVG)



(b) Over the training epoch (MINI-BATCH AVG)

Hình 5: Sự phát triển của các giá trị ngưỡng toàn cục với cả hai thuật toán học liên kết. Ngưỡng thu được trong phương pháp tập trung được hiển thị bằng một đường chấm chấm để so sánh.

Với các thử nghiệm trước đó, người ta đã xác minh rằng trong kịch bản phát hiện phần mềm độc hại trong các thiết bị IoT cụ thể này, việc sử dụng nhiều dữ liệu hơn để đào tạo mô hình mang lại một cải tiến đáng kể, đặc biệt là trên các thiết bị chưa

từng thấy trước đây. Bên cạnh đó, đào tạo dựa trên học liên kết thành công đạt được hiệu suất tập trung theo cách bảo vệ quyền riêng tư.

6.2. Tác động của các cuộc tấn công bất lợi và các biện pháp đối phó khi phát hiện phần mềm độc hại

Khi hiệu suất của phương pháp liên hợp đã được xác minh, bước tiếp theo là đánh giá xem các cuộc tấn công đối thủ khác nhau được đề xuất trong Phần 5 ảnh hưởng như thế nào đến phương pháp liên kết. Bên cạnh đó, các hàm tổng hợp khác nhau được áp dụng để kiểm tra xem chúng cải thiện khả năng phục hồi của mô hình như thế nào trước các cuộc tấn công khác nhau. Để ngắn gọn, các thử nghiệm này tập trung vào tình huống được giám sát và chỉ sử dụng số dư tập dữ liệu với 95% dữ liệu lành tính. Hơn nữa, chúng được thực hiện với thuật toán liên hợp MINI-BATCH AGGREGATION. Kích thước lô $B = 64$ (thay vì $B = 8$) được sử dụng cho tất cả các thử nghiệm đối nghịch, vì nó cho phép cập nhật mượt mà hơn cho các chức năng tổng hợp mạnh mẽ.

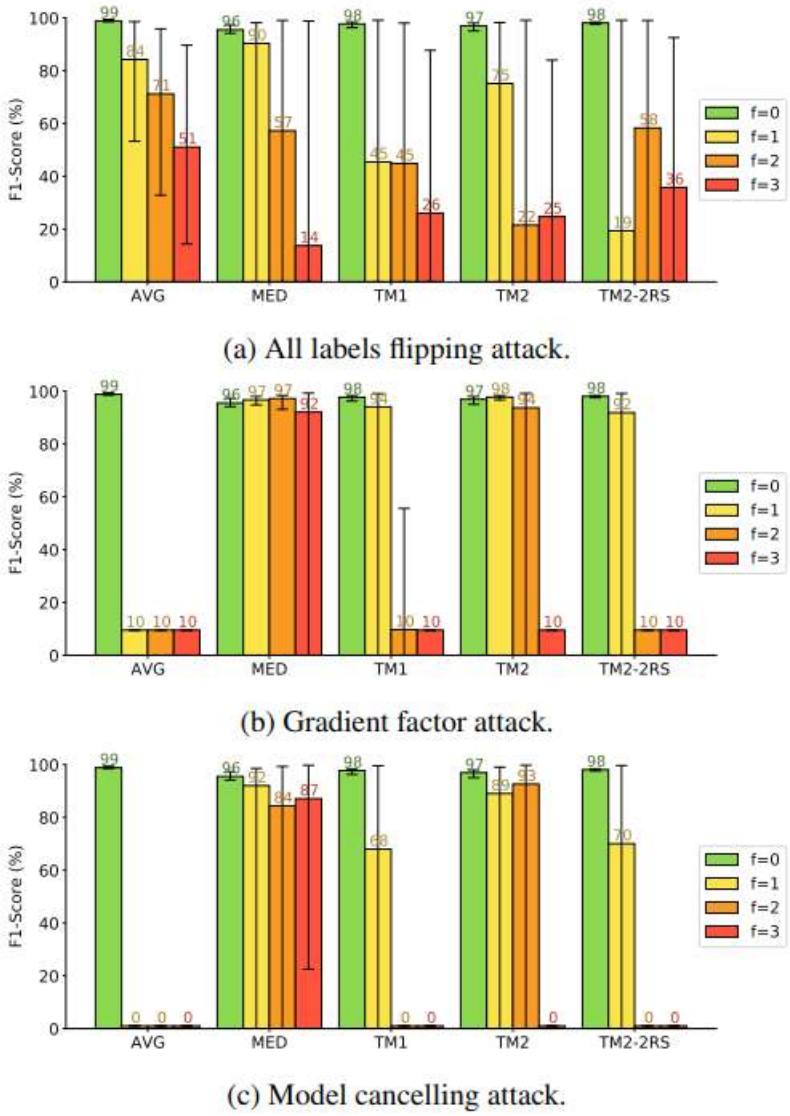
Như đã giải thích trước đó, s-Resampling chỉ hoạt động với một giá trị nhỏ của s và được kết hợp với MED hoặc TM (2) (hoặc các hàm tổng hợp mạnh mẽ khác mà chúng tôi không triển khai). Vì TM (2) tính toán đầu ra của nó bằng cách tính đến nhiều giá trị hơn MED, nên s-Resampling chỉ được thử nghiệm cho TM (2) và với $s = 2$. Sự kết hợp này được gọi là TM (2) ◦ 2-Resampling.

