Tìm hiểu các thuật toán phát hiện xâm nhập

Nhập môn đám bảo và an ninh thông tin (IE105.N14.CNCL)

Huỳnh Tuấn Minh  
Sinh viên Trường Đại học Công nghệ thông tin, Đại học quốc gia thành phố Hồ Chí Minh Ts. Nguyễn Tấn Cầm  
Giảng viên Trường Đại học Công nghệ thông tin, Đại học quốc gia thành phố Hồ Chí Minh

Trương Quốc Đạt  
Sinh viên Trường Đại học Công nghệ thông tin, Đại học quốc gia thành phố Hồ Chí Minh

Phạm Nguyên Nam  
Sinh viên Trường Đại học Công nghệ thông tin, Đại học quốc gia thành phố Hồ Chí Minh

*Trong báo cáo này, chúng tôi sẽ giới thiệu các thuật toán phát hiện xâm nhập đã và đang được sử dụng trong các hệ thống phát hiện xâm nhập. Các thuật toán được chia làm hai phần: phần 1 là thuật toán phát hiện xâm nhập không có khả năng dự đoán thường được sử dụng trên các hệ thống Signature-based Intrusion Detection System cũng như các phần mềm tường lửa, dò tìm virus, Hosted-based Intrusion Detection System. Và phần 2 là các thuật toán phát hiện xâm nhập có khả năng dự đoán thường được sử dụng trên Network-based Intrusion Detection System và Anomaly-based Intrusion Detection System. Và cuối cùng là tiến hành tổng quan 24 thuật toán và đưa ra 2 dạng demo web application và mô phỏng quá trình giao tiếp giữa client và server.*

# Giới thiệu

Hiện nay, với sự phát triển của công nghệ thông tin nói chung và kết nối vạn vật (IoT) nói chung. Mạng Internet dần trở nên phổ biến và trở thành một phần không thể thiếu của chúng ta. Tuy nhiên, bên cạnh những lợi ích mà mạng Internet mang lại chúng còn mang theo những mối nguy hiểm tiềm tàng, những cuộc xâm nhập và lây lan mã độc. Để phát hiện và ngăn chặn sự lây lan không mong muốn của các mã độc, các thuật toán phát hiện xâm nhập là vô cùng cần thiết và không thể thiếu trong các hệ thống phát hiện xâm nhập. Ở báo cáo này, chúng ta sẽ đi tìm hiểu về các thuật toán phát hiện xâm nhập đó.

1. Tổng hợp các từ viết tắt

|  |  |
| --- | --- |
| **Từ viết tắt** | **Định nghĩa** |
| IDS | Intrusion detection system |
| DT | Decision tree |
| ET | Extra tree |
| RF | Random forest |
| ML | Machine Learning |
| DL | Deep Learning |
| DP | Data preprocessing |
| DE | feature engineering |
| N-IDS | Network-based Intrusion Detection System |
| A-IDS | Anomaly-based Intrusion Detection System |

# Data relevant

## Tổng quan

Ở đây, chúng tôi sử dụng 4 bộ dataset bao gồm KDDCup99, NSL-KDD, CICIDS2017 và UNSW-NB15 để chạy 24 thuật toán khác nhau, để tăng thêm phần khách quan chúng tôi thực hiện một số phương pháp Data preprocessing cơ bản như padding value, min-max normalization, standard normailization và chuẩn hóa dữ liệu từ dạng category về dạng numeric và không thực hiện bất kỳ phương pháp Feature engineering và Feature selection nào.

* **KDD:** 4,898,431 x 42 (row x column)
* **NSL-KDD:**
* Trainset: 125,973 x 43 (row x column)
* Testset: 22,544 x 43 (row x column)
* **CICIDS2017:** 2,830,743 x 79 (row x column)
* **UNSW-NB15:**
* Trainset: 82,332 x 45 (row x column)
* Testset: 175,341 x 45 (row x column)

## Phân bố data tổng quan và split data trainset và data testset

Để đảm bảo tính khách quan của tác giả, chúng tôi slit trainset và testset theo theo tỉ lệ 8:2 và đảm bảo sự phân bố của các class tương đồng như sự phân bố của các class trong bộ dataset gốc.

# Các thuật toán phát hiện xâm nhập không có khả năng dự đoán

Đây là các IDS hoạt động dựa trên chữ ký, giám sát các gói tin trên mạng tương tự như cách phần mềm diệt virus hoạt động. Tuy nhiên Signature-Based có thể không phát hiện được những mối đe dọa mới, khi chữ ký để nhận biết nó chưa được IDS cập nhật. Snort chủ yếu được sử dụng IDS dựa trên chữ ký vì nó là phần mềm mã nguồn mở và nhẹ.

1) SNORT

a) Khái niệm:

- SNORT là một hệ thống phát hiện xâm nhập dựa trên mạng được viết bằng ngôn ngữ lập trình C. Nó là phần mềm mã nguồn mở miễn phí. Nó cũng có thể được sử dụng như một trình thám thính gói tin để giám sát hệ thống trong thời gian thực. Quản trị viên mạng có thể sử dụng nó để xem tất cả các gói đến và tìm những gói gây nguy hiểm cho hệ thống. Nó dựa trên công cụ chụp gói thư viện. Các quy tắc khá dễ tạo và thực hiện và nó có thể được triển khai trong bất kỳ loại hệ điều hành và bất kỳ loại môi trường mạng nào.

b) Tính năng:

* Giám sát giao thông thời gian thực
* ghi nhật ký gói
* Phân tích giao thức
* phù hợp với nội dung
* vân tay hệ điều hành
* Có thể được cài đặt trong bất kỳ môi trường mạng nào.
* Tạo nhật ký
* Mã nguồn mở
* Các quy tắc rất dễ thực hiện

c) Điểm mạnh và điểm yếu

* Điểm mạnh:
* Signature được mô hình hóa trên hoạt động xâm nhập đã biết. Vì vậy, người dùng có thể kiểm tra cơ sở dữ liệu chữ ký và nhanh chóng xác định hoạt động xâm nhập nào mà hệ thống phát hiện lạm dụng được lập trình để cảnh báo.
* Hệ thống phát hiện lạm dụng bắt đầu bảo vệ mạng của bạn ngay sau khi cài đặt.
* Có tỷ lệ dương tính giả thấp miễn là các cuộc tấn công được xác định rõ ràng trước.
* Khi báo động kích hoạt, người dùng có thể liên hệ trực tiếp điều này với một loại hoạt động cụ thể xảy ra trên mạng.
* Điểm yếu:
* Việc xử lý toàn bộ lưu lượng rất tốn thời gian và sẽ làm chậm thông lượng của hệ thống
* Phải cập nhật dữ liệu thường xuyên
* Việc lạm dụng IDS sẽ gửi rất nhiều cảnh báo cho các cuộc tấn công không thành công. Từ đó có thể khó kiểm soát
* Kiến thức về các cuộc tấn công phụ thuộc rất nhiều vào hệ điều hành, phiên bản và ứng dụng do đó gắn liền với các môi trường cụ thể

