

تحلیل ترافیک شبکه با یادگیری ماشین برای تشخیص سریع تر حمله قطع سرویس توزیع یافته

محمد ظهیری^{۱*}، کیمیا شیرینی^۲، سینا صمدی قره ورن^{۳*}

۱- دانشجوی کارشناسی ارشد، دانشگاه شهید مدنی آذربایجان، ۲- دانشجوی دکتری، ۳- استاد، دانشگاه تبریز، تبریز، ایران

(دریافت: ۱۴۰۲/۰۷/۲۶، بازنگری: ۱۴۰۲/۰۹/۲۰، پذیرش: ۱۴۰۲/۱۰/۰۴، انتشار: ۱۴۰۲/۱۱/۰۱)

DOR: <https://dor.isc.ac/dor/20.1001.1.26762935.1402.14.4.6.2>

چکیده

در این تحقیق، به بررسی و تحلیل چهار الگوریتم یادگیری ماشین برای شناسایی حملات DDoS پرداخته شده است. برای این منظور، از پایگاه داده (CIC-IDS2017) Intrusion Detection Evaluation Dataset استفاده شده است که شامل نمونه‌های ترافیک بات نت است. الگوریتم‌های KNN، RF، Naive Bayes و J48 با استفاده از ویژگی‌های انتخاب شده با تابع SelectKBest و کتابخانه scikit-learn آموزش داده شدند. نتایج نشان می‌دهند که الگوریتم‌های RF، KNN و J48 از نظر دقت بسیار به هم نزدیک بوده و عملکرد خوبی در شناسایی ترافیک بات نت و ترافیک معمولی داشته‌اند. الگوریتم RF با F1-score بالاتر نسبت به KNN، دقت بیشتری در شناسایی ترافیک بات نت ارائه داده است. از سوی دیگر، الگوریتم Naive Bayes، با وجود دقت کلی بالا، در شناسایی ترافیک بات نت عملکرد ضعیفی داشته و Recall و Precision آن برای دسته‌بندی بات نت بسیار پایین است. الگوریتم J48 نیز عملکرد نسبتاً خوبی داشته، ولی به دلیل مقدار پایین Recall، بخش قابل توجهی از ترافیک حمله به اشتباه به عنوان ترافیک معمولی شناسایی شده است. این تحقیق تأکید دارد که برای مقابله با حملات DDoS، استفاده از الگوریتم‌های مدرن یادگیری ماشین می‌تواند دقت و سرعت شناسایی را بهبود بخشد. در آینده، آزمایش مدل‌ها در شرایط واقعی و با داده‌های متنوع‌تر، به منظور افزایش دقت و قابلیت‌های مدل‌های شناسایی حملات اینترنتی، ضروری است.

کلیدواژه‌ها: یادگیری ماشین، ترافیک شبکه، حمله DDos، درخت تصمیم، نزدیک‌ترین همسایه.

Network Traffic Analysis with Machine Learning for Faster Detection of Distributed Denial of Service Attack

M. Sahiri^{1*}, K. Hirini, S. Samadi Gharehveran^{2*}

Tabriz University

(Received: 2023/10/18, Revised: 2023/12/10, Accepted: 2024/12/25, Published: 2024/01/21)

Abstract

In this research, four machine learning algorithms for detecting DDoS attacks have been investigated and analyzed. For this purpose, the Intrusion Detection Evaluation Dataset (CIC-IDS2017) database, which includes botnet traffic samples, has been used. KNN, RF, Naive Bayes, and J48 algorithms were trained using the selected features with the SelectKBest function and the scikit-learn library. The results show that the RF, KNN, and J48 algorithms are very close in terms of accuracy and have performed well in identifying botnet traffic and normal traffic. The RF algorithm with a higher F1-score compared to KNN has provided more accuracy in identifying botnet traffic. On the other hand, the Naive Bayes algorithm, despite its high overall accuracy, has performed poorly in identifying botnet traffic, and its precision and recall are very low for botnet classification. The J48 algorithm has also performed relatively well, but due to the low recall value, a significant part of the attack traffic has been mistakenly identified as normal traffic. This research emphasizes that to deal with DDoS attacks, the use of modern machine learning algorithms can improve the accuracy and speed of identification. In the future, it will be necessary to test the models in real-world conditions with more diverse data in order to increase the accuracy and capabilities of Internet attack detection models.

Keywords: Ddos Attack, Decision Tree, Machine Learning, Nearest Neighbor, Network Traffic.

*Corresponding Author E-mail: fathollahi.manoochehr@gmail.com

۱. مقدمه

تا قبل از دهه ۱۹۸۰، مفهوم امنیت شبکه به شکل امروزی آن مورد توجه نبود؛ چون بیشتر رایانه‌ها به اینترنت متصل نبودند. با افزایش تعداد رایانه‌های متصل به اینترنت، کمبود امنیت شبکه و مفاهیم آن مورد توجه قرار گرفت. حمله‌های اینترنتی یکی از مهم‌ترین موضوعات در شبکه است. حمله‌های اینترنتی می‌توانند در هر زمان و مکانی در سطح‌ها و انواع مختلفی اتفاق بیفتند. در سال‌های اخیر، پرتکرارترین حمله اینترنتی، حمله DDoS بوده است. زیرا بقیه حمله‌ها تقریباً یک راه حل مؤثر دارند یا حداقل راحت‌تر قابل رفع هستند. اما حمله DDoS تا الان هیچ راه حل تضمینی برای پیشگیری یا حتی تشخیص نداشته است. ترسناک‌ترین موضوع در مورد DDoS، حجم آن است. در ۵ سال گذشته شاهد بزرگ‌ترین حمله‌ها در طول تاریخ اینترنت بوده‌ایم که هر کدام از آن‌ها چند گیگابایت ترافیک و بیش از ۴۰ میلیون درخواست در ثانیه را به سرور می‌فرستد. از طرف دیگر اخیراً حمله DDoS در دستگاه‌های IoT و شبکه‌های SDN افزایش چشمگیری داشته است [۱].

تحلیل ترافیک شبکه نقش بسیار حیاتی در مدیریت و امنیت شبکه‌های مدرن دارد. باتوجه به افزایش حملات سایبری و پیچیدگی‌های آن‌ها، تحلیل ترافیک شبکه به عنوان ابزاری برای شناسایی الگوهای غیرعادی و حملات به کار می‌رود. در مقاله حاضر، تمرکز اصلی بر روی تحلیل ترافیک برای شناسایی حملات DDoS است. این نوع حملات به دلیل توزیع‌شدگی منابع حمله و حجم بالای ترافیک، به راحتی از سیستم‌های سنتی شناسایی عبور می‌کند؛ لذا استفاده از روش‌های پیشرفته تحلیل ترافیک، مانند یادگیری ماشین، می‌تواند به شناسایی به موقع و کاهش اثرات این حملات کمک کند.