Trong các thử nghiệm thực hiện các cuộc tấn công nhiễm độc dữ liệu, cuộc tấn công lật Tất cả các nhãn được chọn để thử nghiệm, vì nó kết hợp cả việc lật ngược nhãn lành tính và tấn công. Vì trọng tâm được đặt vào việc đầu độc dữ liệu có chủ đích, nên $p_{poison} = 1$ luôn được sử dụng. Cách tiếp cận này cho phép xác minh tác động tối đa của cuộc tấn công trong mô hình đã tạo.

Về tấn công ngộ độc mô hình, trong trường hợp tấn công yếu tố gradient, giải phương trình 2 cho $a_{grad} =$. Đối với tổng số 8 máy khách bao gồm 1, 2 và 3 máy khách độc hại, các giá trị được chọn cho a_{grad} lần lượt là −15, −7 và - . Trong

trường hợp tấn công hủy bỏ mô hình, giải phương trình 3 cho a_{param} = . Đối với tổng số 8 máy khách bao gồm 1, 2 và 3 máy khách độc hại, các giá trị được chọn cho a_{param} lần lượt là −7, −3 và - .

Hình 6 cho thấy điểm F1 của mô hình được thử nghiệm trên các thiết bị do khách hàng sở hữu (các thiết bị đã biết) thay đổi như thế nào trong các cuộc tấn công được triển khai khác nhau theo chức năng tổng hợp. Nó cũng cho thấy sự phát triển của số liệu khi số lượng khách hàng độc hại tăng từ 0 lên 3 (hoặc 37,5% tổng số khách hàng trong thiết lập). Cần lưu ý rằng những kết quả này có một phương sai lớn do sự ngẫu nhiên của việc lựa chọn ứng dụng khách nào là độc hại. Mặc dù mỗi thử nghiệm đã được chạy tổng cộng 45 lần, điều này có thể dẫn đến kết quả không lường trước được, chẳng hạn như đôi khi có điểm F1 trung bình tốt hơn với nhiều ứng dụng khách độc hại hơn. Tuy nhiên, những thí nghiệm này là đủ để có được một ý tưởng có cơ sở về mức độ nghiêm trọng của vấn đề đối thủ.



(a) All labels flipping attack.

(b) Gradient factor attack.

(c) Model cancelling attack.

Hình 6: Điểm F1 trong các cuộc tấn công được thử nghiệm khác nhau cho từng chức năng tổng hợp, với f = 0, 1, 2 hoặc 3 máy khách độc hại (tương ứng là 0%, 12,5%, 25% và 37,5% tổng số máy khách). Giá trị tối thiểu và tối đa (trên 45 lần chạy) của Điểm F1 được hiển thị bằng các thanh giới hạn.

Như chúng ta có thể quan sát, tính trung bình (AVG) là chức năng tổng hợp tốt nhất khi tất cả các khách hàng đều đáng tin cậy. Tuy nhiên, khi các máy khách độc hại có liên quan, hiệu suất của nó bị ảnh hưởng nặng nề tùy thuộc vào cuộc tấn công. Cụ thể, dưới yếu tố gradient và các cuộc tấn công hủy bỏ mô hình, ngay cả

một ứng dụng khách độc hại duy nhất cũng đủ để biến mô hình thành một công cụ dự đoán không đổi (lưu ý rằng một công cụ dự đoán tích cực không đổi có Điểm F1 là ∼10% và một công cụ dự đoán tiêu cực không đổi có F1-Điểm 0%). Điều này cho thấy sự cần thiết của việc sử dụng các phương pháp mạnh mẽ hơn khi giả định một mô hình mối đe dọa trong đó chỉ một ứng dụng khách cũng có thể là độc hại.