Thuật toán được triển khai trong SNORT là Aho-Corasick

2) Các thuật toán được sử dụng trong snort

a) Aho-Corasick

Nó là một loại thuật toán so khớp từ điển định vị các phần tử của một tập hợp chuỗi hữu hạn ("từ điển") trong một văn bản đầu vào. Nó khớp với tất cả các chuỗi cùng một lúc. Độ phức tạp của thuật toán là tuyến tính về độ dài của chuỗi cộng với độ dài của văn bản được tìm kiếm cộng với số lượng kết quả khớp. Lưu ý rằng vì tất cả các kết quả phù hợp đều được tìm thấy nên có thể có số lượng kết quả khớp bậc hai nếu mọi chuỗi con khớp (ví dụ: từ điển = a, aa, aaa, aaaa và chuỗi đầu vào là aaaa).

# các thuật toán phát hiện xâm nhập có khả năng dự đoán

Nhìn chung, các thuật toán phát hiện xâm nhập hiện nay thường được sử dụng trên N-IDS và A-IDS là hầu hết các thuật toán Machine Learning cũng như Deep Learning bao gồm các thuật thoán phân lớp (Classification) và phân nhóm (Clustering), học có giám sát (Supervised Learning) và học không giám sát (Unsupervised Learning).

## Ảnh có chứa bàn Mô tả được tạo tự động

1. Sơ lược về các thuật toán phát hiện xâm nhập

Ở một số mô hình phức tạp, người ta còn kết hợp sử dụng các thuật toán Machine Learning/ Deep Learning phục vụ cho các bước Data processing và Feature engieering.

## Các mô hình phát hiện xâm nhập cơ bản

## Ở trong các thuật toán phát hiện xâm nhập cơ bản thường sử dụng từng thuật toán ML/DL cho các bài toán Phân lớp (Classification), Hồi quy (Regression), Gom cụm (Clustering), học có giám sát (Supervised Learning), học bán giám sát (Semi-supervised Learning) và học không giám sát (Unsupervised Learning).. Dưới đây chúng tôi xin được liệt kê những thuật toán tiêu biểu nhất

### Machine Learning:

* **Decision tree:** DT là một trong những thuật toán ML được giám sát cơ bản được sử dụng để phân loại và hồi quy tập dữ liệu đã cho bằng cách áp dụng chuỗi các quyết định. Mô hình có cấu trúc giống như 1 cây bình thường với các nút, nhánh và lá. Mỗi nút biểu thị một thuộc tính hoặc một đặc điểm. Nhánh đại diện cho một quyết định hoặc một quy tắc trong khi mỗi lá đại diện cho một kết quả có thể xảy ra hoặc nhãn lớp. Thuật toán DT tự động chọn các tính năng tốt nhất để xây dựng cây và sau đó thực hiện thao tác cắt bớt để loại bỏ các nhánh không phù hợp khỏi cây để tránh quá khớp.
* **Random Forest:** RF là một bộ phân loại tập hợp được sử dụng để cải thiện độ chính xác. RF bao gồm nhiều Decision Tree. RF có sai số phân loại thấp so với các thuật toán phân loại truyền thống khác. Số lượng cây, kích thước nút tối thiểu và số lượng tính năng được sử dụng để tách từng nút.
* Ưu điểm: Các Forest đã tạo ra có thể được lưu lại để tham khảo trong tương lai. RF khắc phục tình trạng over fitting.
* Trong RF, độ chính xác và tầm quan trọng sự thay đổi được tạo ra một cách tự động.
* Khi xây dựng các cây riêng lẻ trong RF, để chọn nút tốt nhất để tách. Đây giá trị bằng √A, trong đó A là node của các thuộc tính trong tập dữ liệu. Tuy nhiên, RF sẽ tạo ra nhiều cây ồn ào, ảnh hưởng đến độ chính xác và quyết định sai đối với mẫu mới.
* **K-Nearest Neighbor:** là một trong những thuật toán ML được giám sát đơn giản nhất sử dụng ý tưởng về "tính năng tương tự" để dự đoán lớp của một mẫu dữ liệu nhất định. Nó xác định một mẫu dựa trên các hàng xóm của nó bằng cách tính toán khoảng cách của nó với các hàng xóm. Trong thuật toán KNN, tham số k ảnh hưởng đến hiệu suất của mô hình. Nếu giá trị của k rất nhỏ, mô hình có thể dễ bị lắp quá mức. Trong khi, việc lựa chọn giá trị k rất lớn có thể dẫn đến phân loại sai đối tượng mẫu.
* **Support vector machine:** là một thuật toán ML được giám sát dựa trên ý tưởng về maximum margin separating hyperplane trong không gian đặc trưng n chiều. Nó được sử dụng cho các giải pháp của cả hai vấn đề tuyến tính và phi tuyến tính. Đối với các bài toán phi tuyến, các hàm nhân được sử dụng. Ý tưởng đầu tiên là ánh xạ một vectơ đầu vào có chiều thấp vào không gian đặc trưng có chiều cao bằng cách sử dụng hàm nhân. Tiếp theo, một siêu mặt phẳng biên cực đại tối ưu thu được, hoạt động như một đường biên quyết định bằng cách sử dụng các vectơ hỗ trợ. Đối với NIDS, thuật toán SVM có thể được sử dụng để nâng cao hiệu quả và độ chính xác của nó bằng cách dự đoán chính xác các lớp bình thường và độc hại.
* **K-mean clustering:** Clustering là một ý tưởng phân chia dữ liệu thành các cụm (hoặc nhóm) có ý nghĩa, bằng cách đặt các dữ liệu có độ tương đồng cao vào cùng một cụm. K-Mean clustering là một trong những thuật toán ML lặp lại dựa trên trung tâm phổ biến, học theo cách không giám sát. K đại diện cho số lượng tâm (trung tâm của cụm) trong một tập dữ liệu. Để gán các điểm dữ liệu nhất định cho một cụm, thông thường khoảng cách được tính toán. Ý tưởng chính là giảm tổng khoảng cách giữa các điểm dữ liệu và các trung tâm tương ứng của chúng trong một cluster.
* **Artificial neural network:** cũng là một thuật toán ML được giám sát và được lấy cảm hứng từ hoạt động của hệ thống thần kinh của não người. Nó được tạo thành từ các phần tử xử lý được gọi là nơ-ron (nút) và kết nối giữa chúng. Các nút này được tổ chức trong một lớp đầu vào, nhiều lớp ẩn và một lớp đầu ra. Thuật toán lan truyền ngược được sử dụng như một kỹ thuật học tập cho ANN. Ưu điểm chính của việc sử dụng kỹ thuật ANN là khả năng thực hiện mô hình phi tuyến bằng cách học từ các bộ dữ liệu lớn hơn. đạt được một giải pháp tối ưu.
* **Ensemble methods:** Ý tưởng đằng sau các phương pháp tổng hợp là để có được lợi ích từ bộ phân loại khác nhau bằng cách học theo cách tập hợp. Vì mỗi trình phân loại có một số điểm mạnh và điểm yếu. Một số có thể hoạt động tốt trong việc phát hiện một kiểu tấn công cụ thể và cho thấy hiệu suất kém đối với các kiểu tấn công khác. Cách tiếp cận tổng hợp là kết hợp các bộ phân loại yếu bằng cách đào tạo nhiều bộ phân loại và sau đó tạo thành một bộ phân loại mạnh hơn bằng cách chọn sử dụng thuật toán bỏ phiếu.