حمله DDoS در مقایسه با نوع اولیه خودش (DoS) یک تفاوت کوچک ولی مهم دارد؛ این حمله منبع مشخصی ندارد و این موضوع کار را سخت می‌کند؛ برای همین دیوارهای آتش^۱ و IDSها به سختی می‌توانند آن را تشخیص دهند. از طرفی، DDoS انواع مختلفی دارد و در لایه‌های مختلفی از معماری شبکه می‌تواند اتفاق بیفتد. حمله‌های لایه ۷- آسیب‌پذیری اپلیکیشن را هدف قرار می‌دهد - حمله پروتکل SYN Flood و حمله‌های حجمی - پهنای باند شبکه را اشغال می‌کند. متأسفانه ایمن کردن شبکه نسبت به تمامی این موارد کار دشواری است [۲].

در بسیاری از موارد، استفاده از سخت‌افزارهای امنیتی مانند Firewall، IPS، IDS و Load Balancer برای حمله‌های کوچک مناسب هستند. این سخت‌افزارها در حمله‌های بزرگ به دلیل مصرف ram و cpu بالا ممکن است از کار بیفتند. گاهی هم هدف حمله، خود این‌ها هستند. در این مقاله، تلاش می‌کنیم تا با

استفاده از روش‌های یادگیری ماشین ترافیک شبکه - ترافیک جمع‌آوری شده در طول زمان - را بررسی و تحلیل کنیم تا در نهایت الگویی از ترافیک شبکه پیدا کنیم و با این الگو راه‌حلی برای تشخیص سریع‌تر حمله یا حتی رفع آن پیدا کنیم.

در اواخر ماه می سال ۲۰۲۲، یکی از مشتریان سرویس رایگان Cloudflare مورد حمله DDoS قرار گرفت. این حمله با استفاده از باتنت Mantis انجام شد. حجم این حمله ۲۶ میلیون درخواست HTTPS در ثانیه گزارش شد که تا آن زمان سابقه نداشت [۳]. چند روز بعد، حمله‌ای ۷۶٪ بزرگ‌تر، برای یکی از مشتریان Google Cloud رخ داد. حجم این حمله ۴۶ میلیون درخواست HTTPS در ثانیه گزارش شد. برای درک بهتر، این عدد معادل آن است که ترافیک ۲۴ ساعت از wikipedia را در ده ثانیه جای دهیم [۴]. در فوریه سال ۲۰۲۱ هم صرافی رمزارز EXMO مورد حمله‌ای با ۳۰ گیگابایت ترافیک بر ثانیه قرار گرفت [۵، ۶].

این مقاله به بررسی و تحلیل حملات DDoS با استفاده از روش‌های یادگیری ماشین می‌پردازد. هدف اصلی تحقیق، تحلیل ترافیک شبکه و شناسایی الگوهای مؤثر در شناسایی حملات DDoS است. با استفاده از داده‌های جمع‌آوری‌شده و روش‌های پیشرفته یادگیری ماشین، ما به دنبال ایجاد مدلی برای تشخیص سریع‌تر و مؤثرتر حملات DDoS خواهیم بود. به‌ویژه، تلاش خواهیم کرد تا الگوریتم‌های مختلف یادگیری ماشین را بررسی و مقایسه کنیم تا بهترین راه‌حل‌ها را برای مقابله با این تهدیدات ارائه دهیم.

این مقاله از تکنیک‌های پیشرفته انتخاب ویژگی، از جمله 'SelectKBest' با تابع 'chi2'، برای شناسایی مؤثرترین ویژگی‌ها در تشخیص حملات DDoS استفاده کرده است. این رویکرد به بهبود دقت و کارایی الگوریتم‌های یادگیری ماشین در این زمینه کمک کرده است. همچنین عملکرد چهار الگوریتم مختلف یادگیری ماشین (J48، RF، KNN، Naive Bayes) را در تشخیص حملات DDoS به طور جامع مقایسه کرده و با استفاده از معیارهای مختلف از جمله دقت، recall، و F1-score، به شناسایی الگوریتم‌های برتر پرداخته است. برای اولین بار، این تحقیق پیشنهاد می‌دهد که الگوریتم‌های یادگیری ماشین بررسی‌شده در محیط‌های واقعی و عملیاتی تست شوند تا عملکرد آن‌ها در شرایط واقعی ارزیابی شده و نقاط ضعف و قوت آن‌ها شناسایی گردد.

۱-۱. حمله DDoS

حمله قطع سرویس^۲ زمانی اتفاق می‌افتد که سرویس، دستگاه یا شبکه قابلیت ارائه سرویس به کاربران واقعی خود را نداشته باشد. حمله قطع سرویس توزیع شده زیرمجموعه DoS است و زمانی

^۲ Denial-of-Service (DoS)^۱ Firewall

DDoS ترافیک عادی قابل تشخیص است؛ اما برچسب زدن به ترافیک مشکوک کار سختی است و Machine Learning در این زمینه به ما کمک می کند. همچنین نتایج نشان داده که تشخیص حمله های خفیف تر با Machine Learning بسیار بهتر عمل می کند [۸].

ابزارهای زیادی برای ذخیره ترافیک استفاده می شوند. مثلاً IDSها که کل ترافیک شبکه از آن ها عبور می کند. TCPDump یکی دیگر از این ابزارهاست که می تواند ترافیک شبکه را به قالب pcap ذخیره کند. ابزار دیگر که برای این کار استفاده می شود WireShark است که قابلیت جمع آوری ترافیک و نمایش لحظه ای اطلاعات ترافیک و جداسازی برخی از ویژگی های ترافیک را نیز دارد.

ترافیک خام برای ما اهمیتی ندارد، زیرا نمی توان از آن برای مدل های Machine Learning استفاده کرد؛ باید اطلاعات مفید ترافیک از آن جدا شود. برای این کار می توان از ابزار CICFlowMeter استفاده کرد. این ابزار قابلیت استخراج بیش از ۷۰ ویژگی از ترافیک را دارد. طبقه بندی ترافیک شبکه با استفاده ابزارهای یادگیری ماشین توسط ویژگی های رایجی مانند متوسط سائز پکت ها، بیت ریت، بازه زمانی، نوع پکت، درگاهی مقصد، حجم آن و غیره انجام می شود. از این ویژگی ها برای تشخیص نوع ترافیک و در نهایت طبقه بندی آن ها به دودسته حمله یا معمولی استفاده می شود [۱۰]. این ویژگی ها برای هر نوع از DDoS متفاوت هستند. در این مقاله ما ویژگی های حجم کامل پکت درخواست^۱، حجم کامل پکت پاسخ^۲، آهنگ ارسال پکت های هر درخواست بر ثانیه^۳، پورت مقصد^۴، طول هدر پاسخ^۵، آهنگ ارسال بیت درخواست بر ثانیه^۶ است. در انتخاب ویژگی های مورد نیاز، علاوه بر استفاده از مقالات پیشین، از هوش مصنوعی ChatGPT نیز کمک گرفته شده است.

