Journal of Soft Computing and Information Technology (JSCIT)
Babol Noshirvani University of Technology, Babol, Iran
Journal Homepage: jscit.nit.ac.ir
DOI: 10.22034/jscit.2025.535303.2141
Volume 14, Number 3, Autumn 2025, pp. 34-50
Received: 07/17/2025, Revised: 10/11/2025, Accepted: 10/14/2025

Detecting Distributed Denial of Service Attacks in Software Defined Networks using Machine Learning and Statistical Analysis

Mohammd Saeed Mumivand[1], Vahid Haghighatdoost[2*]
1- Department of Computer Engineering, Shahed University, Tehran, Iran.
2*- Department of Computer Engineering, Shahed University, Tehran, Iran.
[1] saeed.mumivand@shahed.ac.ir, [2*]haghighatdoost@shahed.ac.ir

Corresponding author's address: Vahid Haghighatdoost, Faculty of Computer Engineering and Technology, Shahed University, Tehran, Iran.

Abstract- Software-defined networking (SDN) is a novel approach to network architecture that provides flexibility, dynamism, and adaptability for network management. While accepting these advantages, the primary challenges of this type of network architecture are undoubtedly denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks. For this reason, preventing, detecting, and mitigating these attacks is crucial for the continued operation and efficiency of the network. Studies have shown that statistical approaches, such as entropy, or machine learning methods have attracted the attention of researchers in detecting these attacks. Statistical approaches have low computational overhead and high detection speed, but they have less detection ability than machine learning methods. Conversely, machine learning techniques typically require substantial computational resources and extended convergence times to accurately identify network attacks. In this study, a hybrid approach is proposed that integrates both statistical and machine learning-based methods for the detection of distributed denial-of-service attacks within software-defined networks. This integrated strategy aims to overcome the limitations of each method while capitalizing on their respective advantages. To minimize computational overhead and prevent resource saturation, the architecture employs two dedicated controllers, one responsible for traffic routing and the other for detecting anomalous behavior. Using this architecture, even if the routing controller is disrupted, the attack detection controller can identify the distributed denial-of-service attack. The hybrid model leverages entropy metrics in conjunction with the Packet-in rate over defined time intervals to identify suspicious traffic, which is subsequently subjected to in-depth analysis through machine learning algorithms. Using the proposed approach, the attack detection accuracy rate increases up to 99.47%. Furthermore, due to the examination of a large volume of traffic by statistical methods, which create less overhead, the proposed method shows a reduction in controller computational overhead by up to 10% compared to machine learning methods.

Keywords- Distributed denial of service attacks, software-defined networks, entropy, machine learning, dynamic threshold.

مجله علمی رایانش نرم و فناوری اطلاعات
دانشگاه صنعتی نوشیروانی بابل
صفحه مجله: jscit.nit.ac.ir
شناسه دیجیتال: 10.22034/jscit.2025.535303.2141
جلد ۱۴، شماره ۳، پاییز ۱۴۰۴ صفحه ۳۴-۵۰
دریافت: ۱۴۰۴/۰۴/۲۶، بازنگری: ۱۴۰۴/۰۷/۱۹، پذیرش: ۱۴۰۴/۰۷/۲۲

تشخیص حملات منع خدمت توزیع‌شده در شبکه‌های نرم‌افزارمحور با استفاده از رویکردهای ترکیب یادگیری ماشین و تحلیل آماری

محمدسعید مومیوند [۱]، وحید حقیقت‌دوست*[۲]
۱- دانشکده فنی و مهندسی، دانشگاه شاهد، تهران، ایران.
*۲- دانشکده فنی و مهندسی، دانشگاه شاهد، تهران، ایران.

[1] saeed.mumivand@shahed.ac.ir, [2*] haghighatdoost@shahed.ac.ir

* نشانی نویسنده مسئول: وحید حقیقت دوست، تهران، دانشگاه شاهد، دانشکده فنی و مهندسی.