### Deep Learning:

* **CFA**: Bộ dữ liệu KDD đã được sử dụng. Phát hiện các loại xâm nhập mạng khác nhau một cách hiệu quả.
* **Recurrent neural networks:** Recurrent Neural Networks (RNN) mở rộng khả năng của mạng nơ-ron truyền thống và được thiết kế để mô hình hóa dữ liệu trình tự. RNN được tạo thành từ các đơn vị đầu vào, ẩn và đầu ra, trong đó các đơn vị ẩn được coi là các phần tử bộ nhớ. Để đưa ra quyết định, mỗi đơn vị RNN dựa vào đầu vào hiện tại của nó và đầu ra của đầu vào trước đó. RNN được sử dụng rộng rãi trong các lĩnh vực khác nhau như xử lý giọng nói, nhận dạng hoạt động của con người, dự đoán chữ viết tay và hiểu ngữ nghĩa, ... Đối với IDS, RNN có thể được sử dụng để phân loại được giám sát và trích xuất tính năng. RNN thông thường có thể xử lý các chuỗi có độ dài hạn chế và sẽ bị ảnh hưởng bởi short-term memory nếu độ dài của chuỗi dài. Các biến thể RNN khác nhau như Long short-term memory (LSTM) và gated recurrent unit (GRU) được đề xuất để giải quyết những vấn đề này.
* **AutoEncoder:** AutoEncoder (AE) là một kỹ thuật DL phổ biến thuộc họ mạng nơ-ron không có giám sát. Nó hoạt động dựa trên ý tưởng kết hợp đầu ra càng gần với đầu vào càng tốt bằng cách tìm hiểu các tính năng tốt nhất. Nó chứa các lớp đầu vào và đầu ra có cùng kích thước, trong khi kích thước của các lớp ẩn thường nhỏ hơn lớp đầu vào. AE là đối xứng và hoạt động theo kiểu Encoder-Decoder. Các biến thể khác nhau của AE là AE xếp chồng, AE thưa thớt và AE đa dạng
* **Deep neural network:** DNN là một cấu trúc DL cơ bản cho phép mô hình học ở nhiều lớp. Nó bao gồm một lớp đầu vào, một lớp đầu ra và nhiều lớp ẩn. DNN được sử dụng để mô hình hóa các hàm phi tuyến phức tạp. Số lượng lớp ẩn tăng lên sẽ nâng cao mức độ trừu tượng của mô hình để tăng khả năng của nó.
* **Deep belief network:** DBN là một mô hình DL được xây dựng bằng cách xếp chồng nhiều máy Restricted Boltzmann thành các lớp theo sau là lớp phân loại softmax. RBM là mô hình hai lớp (lớp đầu vào và lớp ẩn) với luồng dữ liệu theo cả hai hướng. Trong DBN, mỗi nút trong một lớp được kết nối với mỗi nút khác trong lớp trước và lớp sau, nhưng trong một lớp, các nút không được kết nối. DBN được đào tạo trước bằng cách sử dụng phương pháp học theo lớp tham lam theo cách không được giám sát, theo sau là phương pháp tinh chỉnh có giám sát để học các tính năng hữu ích.
* **Convolutional neural network:** Convolutional neural network (CNN) là một cấu trúc DL phù hợp hơn với dữ liệu được lưu trữ trong mảng. Nó bao gồm một lớp đầu vào, chồng các lớp tích tụ và tổng hợp để trích xuất đặc trưng, và cuối cùng là một lớp được kết nối đầy đủ và một bộ phân loại softmax trong lớp phân loại. CNN rất thành công trong lĩnh vực thị giác máy tính. Đối với IDS, chúng được sử dụng cho mục đích phân loại và trích xuất tính năng được giám sát.

### Review 24 algorithms

Chart, line chart

Description automatically generated

1. Biểu đồ sơ lược 24 thuật toán trên độ đo Acurracy

Chart, line chart

Description automatically generated

1. Biểu đồ sơ lược 24 thuật toán trên độ đo F1 score

Chart

Description automatically generated

1. Biểu đồ sơ lược 24 thuật toán trên độ đo Time taken thời gian thực thi

Trong các hệ thống hoạt động real-time, các độ đo như Accuracy, Precision, Reccall, F1-score,.. cao không là chưa đủ. Bên cạnh đó, các thông số như thời gian thực thi và tài nguyên sử dụng thấp cũng có những vai trò quan trọng cũng như quyết định hệ thống đó có doạt động mạnh mẽ hay không. Do đó, sau khi thực hiện chạy thử 24 thuật toán trên 4 bộ dataset khác nhau, chúng tôi nhận thấy 3 thuật toán Decision Tree, Extra Tree, Random Forest chạy ổn định, các thông số độ đo Acurracy và F1 score tương đối cao, đồng thời thời gian thực thi tương đối thấp. Do đó chúng tôi lấy 3 thuật toán này để phát triển mô hình của riêng mình.

Table

Description automatically generated

1. Bảng thông số các thuật toán trước và sau khi thực hiện các kĩ thuật DP và FE

Ở đây, chúng tôi thực hiện các kỹ thuật Data preprocessing như Min-Max Normalization, Padding values, Label encode, Oversampling (SMOTE).