Coordinate-wise median aggregation (MED) thể hiện khả năng phục hồi tốt hơn trước hầu hết các tình huống tấn công được xem xét. Nhìn chung, nó có kết quả tốt nhất trong số các chức năng tổng hợp được thử nghiệm trong thiết lập đối thủ. Tuy nhiên, điều này vẫn chưa đủ mạnh, đặc biệt là khi xem xét 3 ứng dụng khách độc hại, vì cuộc tấn công lật tất cả các nhãn khiến F1-Score của nó đạt giá trị trung bình khoảng 14%. Ngay cả với một ứng dụng khách độc hại, khi thực hiện tất cả các cuộc tấn công lật nhãn và hủy mô hình, mặc dù Điểm F1 trung bình lần lượt là 90% và 92%, giá trị tối thiểu của chúng trong 45 lần chạy là 0% trong cả hai trường hợp, khiến nó vẫn không đáng tin cậy.

Không có gì đáng ngạc nhiên, *Coordinate-wise trimmed mean* (TM(c)) không thành công khi được sử dụng chống lại nhiều hơn c máy khách độc hại, như được minh họa rõ ràng trong mô hình hủy kết quả tấn công (Hình 6c). Tuy nhiên, điều đó không có nghĩa là chức năng tổng hợp này hoạt động tốt khi $c \geq f$, vì Điểm F1-tối thiểu đạt 0% ngay cả đối với một ứng dụng khách độc hại trong cuộc tấn công lật tất cả các nhãn (Hình 6a). Lợi ích duy nhất của TM (1) và TM (2) so với MED nằm ở hiệu suất khi không có ứng dụng khách độc hại nào tham gia, điều này tốt hơn một chút vì có nhiều tham số hơn được xem xét trong quá trình tính toán mô hình toàn cục. Lợi thế này có thể cao hơn trong trường hợp sử dụng với nhiều khách hàng hơn, nhưng trong trường hợp của chúng tôi, nó quá thấp để biện minh cho việc sử dụng TM.

Cuối cùng, 2-Resampling cho thấy sự cải thiện về độ chính xác trên các thiết bị đã biết khi không có ứng dụng khách độc hại nào tham gia. Tuy nhiên, điều này đi kèm với cái giá là giảm độ mạnh của hệ thống, khiến TM (2) ◦ 2-Resampling có kết quả tương tự như TM (1) hầu hết thời gian. Tuy nhiên, một cải tiến nhỏ so với TM (2), thể hiện trong Hình 6a, phải được lưu ý với 2 và 3 máy khách độc hại. Tương tự như TM, s-Resampling không cung cấp đủ lợi thế để được sử dụng trong một liên kết nhỏ như vậy, nhưng nó có thể trở nên hữu ích hơn ở quy mô lớn hơn.

Theo nhận xét chung, mặc dù khả năng phục hồi của các mô hình đã được cải thiện đáng kể bằng cách sử dụng MED trong các cuộc tấn công ngộ độc mô hình (yếu tố gradient và các cuộc tấn công hủy mô hình), hiệu suất của mô hình vẫn bị giảm đáng kể. Ngoài ra, trong trường hợp tấn công lật tất cả các nhãn, AVG vẫn hoạt động tốt hơn các chức năng khác nhằm cải thiện độ mạnh của mô hình. Những kết quả này cho thấy, mặc dù hiệu suất của mô hình đã được cải thiện, vẫn cần nghiên cứu thêm về các hàm tổng hợp có khả năng chống lại các cuộc tấn công của đối thủ. Chúng tôi tin rằng trong trường hợp các cuộc tấn công khác ảnh hưởng lớn đến trọng số, kết quả sẽ tương tự như các cuộc tấn công bằng yếu tố gradient và các cuộc tấn công hủy mô hình. Tuy nhiên, vẫn chưa biết hiệu suất sẽ bị ảnh hưởng như thế nào trong trường hợp các cuộc tấn công lén lút và tinh vi hơn khác.

7. Thảo luận

Phần này thảo luận về các khía cạnh liên quan của hiệu suất và thiết kế kiến trúc phải được xem xét khi triển khai trên môi trường B5G thực. Mặc dù hiệu suất trong các thử nghiệm phát hiện phần mềm độc hại đã được chứng minh là cao, nhưng các khía cạnh như chi phí truyền thông hoặc tập trung khuôn khổ cần được thảo luận.