تمام اطلاعات به دست آمده از ترافیک قابل استفاده و درست نیستند؛ گاهی حتی دارای اهمیت نیز نیستند. باید اطلاعات به دست آمده را قبل از به کارگیری در یادگیری ماشین پیش پردازش کنیم [۷]. یعنی حذف کردن مقادیر خالی و بی-اهمیت. ما برای تحلیل میزان تأثیر هر ویژگی انتخاب شده، از ابزار SelectBest با ورودی های chi2 و k=4 استفاده کرده ایم. تابع chi2 برای مقادیر غیر منفی کار می کند، اما در ترافیک شبکه همه مقادیر مثبت هستند، پس به راحتی می توان مقادیر منفی را حذف کرد و در نهایت داده های متنی را به عددی تبدیل کرد.

اتفاق می افتد که حمله کننده از چندین دستگاه متخلف با IP های مختلف برای اختلال در ترافیک قربانی استفاده کند [۷].

حمله DDoS انواع مختلفی دارد که از معروف ترین آن ها می توان به Smurf، HTTP-flood، UDP-flood و SIDDOSYN-flood است. پایه تمام این روش ها فرستادن حجم زیادی از ترافیک به شبکه قربانی است [۸]. حمله DDoS ابزارهای مختلفی دارد که تعداد آن ها به بیش از ۲۰ مورد می رسد. هر کدام از این ابزارها، برای یک یا چند نوع خاص از حمله ساخته شده اند. چند مورد از پر استفاده ترین آن ها به طور خلاصه در جدول (۱) آمده است [۹].

جدول ۱. ابزارهای حمله DoS و DDoS

نام	DoS / DDoS	نوع حمله	سیستم عامل	ایجاد بات نت
SlowLoris	DoS	http	windows, linux	no
LOIC	DoS, DDoS	tcp, udp, icmp, http	windows, linux, mac, android	yes
Hping	DoS	icmp, udp, tcp	linux, windows	no
Aldi bot-net	DDoS	http, tcp	windows	yes
HOIC	DDoS	http	windows	yes
HULK	DoS, DDoS	http	linux, windows	no

۲-۱. روش های تشخیص حمله

تکنیک های زیادی برای تشخیص DDoS وجود دارد مانند: Statistical، Machine Learning، Shallow Machine Learning و غیره. از بین تمام فن ها، Machine Learning از همه مناسب تر است و روش های دیگر هر کدام محدودیت هایی دارند [۹].

• **روش Statistical:** از محدودیت های تکنیک Statistical این بود که بیشتر روش های آن مبتنی بر آستانه های تعریف شده توسط برنامه نویس ها و وابسته به اطلاعات قبلی بود. این روش با تغییر روش حمله کارایی آن به شدت کاهش پیدا می کرد و سیستم پویایی برای تغییر در سطح آستانه های حمله نداشت. از طرفی انتخاب ویژگی مناسب برای تحلیل ترافیک در این روش کار دشواری بود و احتیاج به حجم بالایی از ترافیک گذشته داشت [۷].

• **روش Shallow Machine Learning:** مشکل تکنیک Shallow Machine Learning این است که با مقادیر کم ترافیک خوب کار می کند؛ اما در حجم های بالا با مشکل روبه رو می شود. مشکل دیگر نیاز به به روزرسانی های درست متناسب با نوع حمله است [۸]. این روش مسئله را به زیر مسئله های کوچک تر می شکند و در نهایت با حل زیر مسئله ها، مسئله نهایی را حل می کند. این روش از لحاظ سرعت یادگیری بسیار سریع تر از یادگیری ماشین است؛ اما در نهایت روش مناسبی برای حمله های امروزی نیست.

• **روش Machine Learning:** دلیل مناسب بودن روش های Machine Learning این است که می توانند از اطلاعات، ویژگی های مناسب برای تحلیل را استخراج کنند. در بحث

¹ Total Length of Fwd Packets

² Total Length of Bwd Packets

³ Flow Packets/s

⁴ Destination Port

⁵ Bwd Header Length

⁶ Flow Bytes/s

پیش‌پردازش اطلاعات برای جلوگیری از خطا و افزایش عملکرد الگوریتم ضروری است [۱۱].

بعد از انجام پیش‌پردازش، باید مؤثرترین ویژگی‌های دیتاست خود را پیدا کنیم؛ تعداد زیاد ویژگی باعث سخت‌تر شدن کار با داده، کاهش سرعت آموزش و پیش‌بینی و بیش‌برازش در مدل آموزشی می‌شود. کاهش ابعاد باعث سرعت بیشتر در حین آموزش و نتایج دقیق‌تر می‌شود. از جمله روش‌های رایج کاهش ابعاد SelectBest با ورودی chi2 در پکیج scikit-learn است. برای نتیجه‌گیری بهتر و دقیق‌تر در Machine Learning باید پارامترهای ورودی مدل در بهترین حالت ممکن باشند. برای این کار می‌توان از روش Hyperparameter Tuning استفاده کرد. برای جستجوی ابر پارامترها الگوریتم‌های متفاوتی وجود دارند؛ مانند GridSearchCV.

۳-۱. الگوریتم‌های یادگیری ماشین

پس از بررسی مقالات و جستجو منابع، از بین تمامی الگوریتم‌های یادگیری ماشین، تعدادی از آن‌ها که برای این کار مناسب‌ترند را انتخاب کردیم؛ بعضی از این الگوریتم‌ها برای آموزش نیاز به داده آموزشی کمتری نسبت به سایر الگوریتم‌ها دارند [۱۱] و برخی دیگر هم قدرت بالایی در دسته‌بندی دارند [۱۲]. چند نمونه از الگوریتم‌های رایج به شرح زیر هستند:

- **Random Forest:** این الگوریتم بر اساس تعداد زیادی از درخت تصمیم کار می‌کند و می‌تواند بر اساس تعداد مشخصی از پارامترها تصمیم‌گیری کند. ساختار درخت شامل والد و فرزند است که ریشه اولین گره آن و برگ‌ها آخرین گره‌ها هستند. برگ‌ها نتیجه تصمیم‌گیری الگوریتم بر اساس شرایط هستند [۱۱]. از ویژگی‌های مثبت Random Forest این است که اطلاعات ورودی نیاز به پیش‌پردازش زیاد ندارند و هم با داده‌های عددی و هم غیرعددی به راحتی کار می‌کند. دیگر اینکه این الگوریتم برای داده‌های با ابعاد بالا بسیار مناسب است و می‌تواند هم روابط خطی و هم غیرخطی را کنترل کند. از معایب آن هم این است که برای پایگاه‌داده‌های بزرگ، در قسمت آموزش بسیار سنگین عمل می‌کند و پیدا کردن ابر پارامتر را دشوارتر می‌کند.

- **KNN:** این مدل بر اساس نزدیک بودن همسایه‌ها کار می‌کند و مبتنی بر این است که اشیای مشابه نزدیک به هم قرار می‌گیرند. این مدل بیشتر برای تشخیص الگو کاربرد دارد. در این روش، مقدار K باید با دقت انتخاب شود. با تغییرات K مقدار پایداری^۱ مدل می‌تواند افزایش پیدا کند؛ اما اگر از مقدار مشخصی بیشتر شود مقدار خطاها هم‌افزایش پیدا می‌کند [۱۱، ۱۴]. از مزایای این الگوریتم این است که نیاز به

آموزش‌های دوره‌ای ندارد، برای پیاده‌سازی بسیار ساده است و داده آموزشی جدید در هر زمانی می‌تواند به مدل اضافه شود. معایب آن نیز این است که با پایگاه‌داده‌های خیلی بزرگ و با ابعاد بالا به خوبی کار نمی‌کند؛ زیرا محاسبه فاصله برای تمامی رکوردها بسیار زمان‌بر است، دیگر اینکه باید ویژگی‌های پایگاه‌داده مورد استفاده حتماً از نوع عددی باشند و از قبل نرمال‌سازی شده و مرتب باشند.