Và Feature engineering như Feature important (average score) and Oversampling (SMOTE), Gini index.

Có thể thấy ở đây, trước khi thực hiện các ký thuật nêu trên, Decision Tree đứng đầu và sau đó là Extra Tree và cuối cùng là Random Forest. Tuy nhiên sau khi thực hiện các kỹ thuật đã nêu trên thì Random Forest đứng nhất, tiếp theo là Decision Tree và cuối cùng là Extra Tree.

Tuy Extra Tree có các độ đo thấp kể cả trước và sau khi thực hiện các kĩ thuật nêu trên, tuy nhiên Extra Tree lại có thời gian thực thi nhanh nhất.

Do đó, chúng ta có thể thấy mỗi một mô hình đều có ưu nhược điểm khác nhau, để cân bằng hơn và tạo ra một mẫu lớn hơn chúng tôi thực hiện combine các model và đưa ra model cuối cùng.

Có thể thấy ở đây, mô hình được boosting bằng XGBoost xảy ra hiện tượng Overfit do thực hiện các kĩ thuật Data preprocessing và Feature enginering.

Do đó, cần lựa chọn các kĩ thuật và các thuật toán kết hợp với nhau cho thật hợp lí.

Shape

Description automatically generated

1. Biểu đồ 4 mô hình trước khi thực hiện các kỹ thuật DP và FE

Chart, line chart

Description automatically generated

1. Biểu đồ 4 mô hình sau khi thực hiện các kỹ thuật DP và FE

## Các mô hình xâm nhập phức tạp

### Practical Swarm Optimization (PSO)

1. Tổng hợp các mô hình kết hợp PSO

| Practical Swarm Optimization (PSO | | |
| --- | --- | --- |
| Algorithms | Results | Dataset |
| PSO+ k-means | DR= 74.43% FAR: 1.86% | KDD |
| PSO+SVM | Accuracy: 81.8% | NSL-KDD |
| NN-PSO | DR = 99.4%  FAR = 0.6% | KDD |
| PSO+ MCLP | Accuracy = 0.9913  False Alarm Rate = 0.01947 | KDD |
| FLN-PSO | Accuracy with 25 neurons= 0.98  Accuracy with 200 neurons= 0.99 | KDD |
| PSO + neural network | Accuracy = 98.08 using NSL-KDD.  Accuracy = 99.66 using KDD. | KDD NSL-KDD |

* **PSO+ k-means**: Các kết quả thu được đã minh họa kết quả tốt về DR và XA. Ngoài ra, phương pháp đã đạt được thời gian xử lý nhanh hơn.
* **PSO+SVM**: PSO được sử dụng cho cả việc lựa chọn đặc trưng và lựa chọn các tham số SVM, Về thời gian tính toán, phương pháp này khá tốn kém.

Ở đây, PSO đầu tiên thực hiện tối ưu hóa tham số sử dụng SVM để nhận giá trị tối ưu của C (chi phí) và g (tham số gamma). Sau đó PSO thực hiện tối ưu hóa tính năng để có được tính năng tối ưu hóa. sau đó các tham số và tính năng này được trao cho SVM để được độ chính xác cao hơn. Thí nghiệm được thực hiện bằng cách sử dụng bộ dữ liệu NSL-KDD.

* **NN-PSO**: Bộ dữ liệu mô phỏng đã được sử dụng. Trong phương pháp đề xuất, PSO đã được áp dụng để lựa chọn các tính năng.
* **PSO+MCLP**: Cách tiếp cận mang lại kết quả tốt về DR, FAR và thời gian chạy.

Tối ưu hóa bầy đàn hạt (PSO) là một kỹ thuật tối ưu hóa mạnh mẽ và đơn giản để thực hiện đã được sử dụng để cải thiện hiệu suất của bộ phân loại MCLP. Bộ dữ liệu KDD CUP 99 được sử dụng để đánh giá hiệu suất của phương pháp được đề xuất. Kết quả cho thấy mô hình đề xuất có hiệu suất tương đương dựa trên tỷ lệ phát hiện, tỷ lệ cảnh báo sai và thời gian chạy so với hai phân loại điểm chuẩn khác.

* **FLN-PSO**: PSO đã được áp dụng để tối ưu hóa trọng số FLN và điều này giúp cải thiện độ chính xác của hiệu suất phân loại. Kết quả cho thấy độ chính xác của phương pháp là tăng khi số lượng tế bào thần kinh tăng.
* **PSO + neural network**: Bộ dữ liệu KDD và NSL-KDD đã được áp dụng để đánh giá hiệu suất của phương pháp. Các kết quả minh họa rằng phương pháp này cung cấp độ chính xác và hiệu suất cao để phát hiện các loại tấn công khác nhau.Artificial Bees Colony (ABC)

Sự kết hợp giữa PSO và thuật toán neural network được sử dụng để nhận dạng các cuộc xâm nhập, từ đó có thể phân loại các cuộc tấn công một cách hiệu quả, đồng thời giảm số lượng cảnh báo sai và cải thiện tỷ lệ phát hiện. Kết quả thu được cho thấy phương pháp đề xuất có độ chính xác và hiệu suất cao hơn so với các thuật toán khác để phát hiện các loại tấn công khác nhau.

### Artificial Bees Colony (ABC)

1. Tổng hợp các mô hình kết hợp ABC

| Artificial Bees Colony (ABC) | | |
| --- | --- | --- |
| Algorithms | Results | Dataset |
| ABC | DR= 74.43% FAR: 1.86% | KDD |
| ABC | Accuracy: 81.8% | KDD |
| ANN-ABC | DR = 99.4%  FAR = 0.6% | NSL-KDD |

* **ABC**: Bộ dữ liệu KDD cup 99 được sử dụng để đánh giá hiệu suất của phương pháp. Kết quả cho thấy khả năng của phương pháp được đề xuất để phát hiện các loại tấn công khác nhau.