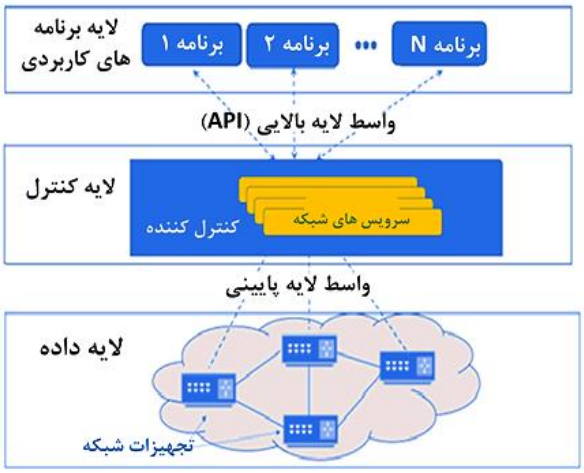
چکیده- شبکه نرم‌افزارمحور(SDN) رویکردی جدید در معماری شبکه می‌باشد که انعطاف پذیری، پویایی و قابلیت سازگاری را برای مدیریت شبکه فراهم می‌کند. ضمن پذیرش این مزایا، بی‌تردید از اصلی‌ترین چالش‌های این نوع معماری شبکه حملات منع خدمت(DoS) و منع خدمت توزیع‌شده(DDoS) هستند. به همین علت پیشگیری، تشخیص و کاهش این حملات امری حیاتی برای ادامه فعالیت و کارایی شبکه است. بررسی‌های صورت گرفته نشان داده‌اند که رویکردهای آماری، مانند آنتروپی یا روش‌های یادگیری ماشین برای تشخیص این حملات مورد توجه محققین قرار گرفته‌اند. رویکردهای آماری سربار محاسباتی کم و دارای سرعت بالا در تشخیص هستند، اما توانایی تشخیص کمتری نسبت به روش‌های یادگیری ماشین دارند. بر عکس، تکنیک‌های یادگیری ماشین معمولاً برای شناسایی دقیق حملات شبکه به منابع محاسباتی قابل توجه و زمان همگرایی طولانی نیاز دارند. در این مطالعه، رویکردی ترکیبی پیشنهاد شده است که روش‌های آماری و مبتنی بر یادگیری ماشین را برای تشخیص حملات منع خدمت توزیع‌شده در شبکه‌های نرم‌افزارمحور ادغام می‌کند. این استراتژی یکپارچه با هدف غلبه بر محدودیت‌های هر روش و در عین حال بهره‌برداری از مزایای مربوط به آنها طراحی شده است. برای به حداقل رساندن سربار محاسباتی و جلوگیری از اشباع منابع، این معماری از دو کنترل‌کننده که یکی مسئول مسیریابی ترافیک و دیگری برای تشخیص رفتار غیرعادی بوده استفاده می‌کند. با استفاده از این معماری حتی در صورت ایجاد اختلال برای کنترل‌کننده مسیریابی، کنترل‌کننده تشخیص حمله می‌تواند حمله منع خدمت توزیع‌شده را شناسایی کند. مدل ترکیبی از معیارهای آنتروپی همراه با نرخ ورود بسته در فواصل زمانی تعریف‌شده جهت شناسایی ترافیک مشکوک استفاده می‌کند که متعاقباً از طریق الگوریتم‌های یادگیری ماشین مورد تجزیه و تحلیل عمیق قرار می‌گیرد. با استفاده از رویکرد مطرح شده نرخ دقت تشخیص حمله تا ۹۹,۴۷٪ افزایش می‌یابد. همچنین به علت بررسی حجم زیادی از ترافیک توسط روش‌های آماری که سربار کمتری ایجاد می‌کنند، در روش پیشنهادی سربار محاسباتی کنترل‌کننده تا ۱۰٪ نسبت به روش یادگیری ماشین کاهش نشان می‌دهد.