7.1. Số lượng khách hàng và kết quả đối đầu

Một trong những hạn chế trong thử nghiệm là số lượng khách hàng được sử dụng thấp, 8 để đào tạo, do sự sẵn có của các bộ dữ liệu thích hợp cho việc học liên kết.

Trong kịch bản B5G thực, việc triển khai thiết bị sẽ đạt tới 10 triệu thiết bị trên km2 theo yêu cầu của ITU (Liên minh Viễn thông Quốc tế) [42]. Tuy nhiên, chúng tôi cho rằng các thử nghiệm là hợp lệ vì mặc dù số lượng đối thủ thấp, cụ thể là 1, 2 và 3, nhưng tỷ lệ phần trăm mà chúng đại diện trên tổng số khách hàng thực hiện khóa đào tạo là tương đối cao, 12,5%, 25% và 37,5 %, tương ứng (xem Hình 6). Do đó, kết quả có thể được ngoại suy cho các môi trường có số lượng khách hàng lớn hơn nhiều nhưng trong đó các đối thủ chỉ chiếm một tỷ lệ nhỏ trong tổng số, không quá 50%. Ngoài ra, các thuật toán tổng hợp mạnh mẽ khác nên được kiểm tra vì các thuật toán hiện tại không cung cấp đủ khả năng phục hồi tấn công khi số lượng máy khách độc hại vượt quá 25%. Về vấn đề này, có những đề xuất thú vị về các thuật toán tổng hợp đã tính đến sự hiện diện có thể có của các máy khách độc hại và đánh giá các biến thể trong các mô hình mà nó chia sẻ.

7.2. Chi phí giao tiếp và tính toán

Mặc dù các yêu cầu về thông lượng B5G vượt xa yêu cầu của giải pháp đề xuất, vì khuôn khổ được thiết kế cho các máy khách đặt tại hoặc gần các điểm truy cập, chi phí giao tiếp và tính toán cần được xem xét. Chúng rất quan trọng để không ảnh hưởng đến hoạt động thường xuyên của các giao diện không dây của các đối tượng IoT và các phần tử mạng cung cấp quyền truy cập vào chúng.

MINI-BATCH AGGREGATION có chi phí truyền thông cao hơn nhiều so với MULTI-EPOCH AGGREGATION vì nó yêu cầu truyền mô hình cho mỗi khách hàng để đào tạo đầy đủ, trong đó B là kích thước lô, E là số kỷ nguyên và n_k là số các mẫu đào tạo của khách hàng k. Lưu ý rằng về chi phí tính toán, nó cũng cho biết số lần cập nhật mô hình cục bộ được thực hiện bởi mỗi khách hàng. Mặt khác, MULTI-EPOCH AGGREGATION chỉ yêu cầu các máy khách truyền mô hình đến máy chủ một lần mỗi vòng, với tổng số T lần truyền cho mỗi máy khách. Tuy nhiên, số lượng cập nhật mô hình cục bộ cũng lớn hơn T lần, tức là cho máy khách k.

Hình 7 cho thấy sự so sánh giữa cả hai thuật toán tổng hợp trong các thí nghiệm của Phần 6.1 về chi phí tính toán và truyền thông. MULTI-EPOCH AVG cho thấy chi phí truyền thông thấp hơn nhiều so với MINI-BATCH AVG, ít hơn ≈1300 lần trong trường hợp tiếp cận có giám sát và ≈2000 lần trong trường hợp không giám sát. Tuy nhiên, số lần lặp lại huấn luyện cục bộ cao hơn 3,75 lần đối với các siêu tham số đã được chọn. Thông lượng trong mạng 5G hoặc B5G thực phải đủ để triển khai bất kỳ thuật toán nào trong hai thuật toán. Ngoài ra, như đã nêu trong Phần 4, các máy khách khung sẽ là các trạm gốc B5G và các điểm truy cập khác, có sức mạnh tính toán tương đối cao. Tuy nhiên, nếu chi phí truyền thông trở thành một vấn đề quan trọng, sẽ là điều đương nhiên nếu bạn chọn phương pháp tiếp cận dựa trên MULTI-EPOCH AVG.