- **Naive Bayesian:** این مدل یک مدل بر پایه احتمال است که بر اساس نظریه Bayes طراحی شده است (قانون بیز). این الگوریتم از یک الگوریتم تشکیل نشده، بلکه مجموعه‌ای از الگوریتم‌ها است که بر پایه نظریه Bayes کار می‌کنند. این الگوریتم احتمال وقوع یک احتمال را به گونه‌ای محاسبه می‌کند که انگار اتفاق افتاده است. از مزایای این الگوریتم این است که می‌تواند با داده‌هایی با ویژگی‌های زیاد هم به خوبی کار کند [۸، ۱۲]. دیگر مزایای آن، راحتی پیاده‌سازی، احتیاج نداشتن به داده آموزشی زیاد است و اینکه سرعت بالایی دارد تا حدی که می‌تواند برای پیش‌بینی‌های در لحظه هم استفاده شود. از ضعف‌های آن این است که تمامی ویژگی‌های دیتا را مستقل فرض می‌کند که کارایی این الگوریتم را در دنیای واقعی محدود می‌کند.

- **J48:** این الگوریتم یک الگوریتم دسته‌بندی است که درخت تصمیم‌هایی بر اساس نظریه اطلاعات تولید می‌کند. این الگوریتم توسعه‌یافته الگوریتم ID3 رأس کوبلان است. در این الگوریتم درخت تصمیم‌های تولید شده توسط C4.5 برای دسته‌بندی استفاده می‌شوند. در پیاده‌سازی C4.5، به این الگوریتم ویژگی‌های دیگری مانند محاسبه برای مقادیر گم شده، بازه‌بندی مقادیر پیوسته اضافه شده است.

کارهای مرتبط بسیاری در سال‌های اخیر در زمینه پیاده‌سازی تشخیص DDos به کمک یادگیری ماشین انجام شده و هر کدام از الگوریتم‌ها، ویژگی‌ها و پایگاه‌داده‌های خاصی استفاده کرده‌اند. اکثر الگوریتم‌های مطرح شده در این مقالات شامل Decision Tree، Naive bayesian، Random Forest و KNN هستند [۸، ۱۴]. در بسیاری از کارهای گذشته همچنان از پیاده‌سازی‌های قدیمی‌تر مانند روش Statistical و Shallow Machine Learning استفاده شده است که همان‌طور که پیش‌ازین گفته شد، امروزه دیگر روش‌های مؤثری در تشخیص DDos نیستند. همچنین بخشی از این کارها از مفهوم مشابهت ترافیک استفاده کرده‌اند تا DDos را تشخیص دهند.

بهال و همکاران [۸] از پایگاه‌داده جمع‌آوری شده توسط Mouhammd Alkasassbe، Ahmad B.A Hassant، Hazi Al-Naymat و Mohammad Almsidin استفاده کردند که شامل چهار نوع حمله DDos می‌شود؛ اما از لحاظ تعداد رکوردها بسیار کمتر

^۱ Stability

استفاده از ابزار hping3 برای تولید پایگاه داده که احتمال دارد مدل نهایی، بعد از آموزش روی دیگر پایگاه داده ها، نتواند به خوبی عمل کند. استفاده کردن از ۳ ویژگی و دقت ۱۰۰٪ برای RandomForest نیازمند بررسی بیشتر است.	NaiveBayes – 96.93%			
دو مورد از الگوریتم های انتخابی ما در بین الگوریتم های این مقاله وجود نداشت که نیازمند پایاده سازی بود.	NaiveBayes – 97.65% KNN – 99.88% RandomForest – 100%	SYN-Flood ICMP-Flood	۲۰۲۰	[۹]
استفاده از بیش از ۲۰ ویژگی برای آموزش مدل ها	SVM – 84.32% DecisionTree – 94.43% NaiveBayes – 74.63% ANN – 63.97% USML – 94.67% MLP – 96.5% SVM – 95.73% KNN – 97.83% J48 – 97.89% ensemble – 99.1%	Botnet Land Smurf Neptune Teardrop Back	۲۰۱۹	[۱۰] [۱۱]

۲. روش تحقیق

روش انجام این تحقیق برای تحلیل ترافیک شبکه، در ابتدا با تقسیم پایگاه داده به دو بخش آموزش و آزمایش شروع می شود. بعد ویژگی های انتخاب شده باتوجه به مقالات پیشین و ChatGPT، از پایگاه داده جدا می شوند. در مرحله بعدی، به کمک کتابخانه Scikit-learn و با تابع SelectBest با $\text{score_func}=\text{chi2}$ و $k=4$ ویژگی های مؤثرتر انتخاب می شوند و هر مدل با ویژگی های جدید آموزش داده می شود و نتایج آن ذخیره می شود. در نهایت، در مرحله بررسی و مقایسه نتایج هر الگوریتم با دیگر الگوریتم ها مقایسه می شود.

کتابخانه scikit-learn یک کتابخانه متن باز نوشته شده با Python است. این کتابخانه به ما کمک می کند تا به راحتی Machine Learning را با کاربردهای Python ترکیب کنیم. Scikit-learn شامل مجموعه گسترده ای از روش های رگرسیون، دسته بندی، ماتریس واریانس، کاهش ابعاد، پیش پردازش اطلاعات، آزمایش و مقایسه می شود. این کتابخانه در لینک <http://scikit-learn.org> در دسترس است [۱۳].

به دلیل در دسترس نبودن تجهیزات لازم برای تهیه پایگاه داده حمله، از داده های آماده در منابع اینترنتی معتبر استفاده شده است. پایگاه داده مورد استفاده Intrusion Detection Evaluation Dataset (CIC-IDS2017) است. این دیتاست توسط UNB جمع آوری شده است. پایگاه داده اصلی شامل انواع حمله های اینترنتی مانند SQL Injection، SSH-Patator، Port Scan و XSS است. برای این مقاله فقط بخش Bot این پایگاه داده مورد استفاده قرار گرفته است. بخش Bot شامل حمله بات نت ARES یکی از بات نت های شناخته شده در اینترنت است که دارای قابلیت های زیادی از جمله حمله DDos، گرفتن نامگرفت از صفحه قربانی، اجرای دستور، انتقال فایل از سیستم به سیستم قربانی و تکتیر خودش است. نصب و راه اندازی ARES بسیار راحت و سریع است

از پایگاه داده مورد استفاده در این مقاله است. Naive Bayesian، RF، MLP و J48 چهار الگوریتم استفاده شده در این تحقیق هستند که نتایج آن ها شرح داده شده است. در پایگاه داده این مقاله اشاره ای به حمله های بات نت نشده است.