Một cách tiếp cận mới lấy cảm hứng từ hành vi tự vệ của loài ong trong tự nhiên được đề xuất để cải thiện Hệ thống Phát hiện Xâm nhập (IDS). Ở các đàn ong mật, những người bảo vệ phân biệt ong cùng tổ với những con không cùng tổ ở lối vào tổ ong bằng cách sử dụng phương pháp chứa các phương pháp Không mong muốn-Vắng mặt (UA) hoặc Có mặt mong muốn (DP) và Quyết định lọc (FD). Các phương pháp này được sử dụng để phát hiện kẻ xâm nhập và phân loại loại của nó. Theo cách tiếp cận được đề xuất, bộ phát hiện UA chịu trách nhiệm phát hiện các cuộc tấn công được xác định trước dựa trên dấu hiệu tấn công của chúng. Mạng thần kinh được đào tạo bởi Bees Algorithm (BA) đã được sử dụng để tìm hiểu các kiểu tấn công được đưa ra trong tập dữ liệu huấn luyện và sử dụng các mẫu này để tìm các cuộc tấn công cụ thể trong tập dữ liệu thử nghiệm. Trình phát hiện DP chịu trách nhiệm phát hiện các hành vi bất thường dựa trên mô hình hành vi bình thường được đào tạo. Cuối cùng, phương pháp FD được sử dụng để huấn luyện bộ phát hiện UA trong thời gian thực để phát hiện các xâm nhập mới.

* **ABC**: Bộ dữ liệu KDD đã được sử dụng, các kết quả minh họa rằng độ chính xác của phương pháp tiếp cận không thay đổi khi số lượng thế hệ tối đa tăng lên.

Trong thuật toán ABC, mỗi nguồn thực phẩm đại diện cho một giải pháp khả thi (nghĩa là nó có thể được coi là một hạt trong PSO) đối với vấn đề được xem xét và kích thước của nguồn thực phẩm thể hiện chất lượng của giải pháp. Thuật toán này đại diện cho một đàn ong nhân tạo (ở đây được gọi đơn giản là ong) bao gồm ba loại được tuyển dụng, người quan sát và người do thám. Nửa đầu của đàn ong bao gồm những con ong được tuyển dụng, trong khi nửa sau bao gồm những con ong quan sát. Thuật toán ABC giả định chỉ có một con ong được tuyển dụng cho mỗi nguồn thức ăn – cụ thể là số lượng nguồn thức ăn bằng với số lượng ong được tuyển dụng. Khi một nguồn thức ăn bị bỏ rơi, một con ong được tuyển dụng đang làm việc trên nguồn thức ăn đó sẽ trở thành người do thám cho đến khi tìm thấy nguồn thức ăn mới, lúc đó nó lại được tuyển dụng trở lại.

* **ANN-ABC**: Để đánh giá hiệu suất của phương pháp, bộ dữ liệu NSL-KDD đã được sử dụng. ABC đã được áp dụng để cải thiện trọng số của MLP. Tất cả các tính năng trong bộ dữ liệu đã được sử dụng và kết quả cho thấy độ chính xác hơi thấp.

Thuật toán tổ ong lai nhân tạo (ABC) và Perceptron đa lớp (MLP) đã được đề xuất để xây dựng một IDS mạng hiệu quả. MLP được sử dụng như một bộ phân loại để phân biệt các gói bình thường và bất thường trong lưu lượng mạng. Cấu trúc của MLP đã được tạo ra dựa trên các tính năng của bộ dữ liệu (NSL-KDD 99). Ngoài ra, thuật toán ABC được sử dụng để đào tạo MLP bằng cách tối ưu hóa các giá trị của trọng số và độ lệch liên kết.

### Cuttlefish Algorithm (CFA)

1. Tổng hợp các mô hình kết hợp CFA

| Cuttlefish Algorithm | | |
| --- | --- | --- |
| Algorithms | Results | Dataset |
| CFA | Using 5 features Detection Rate = 91%, FPR = 3.917  With 41 features Detection Rate = 71.087, FPR = 17.685 | KDD |
| CFA+ D | PSP (accuracy) = 92.17% | KDD |
| FGLCC + CFA | Accuracy= 95.03%  False Positive Rate= 1.65% | KDD |

* **CFA**: Bộ dữ liệu KDD-99 đã được sử dụng. Nghiên cứu kết luận rằng kết quả được cải thiện bằng cách giảm số lượng tính năng và nó mang lại DR cao hơn với FAR thấp hơn.

Bài báo này trình bày một cách tiếp cận lựa chọn tính năng mới dựa trên thuật toán tối ưu hóa mực nang được sử dụng cho các hệ thống phát hiện xâm nhập (IDS). Bởi vì IDS xử lý một lượng lớn dữ liệu, một trong những nhiệm vụ quan trọng của IDS là giữ chất lượng tốt nhất của các tính năng đại diện cho toàn bộ dữ liệu và loại bỏ các tính năng dư thừa và không liên quan. Mô hình được đề xuất sử dụng thuật toán mực nang (CFA) làm chiến lược tìm kiếm để xác định tập hợp con tối ưu của các tính năng và bộ phân loại cây quyết định (DT) để đánh giá các tính năng được chọn do CFA tạo ra. Bộ dữ liệu KDD Cup 99 được sử dụng để đánh giá mô hình đề xuất. Kết quả cho thấy tập hợp con tính năng thu được bằng cách sử dụng CFA mang lại tỷ lệ phát hiện và tỷ lệ chính xác cao hơn với tỷ lệ cảnh báo sai thấp hơn, khi so sánh với kết quả thu được khi sử dụng tất cả các tính năng.

* **CFA+D**: Bộ dữ liệu KDD cup99 đã được sử dụng. Cách tiếp cận này kết hợp (CFA, DT, RFGA) để thu được kết quả tốt cho ID, hệ thống đề xuất hoạt động tốt hơn khi chọn 5 tính năng thay vì sử dụng tất cả. U2R hoạt động không tốt, trong mọi trường hợp, kết quả tốt hơn đã đạt được khi sử dụng 5 tính năng ngoại trừ U2R.

Hệ thống được đề xuất sử dụng một tác nhân có tên là Tác nhân tạo quy tắc và tính năng (RFGA) để tạo một tập hợp con các tính năng với các quy tắc tương ứng. Tác nhân RFGA sử dụng CFA để tìm kiếm tập hợp con các tính năng tối ưu, trong khi DT được sử dụng làm phép đo trên các tính năng đã chọn. Mô hình đề xuất được thử nghiệm trên bộ dữ liệu KDD Cup 99. Các kết quả thu được cho thấy rằng hệ thống được đề xuất cho hiệu suất tốt hơn ngay cả với một tập hợp con nhỏ gồm 5 tính năng khi so sánh với việc sử dụng tất cả 41 tính năng.

* **FGLCC + CFA**: Bộ dữ liệu KDD-CUP99 đã được sử dụng. Các kết quả thu được cung cấp độ chính xác hiệu suất cao.