	MULTI-EPOCH AVG	MINI-BATCH AVG
Number of model transmissions	$T = 30$	$E \cdot \frac{n_k}{B}$ $= 4 \cdot 9875$ $= 39500$
Communication cost assuming a model of size 94 kB	2.82 MB	3.713 GB
Number of local training steps	$T \cdot E \cdot \frac{n_k}{B}$ $\simeq 30 \cdot 4 \cdot 1234$ $= 148080$	$E \cdot \frac{n_k}{B}$ $= 4 \cdot 9875$ $= 39500$

(a) Computation and communication costs in the supervised approach. When using MULTI-EPOCH AVG, $\frac{n_k}{B} = \frac{79000}{64} \simeq 1234$, and using MINI-BATCH AVG, $\frac{n_k}{B} = \frac{79000}{8} = 9875$.

	MULTI-EPOCH AVG	MINI-BATCH AVG
Number of model transmissions	$T = 30$	$E \cdot \frac{n_k}{B}$ $\simeq 120 \cdot 494$ $= 59280$
Communication cost assuming a model of size 27 kB	810 kB	1.6 GB
Number of local training steps	$T \cdot E \cdot \frac{n_k}{B}$ $\simeq 30 \cdot 120 \cdot 62$ $= 223200$	$E \cdot \frac{n_k}{B}$ $\simeq 120 \cdot 494$ $= 59280$

(b) Computation and communication costs in the unsupervised approach. When using MULTI-EPOCH AVG, $\frac{n_k}{B} = \frac{3950}{64} \simeq 62$, and using MINI-BATCH AVG, $\frac{n_k}{B} = \frac{3950}{8} \simeq 494$.

Hình 7. Chi phí tính toán và giao tiếp trên mỗi khách hàng. Chi phí truyền thông là từ quan điểm của khách hàng và phải được xem xét theo cả hai hướng (tải xuống và tải lên). Các kích thước mô hình giả định tương ứng với các kiến trúc lớn nhất đã được sử dụng trong các thử nghiệm của chúng tôi, cho cả phương pháp tiếp cận được giám sát và không được giám sát.

7.3. Phân quyền và bất đồng bộ

Mặc dù việc đào tạo các mô hình được phân cấp ở từng máy khách, nhưng việc có một máy chủ phụ trách tổng hợp mô hình có nhiều ưu điểm, chẳng hạn như kiểm soát mô hình chung được tạo ra, phối hợp giữa các máy khách,… Tuy nhiên, thiết kế này cũng mang lại một số nhược điểm.

Máy chủ trở thành điểm trung tâm của sự thất bại, nơi nút cổ chai hoặc cuộc tấn công có thể khiến không thể tổng hợp các mô hình cục bộ được nữa và chỉ có thể sử dụng các mô hình cục bộ trên mỗi máy khách. Do đó, cần phải đảm bảo mở rộng quy mô chính xác các chức năng của máy chủ để đảm bảo không xảy ra tắc nghẽn và sử dụng các giải pháp bảo mật thích hợp để ngăn chặn các cuộc tấn công vào máy chủ càng nhiều càng tốt. Một giải pháp bổ sung sẽ là điều chỉnh nền tảng theo hướng tiếp cận hoàn toàn phi tập trung, nơi các mô hình được chia sẻ bằng cách sử dụng Blockchain và mỗi khách hàng thực hiện tổng hợp cục bộ, loại bỏ sự cần thiết của người điều phối trong quá trình này.

Một bất lợi khác là cần có sự đồng bộ hóa giữa các máy khách khi gửi các mô hình của họ để tổng hợp. Máy khách bị lỗi hoặc chậm do không đồng bộ có thể khiến máy chủ không thực hiện đào tạo chính xác. Hiện tại, khuôn khổ giải quyết vấn đề này bằng cách đặt thời gian chờ để gửi các mô hình để nếu một trong các máy khách không phản hồi kịp thời, nó sẽ bị bỏ qua từ bước tổng hợp đó. Trong trường hợp này, một giải pháp dựa trên Blockchain cũng có lợi vì nó có thể được sử dụng như một kho lưu trữ không đồng bộ, nơi mỗi khách hàng có thể xuất bản các mô hình của mình mỗi khi đào tạo chúng cục bộ.

Mặc dù lợi ích của nó là giải quyết cả hai nhược điểm, nhưng điều cần thiết phải xem xét là việc sử dụng Blockchain cũng mang lại cho nó một loạt các mối đe dọa cần bao trùm, chẳng hạn như các cuộc tấn công đa số hoặc các cuộc tấn công xác thực khối.