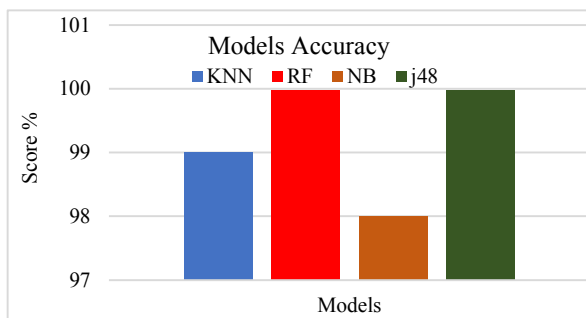
سه ویژگی delta time ، packet length و protocol باعث تسريع و ساده تر کردن بخش های preprocessing و feature extraction می شود [۱۳]. کاهش تعداد ویژگی ها باعث می شود مدل آموزشی بتواند سریعتر آموزش ببیند و از طرفی طیف وسیع تری از حملات را با آموزش روی پایگاه داده مربوطه تشخیص دهد. الگوریتم های مورد استفاده Naive bayesian، KNN و RF هستند. همچنین این بحث مطرح شده که سیستم پایاده سازی شده باید به گونه ای باشد که روی هر سیستمی با هر توان سخت افزاری اجرا شود و یکی از دلایل انتخاب تعداد کم ویژگی هم این موضوع است. در این مقاله از ابزار hping3 برای تولید ترافیک استفاد شده که این احتمال را به وجود می آورد که مدل آموزشی با ابر پارامترهای پیدا شده، نتواند روی دیگر ترافیک های DDos که با دیگر ابزارها تولید شده، به خوبی عمل کند.

گرامر [۱۴]، روش جدیدتری برای تشخیص DDos مطرح شده به گونه ای که از چهار الگوریتم Machine Learning به طور همزمان در کنار مدل های خودشان را برای تشخیص تولید میکنند و نتیجه نهایی حاصل majority voting از نتیجه به دست آمده از هر چهار مدل است. این نوآوری دقت تشخیص را تا ۲٪ افزایش داده است. اما تعداد ویژگی های انتخاب شده برای آموزش این مدل ها بیش از ۲۰ عدد است که امر آموزش مدل ها و پیدا کردن ابر پارامتر را کمی دشوار میکند.

داس و همکاران [۱۶]، موضوع حمله DDos به وسیله بات نت به طور مفصل مورد بحث و ارزیابی قرار گرفته و کمک شایانی در زمینه شناخت این نوع از حمله و درک بهتر آن به ما کرد. در نهایت هم با بررسی دقیق الگوریتم های USML، Decision Tree، Naive bayes، SVM و ANN و مقایسه نتایج آن ها به انتخاب الگوریتم ها در این مقاله کمک کرده است. اسدی و همکاران [۱۷]، روش پیشنهادی از روش یادگیری جمعی رأی گیری برای بهبود دقت و صحت در تشخیص حملات انکار سرویس استفاده می کند. روش پیشنهادی سه مرحله پیش پردازش داده ها، مرحله انتخاب و کاهش ویژگی و در نهایت مرحله پیش بینی حمله را شامل می شود.

جدول ۲. بررسی اجمالی کارهای پیشین

مرجع	سال انتشار	DDoS type	Algorithm – Accuracy%	معایب
[۶]	۲۰۲۰	UDP-Flood Smurf SIDDOS HTTP-Flood	J48 - 98.64% MLP - 98.63% RandomForest – 98.10%	نبودن حمله بات نت در پایگاه داده



شکل ۲. تفاوت انتخاب درگاه مقصد در دقت مدل‌ها
به کمک تابع SelectBest از کتابخانه scikit-learn و با
score_func=chi2 و k=4 ویژگی‌های انتخابی در مرحله تجربی را
بررسی کردیم. جدول (۴) نتایج بررسی را نشان می‌دهد.

جدول ۴. موثرترین ویژگی در تشخیص بات‌نت

امتیاز	ویژگی
1.31e7	Total Length of Fwd Packets
5.31e7	Total Length of Bwd Packets
3.9e4	Destination Port
2.52e7	Flow Packets
4.58e5	Bwd Header Length
1.7e9	Flow Bytes

باتوجه به جدول (۴)، ویژگی Backward Bytes/s موثرترین ویژگی
در تشخیص بات‌نت است. در این مرحله چهار ویژگی که بیشترین
مقدار تأثیر را داشتند انتخاب شده‌اند. پس از استاندارد کردن
مقادیر پایگاه داده، این مرحله باتوجه به الگوریتم این متفاوت است.
برای هر الگوریتم باید پارامترهای مختلفی آزمایش تا در نهایت
ابر پارامتر آن پیدا شود. برای پیدا کردن ابر پارامتر از کتابخانه
scikit-learn و تابع GridSearchCV استفاده شده است. در طول
بررسی‌ها برای پیدا کردن ابر پارامتر، الگوریتم Random Forest
به شدت کندتر از دیگر الگوریتم‌ها عمل کرد؛ اما دیگر الگوریتم‌ها
نسبتاً سرعت بیشتری داشته‌اند. ابر پارامترهای هر الگوریتم در
جدول (۵) آمده است.

جدول ۵. ابر پارامترهای در نظر گرفته شده در هر روش

الگوریتم	ابر پارامتر
Naive Bayes	var_smoothing: 1.0
K-NN	Metric: manhattan
Random Forest	n_neighbors: 2
Decision Tree (J48)	weights: uniform

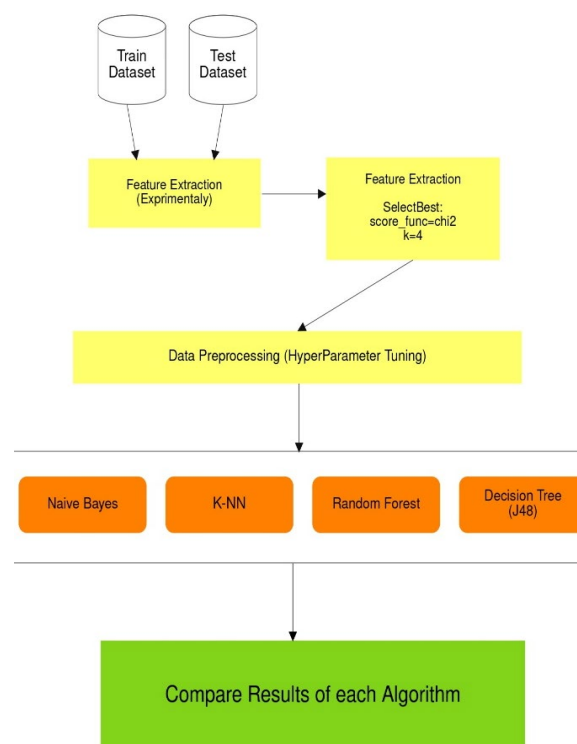
۳. نتایج و بحث

در این مرحله هر مدل با ویژگی‌های انتخاب شده با Select Best
آموزش داده می‌شوند. گزارش کامل دقت، ماتریس درهم‌ریختگی،
بازیابی، FPR، TPR، ROC Area، F-measure در ادامه آمده است.
در جدول‌های گزارش دسته‌بندی، مقدار ۱ یعنی DDoS و مقدار
صفر یعنی ترافیک معمولی.