Các phương thức Filter và wrapper được đặt tên tương ứng là nhóm đối tượng dựa trên thuật toán hệ số tương quan tuyến tính (FGLCC) và thuật toán mực nang (CFA). Decision tree được sử dụng làm bộ phân loại trong phương pháp đề xuất. Để xác minh hiệu suất, phương pháp được đề xuất đã được áp dụng trên bộ dữ liệu lớn KDD Cup 99. Kết quả đã xác minh độ chính xác cao (95,03%) và tỷ lệ phát hiện (95,23%) với tỷ lệ dương tính giả thấp (1,65%) so với các phương pháp hiện có trong tài liệu

# 4) Genetic Algorithm (GA)

1. Tổng hợp các mô hình kết hợp GA

| Genetic Algorithm (GA) | | |
| --- | --- | --- |
| Algorithms | Results | Dataset |
| GA | DR= 74.43% FAR: 1.86% | KDD |
| PSO-GA | Accuracy: 81.8% | KDD |
| GA+ FA | DR = 99.4%  FAR = 0.6% | NSL-KDD |
| GA + Fuzzy Logic | Accuracy = 0.9913 | NSL-KDD |
| GA+ Fuzzy | Accuracy = 0.9913 | NSL-KDD |

1. Sample of a Table footnote. (*Table footnote*)

* **PSO + GA**: Bộ dữ liệu KDD Cup đã được sử dụng để đánh giá phương pháp này.Nguồn tham số được chọn bằng PSO vàquy tắc phân loại đã được tạo, GA được sử dụng để phân loại bộ dữ liệu thành tấn công và bình thường.

Tối ưu hóa bầy đàn hạt (PSO) là phương pháp ngày càng được quan tâm với những kết quả đáng chú ý cho việc thiết kế IDS trong thời gian gần đây. Thật không may, việc xác định các giá trị tham số trở thành một vấn đề tối ưu hóa trong khả năng thực hiện của PSO. IDS luôn phải xử lý lượng dữ liệu khổng lồ khiến quá trình đào tạo và thử nghiệm chậm và tỷ lệ phát hiện thấp. Vì vậy, lựa chọn tính năng là một trong những chủ đề chính trong IDS. Do đó, một công nghệ mới có tên là “Hiệu suất tìm kiếm tối ưu toàn cầu của tối ưu hóa bầy đàn hạt (PSO)” được sử dụng để chọn tham số từ bộ dữ liệu KDD Cup 99 và mô hình tham số kết quả được cung cấp cho GA. Nó tạo ra các quy tắc phân loại chính xác cho IDS. Phương pháp này rất dễ thực hiện và có ít thông số để điều chỉnh.

* **GA + FA**: Bộ dữ liệu NSL đã được sử dụng. Kết quả cho thấy khả năng của phương pháp xử lý các tập dữ liệu quy mô lớn.

Các thuật toán di truyền (GA) được sử dụng rộng rãi làm thuật toán tìm kiếm để tạo ra các bộ phát hiện bất thường. Đó là một kỹ thuật trí tuệ nhân tạo lấy cảm hứng từ quá trình tiến hóa sinh học, chọn lọc tự nhiên và tái tổ hợp gen để tạo ra các giải pháp hữu ích cho việc tối ưu hóa vấn đề [6]. GA sử dụng dữ liệu dưới dạng nhiễm sắc thể tiến hóa thông qua các bước sau: chọn lọc (thường là chọn lọc ngẫu nhiên), trao đổi chéo (tái tổ hợp để tạo ra nhiễm sắc thể mới) và toán tử đột biến. Cuối cùng, một chức năng phù hợp được áp dụng để chọn những cá nhân tốt nhất (rất phù hợp). Quá trình này được lặp lại trong một số thế hệ cho đến khi đạt được cá thể (hoặc nhóm cá thể) đáp ứng chặt chẽ điều kiện mong muốn. GA vẫn đang được sử dụng cho đến thời điểm hiện tại để tạo trình phát hiện bất thường bằng cách sử dụng chức năng phù hợp dựa trên số lượng phần tử trong tập huấn luyện được bao phủ bởi trình phát hiện và cả khối lượng trình phát hiện .

* **GA + Fuzzy Logic**: Cung cấp kết quả đạt yêu cầu và chất lượng cao về bất thường DR và FPA.

Thuật toán di truyền được sử dụng để tạo Chữ ký số của Phân đoạn mạng bằng cách sử dụng Phân tích luồng, trong đó thông tin được trích xuất từ dữ liệu luồng mạng được sử dụng để dự đoán hành vi lưu lượng mạng trong một khoảng thời gian nhất định. Hơn nữa, lược đồ Logic mờ được áp dụng để quyết định xem một thể hiện có đại diện cho sự bất thường hay không, khác với một số cách tiếp cận có trong tài liệu. Thật vậy, người ta đã đề xuất một hệ thống chuyên gia có khả năng giám sát lưu lượng mạng với các luồng IP trong khi các hành vi dự kiến được tạo ra trong khoảng thời gian đều đặn, đưa ra cảnh báo khi có sự cố có thể xảy ra. Hệ thống phát hiện bất thường được đề xuất tự động phơi bày các sự cố mạng

* **GA+ Fuzzy**: Bộ dữ liệu NSL-KDD đã được sử dụng. Phương pháp đề xuất phát hiện các cuộc tấn công một cách hiệu quả. Thêm vào, giảm FAR.

Hệ thống phát hiện xâm nhập dựa trên kết hợp (GA-Fuzzy) để xử lý Bộ dữ liệu NSL-KDD dung lượng lớn nhằm phát hiện các cuộc tấn công một cách hiệu quả và để giảm tỷ lệ cảnh báo phân loại sai. Ở đây, thuật toán di truyền (GA) được sử dụng để tạo mẫu mới (tính năng mới, bản ghi) để huấn luyện bộ phân loại mờ một cách hiệu quả. Chúng tôi sử dụng Phân tích thành phần nguyên tắc (PCA) làm phương pháp lựa chọn tính năng giúp loại bỏ dữ liệu không liên quan và dư thừa khỏi bộ dữ liệu NSL-KDD nhằm cải thiện hiệu quả và đạt được độ chính xác phát hiện 99,96% và tỷ lệ cảnh báo sai 0,04%.

# Demo

Ở đây, chúng tôi thực hiện 2 dạng demo bao gồm như sau:

1. **Dạng 1**: Mở socket và thực hiện quá trình mô phỏng bắt tay 3 bước. Bên client sẽ gửi data cho server và bên server sẽ thực hiện detect data đó và trả về kiểu xâm nhập độc hại. Đồng thời, lịch sử detect còn được lưu lại với tập tin log.txt trong server.