8. Kết luận và công việc trong tương lai

Công trình này đề xuất một khuôn khổ bảo vệ quyền riêng tư để phát hiện phần mềm độc hại IoT, thúc đẩy FL để đào tạo và đánh giá cả các mô hình được giám sát và không được giám sát mà không chia sẻ dữ liệu nhạy cảm. Khung này được thiết kế để triển khai trên các nút mạng cung cấp quyền truy cập vào các thiết bị IoT trong mạng Wifi, 5G hoặc B5G, giảm tải tính toán từ chính thiết bị IoT. Theo nghĩa này, phía máy khách được thiết kế để triển khai trên RAN trong khi phía máy chủ được thiết kế để triển khai Fog / Cloud. Để chứng minh tính khả thi của nó trong một kịch bản IoT thực tế, tập dữ liệu N-BaIoT đã được sử dụng do tính không đồng nhất và dễ phân chia về các thiết bị IoT và các mẫu phần mềm độc hại. Sử dụng N-BaIoT, chúng tôi đã so sánh hiệu suất của: i) phương pháp liên kết, trong đó tất cả chủ sở hữu thiết bị đào tạo mô hình của riêng họ, được tổng hợp định kỳ trong một máy chủ, ii) thiết lập không bảo vệ quyền riêng tư, trong đó toàn bộ tập dữ liệu được tập trung và được đào tạo bởi máy chủ, và iii) thiết lập cục bộ trong đó mỗi chủ sở hữu thiết bị đào tạo một mô hình riêng lẻ và biệt lập. So sánh này đã chỉ ra rằng việc sử dụng dữ liệu đa dạng hơn và lớn hơn, như được thực hiện trong các phương pháp liên hợp và tập trung, có tác động tích cực đáng kể đến hiệu suất của mô hình cả trong kịch bản được giám sát và không được giám sát. Bên cạnh đó, nó đã được chứng minh rằng tính riêng tư của dữ liệu có thể được bảo toàn mà không làm mất hiệu suất của mô hình bằng cách tuân theo phương pháp liên kết. Khả năng phục hồi của các mô hình liên kết chống lại các máy khách độc hại đã được kiểm tra thông qua các cuộc tấn công đối nghịch sau: i) cuộc tấn công làm nhiễm độc dữ liệu lật tất cả các nhãn, ii) cuộc tấn công đầu độc mô hình nhân các gradient với một yếu tố tiêu cực và iii) một cuộc tấn công hủy bỏ mô hình. Kết quả cho thấy rằng nếu không sử dụng một kỹ thuật mạnh mẽ để tổng hợp các mô hình, một ứng dụng khách độc hại trong liên kết có thể làm hỏng mô hình. Một số chức năng tổng hợp mạnh mẽ, hoạt động như một biện pháp đối phó với các cuộc tấn công của đối thủ, đã được áp dụng để giải quyết vấn đề này, với tính năng tổng hợp trung bình cho thấy những cải tiến đầy hứa hẹn nhưng vẫn chưa đủ. Bước đầu tiên này theo hướng làm cho hệ thống mạnh mẽ chống lại các cuộc tấn công cho thấy rằng vẫn cần nhiều nỗ lực để đạt được kết quả thỏa mãn.

Trong tương lai sẽ có có kế hoạch đánh giá tác động của các cuộc tấn công đối thủ trong kịch bản không được giám sát để xác minh rằng chúng ảnh hưởng đến kết quả theo cách tương tự như đối với đối tác được giám sát. Hơn nữa, thử nghiệm độ bền của mô hình chống lại các cuộc tấn công trốn tránh, sử dụng các mẫu đối thủ giả mạo để tránh bị phát hiện tại thời điểm đánh giá, cũng có thể là một hướng đi thú vị trong tương lai. Ngoài ra, công việc này có kế hoạch nghiên cứu sâu hơn về các biện pháp đối phó hiện có chống lại các cuộc tấn công của đối thủ, chẳng hạn như Krum, Bulyan và AUROR.

Khả năng mở rộng trong các kịch bản B5G thực cũng là một vấn đề mà không thể nghiên cứu với bất kỳ bộ dữ liệu có sẵn nào, làm nảy sinh nhu cầu tạo ra một bộ dữ liệu lớn hơn và đa dạng hơn nhiều. Việc triển khai kiến trúc theo cách phân tán hoàn toàn bằng cách sử dụng Blockchain để trao đổi các mô hình liên kết cũng được xem xét. Bên cạnh đó, việc kết hợp Blockchain vào khuôn khổ có thể cải thiện các mối quan tâm về bảo mật và quyền riêng tư có thể có của khách hàng.