شکل‌های (۳)، (۴)، (۵) و (۶) به ترتیب نشانگر ماتریس
درهم‌ریختگی روش‌های ارائه شده می‌باشند. در شکل (۷) نتایج
مدل‌ها به صورت جداگانه مورد بررسی قرار گرفته‌اند. پس از
بررسی نتایج به دست آمده از آزمایش‌های انجام شده، سه

و تقریباً هر کسی با کمترین دانش برنامه‌نویسی می‌تواند این بات
نت را اجرا کند که این امر حمله‌های بات‌نت را بسیار جدی‌تر و
خطرناک‌تر از قبل می‌کند.

پایگاه داده مورد استفاده از لحاظ کیفیت ترافیک
جمع‌آوری شده دارای رتبه بالایی بین محققان است؛ و برای راحتی
کار محققان آینده، سازندگان آن، بخش‌های مفید اطلاعات آن را
جدا کرده و تا حد ممکن اطلاعات آن پیش‌پردازش شده است
[۷]. این دیتاست در صبح روز جمعه، ۷ ژوئیه سال ۲۰۱۷ از
ساعت ۱۰:۰۲ تا ۱۱:۰۲ جمع‌آوری شده و حمله آن توسط
BotNet ARES انجام شده است. در تقسیم پایگاه داده، ۲۰٪ از
پایگاه داده به بخش آموزش و ۸۰٪ آن برای آزمایش
اختصاص داده شده است.

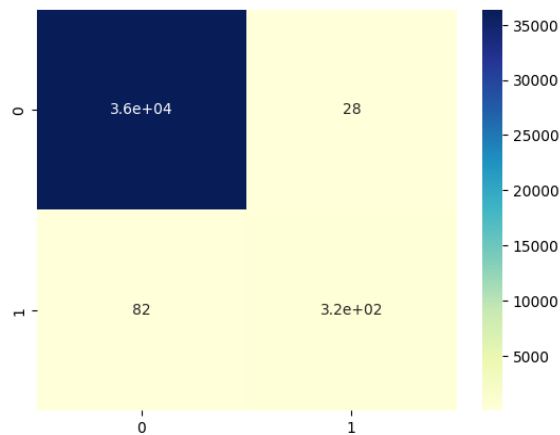


شکل ۱. فلوچارت روش ارائه شده

جدول (۳) ویژگی‌های انتخاب شده به روش تجربی را نشان
می‌دهد.

جدول ۳. ویژگی‌های انتخاب شده در روش

ردیف	ویژگی
۱	Total Length of Fwd Packets
۲	Total Length of Bwd Packets
۳	Destination Port
۴	Flow Packets
۵	Bwd Header Length
۶	Flow Bytes



شکل ۶. ماتریس درهم‌ریختگی J48

در جدول (۶)، precision نشان می‌دهد که نسبت پیش‌بینی‌های درست نسبت به تمام پیش‌بینی‌های Bot حدود ۸۵٪ است که دقت بالایی است. Recall نشان می‌دهد که ۸۲٪ از مقدار ترافیک Bot به درستی دسته‌بندی شده‌اند. fl-score نزدیک بودن مقادیر precision و recall و fl-score نشان می‌دهد که مدل هم در تشخیص ترافیک معمولی و هم در تشخیص ترافیک بات نت به خوبی عمل کرده است.

فاصله نسبتاً زیاد بین مقدار accuracy و fl-score کمی نگران‌کننده است. زیرا الگوریتم دقت بالایی در تشخیص دارد؛ اما برای تشخیص ترافیک بات نت دقت کمتری دارد که این می‌تواند در عمل باعث شود که بخشی از ترافیک حمله به سمت سرورها روانه شود.

جدول ۶. عملکرد روش KNN

	precision	recall	f1-score	support
Benign	1.00	1.00	1.00	36406
Bot	0.85	0.82	0.83	401

جدول ۷. عملکرد روش RF

	precision	recall	f1-score	Support
Benign	1.00	1.00	1.00	36406
Bot	0.85	0.82	0.83	401

جدول ۸. عملکرد روش Naive base

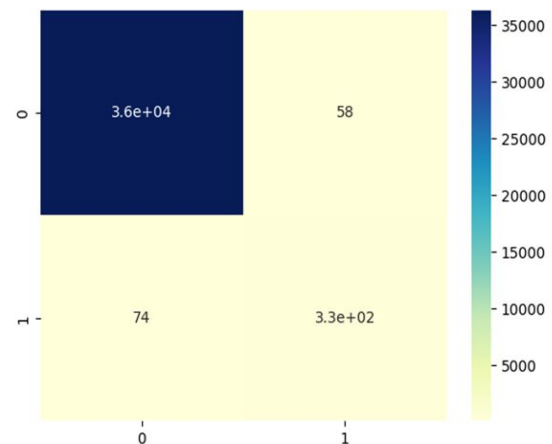
	precision	recall	f1-score	support
Benign	0.99	1	0.99	36406
Bot	0.10	0.01	0.02	401

جدول ۹. عملکرد روش J48

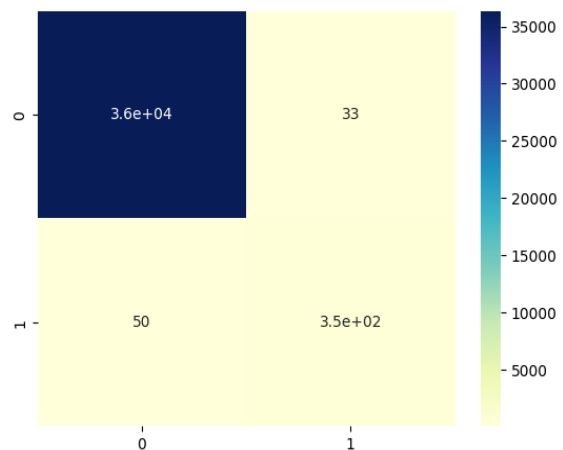
	precision	recall	f1-score	support
Benign	1	1	1	36406
Bot	0.92	0.80	0.85	401

شکل (۷) نشان می‌دهد که خطای RF نسبت به KNN در دسته‌بندی ترافیک کمتر است. در جدول (۱۰)، precision نشان می‌دهد که نسبت ترافیک Bot تشخیص داده شده درست، به کل تشخیص‌های Bot برابر ۹۱٪ است که نسبت به KNN پیشرفت

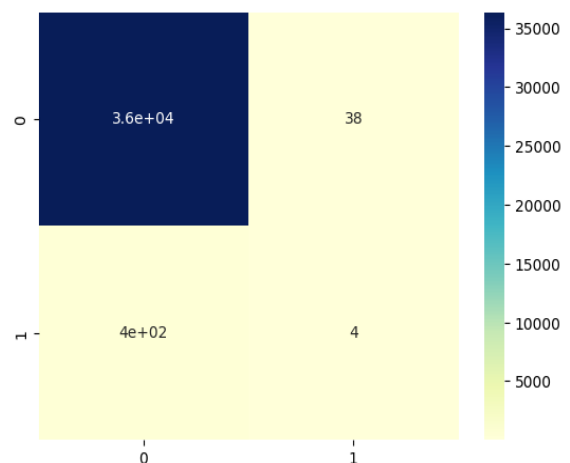
الگوریتم RF، KNN و J48 از لحاظ دقت بسیار به هم نزدیک بوده و الگوریتم Naive Bayes از همه ضعیف‌تر عمل کرده است. با اینکه دقت این الگوریتم حدود ۹۸٪ بوده ولی دیگر شاخص‌ها مانند بازایی، دقت و fl-score اختلاف زیادی با دیگر مدل‌ها داشته است. شکل (۵) نشان می‌دهد که خطای مدل KNN در تشخیص ترافیک بسیار پایین بوده و عمل کرد بالایی دارد.