Graphical user interface, text

Description automatically generated

1. Mô phỏng cilent và server

Ở hình trên, chúng tôi thực hiện mô phỏng lại client và server bằng 2 folder client và server và mở socket. Ban đầu, ở client có file data.txt chính là dữ liệu cần detect ở đây. Ở server thì không có file data.txt và file log.txt lưu lại lịch sử detect.

A screenshot of a computer

Description automatically generated with medium confidence

1. Mô phỏng quá trình chuyển data và lưu lại lịch sử phát hiện

Ở đây, sau khi thực hiện mô phỏng quá trình bắt tay 3 bước bằng việc mở socket thì bên server nhận được dữ liệu từ client gửi qua dưới dạng file data.txt và thực hiện detect dữ liệu đó. Đồng thời, lưu dữ liệu đã detect dưới dạng file log.txt.

Graphical user interface

Description automatically generated

1. Các kết quả sau khi thực hiện phát hiện xâm nhập

Thông tin detect bao gồm thời gian thực hiện detect và kiểu xâm nhập độc hại.

Ngoài ra còn hỗ trợ cho phép xem các độ đo như acurracy, precision, recall, f1-score, các thông tin khác cho từng class đã detect và ma trận nhầm lẫn.

A picture containing table

Description automatically generated

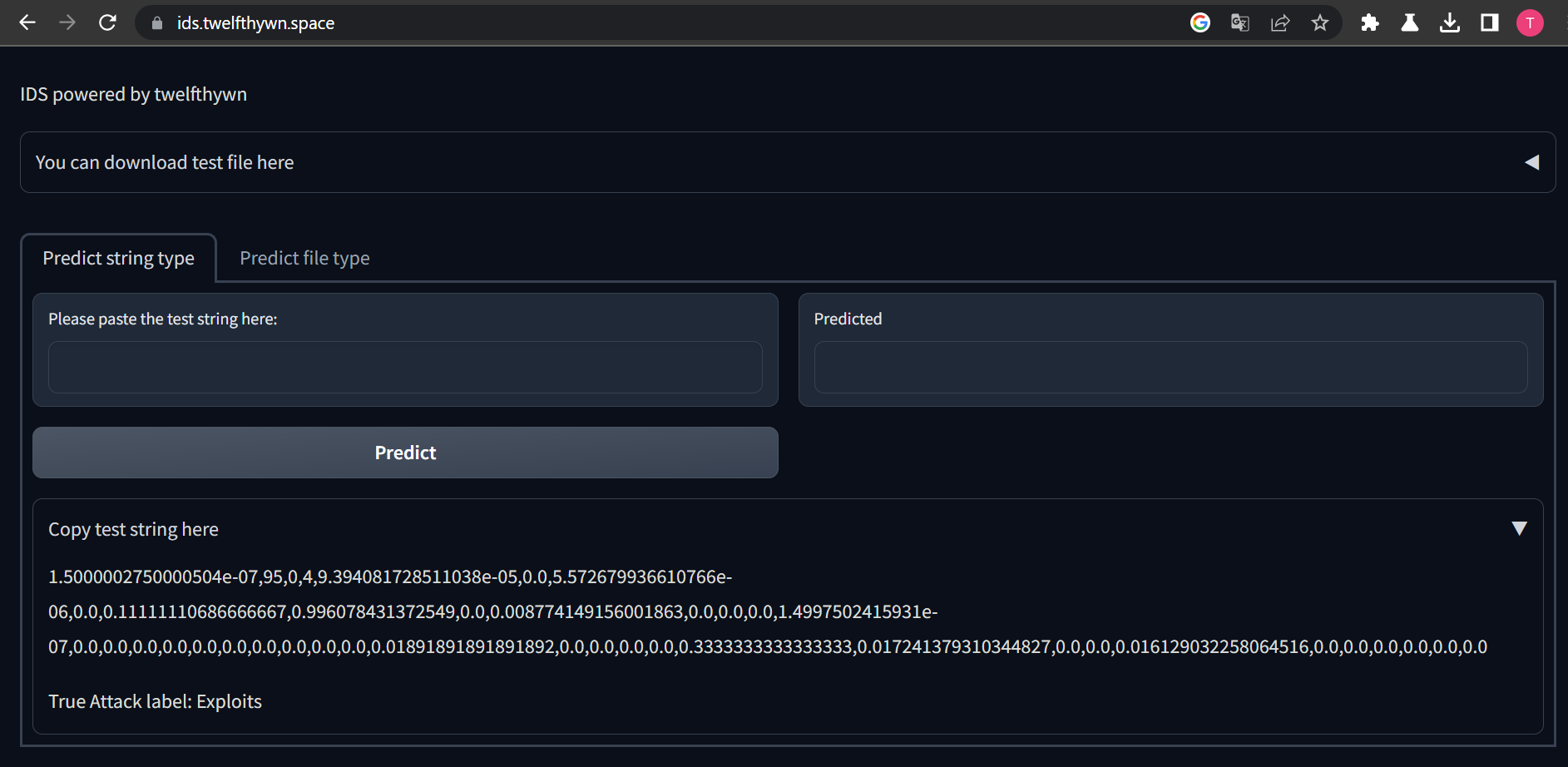
1. Hỗ trợ cho người sử dụng bằng 2 file thực thi .bat

Cuối cùng, để hỗ trợ cho việc thuận tiện demo, chúng tôi còn tạo 2 file exe\_client.bat và exe\_server.bat bên ngoài, chỉ cần mở 2 file lên là có thể thực hiện demo.

1. **Dạng 2**: Deploy model thành dạng website application

Trang web chúng tôi chia làm 2 loại chức năng chính là phát hiện xâm nhập dụa trên chuỗi đầu vào và dựa trên file dataset các bạn tải lên website của chúng tôi.

### Phát hiện xâm nhập dựa trên chuỗi đầu vào



1. Tổng quan trang web gồm 2 phần

Đầu tiên trang web của chúng tôi gồm có cá chức năng như giới thiệu, link để tải file test.txt, file test.csv, link source code cũng như chúng năng phát hiện xâm nhập dựa trên chuỗi. Đồng thời, để tạo thuận tiện hơn cho người sử dụng, chúng tôi còn để dòng mẫu cho người dùng có thể trực tiếp nhập vào và thử sử dụng tính năng của chúng tôi.