شکل ۳. ماتریس درهم‌ریختگی KNN



شکل ۴. ماتریس درهم‌ریختگی Random Forest



شکل ۵. ماتریس درهم‌ریختگی Naive Bayes

خطای تشخیص تنها عامل مؤثر نیست. مقدار recall هم در میزان دقت الگوریتم تأثیر گذار است. با توجه به جدول (۱۰)، precision نشان‌دهنده این است که ترافیک حمله با دقت ۹۲٪ درست تشخیص داده شده است. Recall نشان می‌دهد که بخشی از ترافیک حمله را به اشتباه به عنوان ترافیک معمولی در نظر گرفته است. با توجه به F1-score میتوان گفت که J48 نسبت به KNN بهتر عمل کرده اما باز هم به دلیل کمتر بودن recall تا حدود ۲۰٪ از ترافیک حمله روانه سرور می‌شود. به طور کلی الگوریتم J48 عمل کرد نسبتاً بهتری از بین سایر الگوریتم‌ها داشت و بسیار به RF نزدیک است.

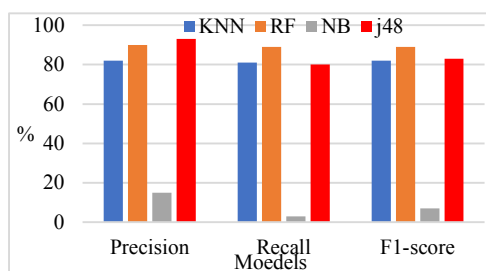
۴. نتیجه‌گیری

حمله در گذشته به کمک روش‌های قدیمی‌تر، تلاش‌های زیاد برای تشخیص DDoS شده است و هر کدام از آن‌ها در زمان خود موفق بوده‌اند؛ اما امروزه به دلیل پیچیدگی حمله‌ها و افزایش حجم و تعداد آن‌ها و از جمله راحتی و سرعت انتشار بات‌نت‌ها، بهتر است که از روش‌های مدرن‌تری مانند Machine Learning برای تشخیص استفاده کنیم که هم از لحاظ دقت و هم سرعت قابل‌اعتماد هستند. در این مقاله، ما از ۴ الگوریتم Machine Learning برای تشخیص حمله استفاده کرده‌ایم. در پایان پس از بررسی نتایج به‌دست‌آمده از آزمایش‌های انجام شده، سه الگوریتم RF، KNN و J48 از لحاظ دقت بسیار به هم نزدیک بوده و الگوریتم Naive Bayes از همه ضعیف‌تر عمل کرده است. با اینکه دقت این الگوریتم حدود ۹۸٪ بود؛ ولی دیگر شاخص‌ها مانند precision، recall و f1-score اختلاف زیادی با دیگر مدل‌ها داشته است. نتایج نشان داد که الگوریتم‌های RF، KNN و J48 از نظر دقت به طور قابل‌توجهی نزدیک به یکدیگر عمل کرده‌اند، درحالی‌که Naive Bayes عملکرد ضعیف‌تری نسبت به دیگر مدل‌ها داشته است. دقت بالای Naive Bayes حدود ۹۸٪، با این حال، شاخص‌های دیگری مانند precision، recall و f1-score آن اختلاف قابل توجهی نسبت به سایر الگوریتم‌ها نشان می‌دهد. به‌طور خاص، Naive Bayes در تشخیص ترافیک Bot به‌طور معناداری ضعیف عمل کرده و این مسئله می‌تواند منجر به از دست رفتن بخش قابل توجهی از ترافیک حمله شود. در مقایسه با KNN و RF که دقت بالایی حدود ۹۹٪ دارند، RF به‌ویژه در مورد f1-score عملکرد بهتری داشته و نشان‌دهنده توانایی بالاتر آن در دسته‌بندی دقیق‌تر ترافیک‌های حمله است. همچنین، J48 نیز عملکرد نسبتاً خوبی از خود نشان داده است، هرچند که همچنان در شناسایی ترافیک حمله نسبت به RF کمی ضعیف‌تر عمل کرده است. تحلیل‌های بیشتر شامل ماتریس‌های درهم‌ریختگی، FPR، TPR و نمودار ROC نیز نشان‌دهنده نقاط قوت و ضعف هر الگوریتم در شناسایی انواع ترافیک بودند. به‌ویژه، ماتریس‌های درهم‌ریختگی نشان می‌دهند که مدل‌های RF و KNN در تشخیص دقیق ترافیک‌های عادی و حمله موفق‌تر

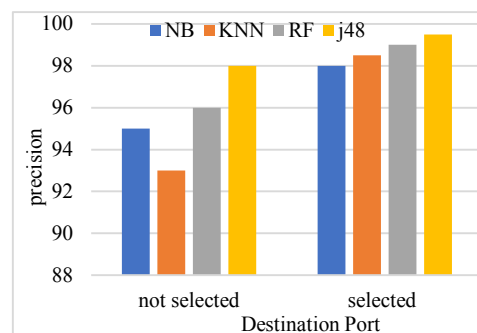
بزرگی است. همچنین recall نشان می‌دهد که ۸۸٪ از مقدار ترافیک Bot به درستی دسته‌بندی شده‌اند. f1-score مانند KNN نزدیک‌بودن مقادیر recall و precision و f1-score را نشان می‌دهد که مدل هم در تشخیص ترافیک معمولی و هم در تشخیص ترافیک بات‌نت به خوبی عمل کرده است.