Text

Description automatically generated

1. Giới thiệu, link tải dataset và source code

Bao gồm các link tải file test trên google drive, link github, đồng thời người dùng cũng có thể tìm thấy file test cũng như source code trong link github mà chúng tôi đã cung cấp.

Text

Description automatically generated

1. Nội dung file test.txt và test.csv

Chúng tôi thực hiện lấy ngẫu nhiên một record trong file test. Bỏ giá trị index đầu và giá trị label “Exploits” cuối để thực hiện đưa vào phát hiện xâm nhập.

Graphical user interface, text

Description automatically generated

1. Kết quả sau khi phát hiện xâm nhập dựa trên chuỗi thành công

Tiến hành bấm Submit trường dữ liệu chúng ta vừa lấy và thực hiện phát hiện xâm nhập bằng nút “Predict”. Kết quả hiện lên kiểu xâm nhập độc hại như nhãn đúng ban đầu của trường dữ liệu là “Exploits” với thông điệp “Warning!!!!! Attacked: Exploits”.

### Pháp hiện xâm nhập dựa trên file đầu vào

A screenshot of a computer

Description automatically generated with medium confidence

1. Tổng quan chức năng phát hiện xâm nhập dựa trên file

Về loại thứ 2 của chúng tôi là phát hiện xâm nhập dưa trên file input đầu vào. Khi người dùng thực hiện trích xuất dữ liệu từ trang web của các bạn và lưu file dưới dạng đuôi .txt và .csv, họ có thể tải nó lên website của chúng tôi.

A screenshot of a computer

Description automatically generated with medium confidence

1. Tổng quan chức năng upload file test và review file

Ở đây, chúng tôi còn hỗ trợ người dùng xem lại nội dung trong file mình đã tải lên và thực hiện phát hiện xâm nhập bằng nút “Predict”

A screenshot of a computer

Description automatically generated with medium confidence

1. Kết quả sau khi phát hiện xâm nhập dựa trên file thành công

Sau khi thực hiện phát hiện xâm nhập bằng nút “Predict”, chúng tôi có cung cấp một số chức năng như xem kết quả phát hiện xâm nhập, các thông số như Acurracy, Precision, Recall, F1-score, ma trận nhầm lẫn,..

# Kết luận

- Tùy vào yêu cầu bài toán, cần lựa chọn thuật toán phù hợp và tối ưu.

- Mỗi thuật toán và các model khác nhau cần lựa chọn các kĩ thuật Data preprocessing và Feature engineering hợp lý.

- Hiện nay, Machine Learning và Deep Learning đang ngày càng thể hiện được tầm quan trọng cũng như vai trò không thể thiếu của mình. Nhất là trong các hệ thống phát hiện xâm nhập, các hệ thống hoạt động trên môi trường mạng hay các hệ thống hoạt động trên lượng dữ liệu liên tục thay đổi theo thười gian.

##### Acknowledgment *(Heading 5)*

Cuối cùng, với kết quả của báo cáo này, chúng em xin gửi lời cảm ơn dến thầy Nguyễn Tấn Cầm đã nhiệt tình hỗ trợ chúng em hoàn thành báo cáo môn học và cũng đã dạy chúng em rất nhiều kiến thức mới và rất bổ ích. Chúng em xin cảm ơn.

##### Tham Khảo

1. Network intrusion detection system: A systematic study of machine learning and deep learning approaches.
2. Li, Z., Y. Li, and L. Xu. Anomaly intrusion detection method based on k-means clustering algorithm with particle swarm optimization. in 2011 international conference of information technology, computer engineering and management sciences. 2011. IEEE.
3. Ahmad, I.J.I.J.o.D.S.N., Feature selection using particle swarm optimization in intrusion detection. 2015. 11(10): p. 806954.
4. Bamakan, S.M.H., et al., A new intrusion detection approach using PSO based multiple criteria linear programming. 2015. 55: p. 231-237.
5. Ali, M.H., et al., A new intrusion detection system based on fast learning network and particle swarm optimization. 2018. 6: p. 20255-20261.
6. Shokoohsaljooghi, A. and H.J.I.J.o.I.T. Mirvaziri, Performance improvement of intrusion detection system using neural networks and particle swarm optimization algorithms. 2019: p. 1-12.
7. Ali, G.A. and A. Jantan. A new approach based on honeybee to improve intrusion detection system using neural network and bees algorithm. in International Conference on Software Engineering and Computer Systems. 2011. Springer.
8. Bae, C., et al., A novel anomaly-network intrusion detection system using ABC algorithms. 2012. 8(12): p. 8231-8248.
9. Mahmod, M.S., et al., Hybrid intrusion detection system using artificial bee colony algorithm and multi-layer perceptron. 2015. 13(2): p. 1.
10. Sharma, S., A. Gupta, and S. Agrawal. An intrusion detection system for detecting denial-of-service attack in cloud using artificial bee colony. in Proceedings of the International Congress on Information and Communication Technology. 2016. Springer.
11. Hajisalem, V. and S.J.C.N. Babaie, A hybrid intrusion detection system based on ABC-AFS algorithm for misuse and anomaly detection. 2018. 136: p. 37-50.
12. Yang, J., et al. Modified naive bayes algorithm for network intrusion detection based on artificial bee colony algorithm. in 2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS). 2018. IEEE.
13. Eesa, A.S., Z. Orman, and A.M.A.J.E.S.w.A. Brifcani, A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems. 2015. 42(5): p. 2670-2679.
14. Eesa, A.S., A.M. Abdulazeez, and Z.J.S.J.o.U.o.Z. Orman, A DIDS Based on The Combination of Cuttlefish Algorithm and Decision Tree. 2017. 5(4): p. 313-318.
15. Mohammadi, S., et al., Cyber intrusion detection by combined feature selection algorithm. 2019. 44: p. 80-88.
16. Hoque, M.S., et al., An implementation of intrusion detection system using genetic algorithm. 2012.
17. Kumar, K.P.M.J.I.J.o.S., Engineering and C. Technology, Intrusion Detection system for malicious traffic by using PSO-GA algorithm. 2013. 3(6): p. 236.
18. Ghanem, T.F., W.S. Elkilani, and H.M.J.J.o.a.r. Abdul-Kader, A hybrid approach for efficient anomaly detection using metaheuristic methods. 2015. 6(4): p. 609-619.
19. Hamamoto, A.H., et al., Network anomaly detection system using genetic algorithm and fuzzy logic. 2018. 92: p. 390-402.
20. Pradeep Mohan Kumar, K., et al., Intrusion detection system based on GA‐fuzzy classifier for detecting malicious attacks. 2019: p. e5242.