جدول ۱۰. مقایسه روش‌های ارائه شده

الگوریتم	دقت
KNN	۹۹/۴۵٪
RF	۹۹/۶۷٪
Naive Bayes	۹۸/۸٪
J48	۹۹/۶۴٪



شکل ۷. مقایسه دقت روش‌های ارائه شده



شکل ۸. مقایسه F1-score، precision و recall مدل‌های ارائه شده

در RF و KNN مقدار دقت بسیار نزدیک به هم است (هر دو حدود ۹۹٪) اما f1-score بالاتر در RF نشان می‌دهد که دسته‌بندی ترافیک حمله با RF دقیق‌تر از KNN انجام شده و در عمل قابل اعتمادتر است. شکل (۸) نشان می‌دهد که الگوریتم Naive Bayes در طبقه‌بندی ترافیک حمله اصلاً خوب عمل نکرده اما ترافیک معمولی را به خوبی تشخیص داده است. با توجه به جدول (۱۰) مقادیر precision و recall و f1-score برای Bot نشان می‌دهد که مدل Naive Bayes در دسته‌بندی ترافیک حمله اصلاً خوب عمل نکرده است. تقریباً میتوان گفت که این الگوریتم نمیتواند ترافیک حمله را تشخیص دهد؛ این مورد باعث می‌شود در عمل، بخش اعظم ترافیک حمله روانه سرور شود که اصلاً قابل قبول نیست. شکل (۸) نشان می‌دهد که J48 نسبت به الگوریتم دیگر، در تشخیص‌های حمله خطای کمتری داشته است. اما کمترین

Conference on Emerging Smart Computing and Informatics. 2020, 234-237. DOI: 10.1109/ESCI48226.2020.9167642

[12] Kazemitabar, J.; Taheri, R.; Kheradmandian, H. "A Novel Technique for Improvement of Intrusion Detection via Combining Random Forrest and Genetic Algorithm"; J. Adv. Defense Sci. & Technol. 2019, 287-296 (In Persian). Dor: 20.1001.1.26762935.1398.10.3.9.5

[13] Das, S.; Mahfouz, A. M.; Venugopal, D.; Shiva, S. "DDoS Intrusion Detection Through Machine Learning Ensemble"; IEEE 19th international conference on software Quality, Reliability and Security Companion. 2019, 471-477. DOI: 10.1109/QRS-C.2019.00090

[14] Asgharian, H.; Ahmad A.; Raahemi, B. "Engineered Feature Set to Detect Flooding Attacks in SIP Based VoIP"; J. Adv. Defense Sci. & Technol. 2019, 8, 61-69 (In Persian). Dor: 20.1001.1.26762935.1396.8.1.7.5

[15] Pande, S.; Khamparia, A.; Gupta, D.; Thanh, D. N. "DDoS Detection Using Machine Learning Technique"; Recent Studies on Computational Intelligence: Doctoral Symposium on Computational Intelligence, Springer Singapore 2021, 59-68. DOI:10.1007/978-981-15-8469-5_5

[16] Abdulla, N. N.; Hasoun, R. K. "Review of Detection Denial of Service Attacks Using Machine Learning Through Ensemble Learning"; Iraqi Journal for Computers and Informatics. 2022, 48, 13-20. DOI:10.25195/ijci.v48i1.349

[17] Sattari, M. T.; Shirini, K.; Javidan, S. "Evaluating the Efficiency of Dimensionality Reduction Methods in Improving the Accuracy of Water Quality Index Modeling using Machine Learning Algorithms"; Water and Soil Management and Modelling. 2024, 4, 89-104. DOI: 10.22098/mmws.2023.12434.1241

بوده‌اند. برای تشخیص مؤثر حملات DDoS، استفاده از الگوریتم‌های یادگیری ماشین می‌تواند بهبود قابل توجهی در دقت و سرعت شناسایی داشته باشد. با این حال، برای کاربردهای عملی و واقعی، نیاز به بررسی و آزمایش این مدل‌ها در محیط‌های عملیاتی وجود دارد. علاوه بر این، هدف آینده تحقیق توسعه و آموزش مدل‌هایی است که قادر به شناسایی و طبقه‌بندی انواع مختلف حملات اینترنتی به‌طور جامع‌تری باشند. از آنجایی که این کار یک تحقیق بر روی یک پایگاه داده بوده، باید این مدل‌ها در دنیای واقعی آزمایش شوند و نتایج آن‌ها به طور کامل و دقیق بررسی شود. در آینده ما این الگوریتم‌ها را روی سرور اجرا می‌کنیم و آن‌ها را بررسی می‌کنیم. این مدل‌های فقط برای تشخیص DDoS هستند و دیگر حمله‌ها را پشتیبانی نمی‌کنند. یکی از اهداف آینده آموزش مدلی است که بتواند انواع حمله‌های اینترنتی را تشخیص و طبقه‌بندی کند.

۵. مراجع‌ها

[1] Asadi, M.; Bagheri, Z. "Detection of Denial of Service Attacks by Ensemble Learning Method"; J. Adv. Defense Sci. & Technol. 2023, 14, 51-68 (In Persian). Dor:20.1001.1.26762935.1402.14.1.5.5

[2] Wang, B.; He, Y.; Shui, Z.; Xin, Q.; Lei, H. "Predictive Optimization of DDoS Attack Mitigation in Distributed Systems using Machine Learning"; Appl. Comput. Eng. 2024, 64, 95-100. DOI:10.54254/2755 2721/64/20241350

[3] Mittal, M.; Kumar, K.; Behal, S. "Deep Learning Approaches for Detecting DDoS Attacks: A Systematic Review"; Soft computing 2023, 27, 13039-13075. DOI:10.1007/s00500-021-06608-1

[4] Kumari, P.; Jain, A. K. "A Comprehensive Study of DDoS Attacks Over IoT Network and Their Countermeasures"; Computers & Security. 2023, 127, 103096. DOI:10.1016/j.cose.2023.103096

[5] Mittal, M.; Kumar, K.; Behal, S. "Deep Learning Approaches for Detecting DDoS Attacks: A Systematic Review"; Soft Computing 2023, 27, 1-37. DOI:10.1007/s00500-021-06608-1

[6] Sattari, M. T.; Bagheri, R.; Shirini, K.; Allahverdipour, P. "Modeling Daily and Monthly Rainfall in Tabriz using Ensemble Learning Models and Decision Tree Regression"; Scientific Journal of Golestan University, 2024, 5, 31-48 DOI: 10.30488/CCR.2024.433394.1192

[7] Saleh Esfehiani, M.; Abo Ali, M. "An IDS for Detection of Active Attacks Against Routing in Mobile Ad Hoc Networks"; J. Adv. Defense Sci. & Technol. 2010, 1, 15-22 (In Persian). Dor: 20.1001.1.26762935.1389.1.1.2.1

[8] Behal, S.; Kumar, K. "Characterization and Comparison of DDoS Attack Tools and Traffic Generators: A Review"; Int. J. Netw. Secur. 2017, 19, 383-393 DOI:10.6633/IJNS.201703.19(3).07

[9] Michelena, Á.; Avelaira-Mata, J.; Jove, E.; Bayón-Gutiérrez, M.; Novais, P.; Romero, O. F.; Aláiz-Moretón, H. "A Novel Intelligent Approach for Man-in-the-Middle Attacks Detection Over Internet of Things Environments Based on Message Queuing Telemetry Transport"; Expert Systems 2024, 41, e13263. DOI:10.1111/exsy.13263.

[10] Choorod, P.; George, W.; Anil Fernando. "Classifying for Traffic Encrypted Payload using Machine Learning"; IEEE Access. 2024. DOI: 10.1109/ACCESS.2024.3356073

[11] Priya, S. S.; Sivaram, M.; Yuvaraj, D.; Jayanthiladevi, A. "Machine Learning Based DDoS Detection"; International