واژه‌های کلیدی: حملات منع خدمت توزیع‌شده، شبکه‌های نرم‌افزارمحور، آنتروپی، یادگیری ماشین، آستانه پویا.

۱- مقدمه

یکی از ارکان اصلی امنیت در شبکه، بحث دسترسی‌پذیری می‌باشد که مهاجمان با اهداف و مقاصد مختلف نظیر ضربه زدن به اعتبار یک سازمان سعی در ایجاد اختلال و جلوگیری از ایجاد دسترسی به شبکه را دارند. در این میان حملات منع خدمت توزیع‌شده[۱] از طریق ارسال حجم عظیمی از ترافیک برای یک کاربر یا سرویس دهنده، پهنای باند شبکه را اشغال و منابع حافظه را از ارائه دهندگان خدمت با ایجاد ترافیک غیر واقعی به خود اختصاص می‌دهند و مانع از ارائه خدمات به درخواست‌های کاربران عادی می‌شوند[۱].

در شبکه‌های فعلی با ایجاد هر گونه تغییر حتی جزئی در شبکه نیاز به پیکربندی مجدد شبکه امری الزامی است و امکان بکارگیری الگوریتم و نرم‌افزارها را برای مدیریت شبکه ندارند. همچنین در صورت ایجاد سیاست‌های متناقض، دید جامعی جهت اتخاذ تصمیمات لازم به علت توزیع‌شدگی بخش‌های کنترلی در تجهیزات شبکه وجود ندارد. به علت وجود این دسته از محدودیت‌ها در شبکه‌های فعلی، معماری شبکه‌های نرم‌افزارمحور[۲] مورد توجه پژوهشگران حوزه شبکه قرار گرفته‌اند[۲, ۳].



شکل۱: معماری شبکه نرم‌افزارمحور

شبکه نرم‌افزارمحور نوع جدیدی از معماری شبکه است که از سوی مرکز تحقیقاتی دانشگاه استنفورد مطرح گردید. سطح کنترل، که به‌عنوان کنترل‌کننده شبکه نیز شناخته می‌شود، مسئول تصمیم‌گیری درباره نحوه ارسال ترافیک بر اساس سیاست‌های کلی شبکه است. از سوی دیگر، سطح داده تنها وظیفه انتقال ترافیک را بر عهده دارد و بر اساس دستور العمل‌های صادرشده توسط سطح کنترل عمل می‌کند[۴]. در معماری این نوع از شبکه‌ها پروتکل استاندارد بین سطح داده و سطح کنترل OpenFlow می‌باشد[۵].

در عصر حاضر، با گسترش فناوری‌هایی مانند IoT، 5G و ابر محاسباتی، شبکه‌های نرم‌افزارمحور با چالش‌های امنیتی جدی روبرو هستند. جداسازی لایه کنترل‌کننده از لایه داده اگرچه مدیریت شبکه را تسهیل می‌کند، اما SDN را آسیب‌پذیرتر به حملات می‌سازد، به ویژه حملات منع خدمت توزیع‌شده (DDoS) که کنترل‌کننده مرکزی را هدف قرار می‌دهند. این حملات می‌توانند کل شبکه را فلج کنند، زیرا کنترل‌کننده SDN به عنوان نقطه شکست واحدی تلقی می‌شود[۶]. به دلیل ویژگی‌های شبکهٔ SDN، زمانی که حمله DDoS رخ می‌دهد، جریان‌های ورودی به سوئیچ رشد قابل توجهی را نشان داده و تعداد زیادی از بسته‌های ورودی به کنترل‌کننده ارسال می‌شود. در این فرآیند میزبان قربانی، کنترل‌کننده و سوئیچ‌های متصل به آن تحت تاثیر این حمله قرار می‌گیرند[۷, ۸]. با وجود طراحی ساده حملات DDoS، در سطح حملات خطرناک دسته‌بندی شده که شناسایی و دفاع در مقابل آنها هزینه‌های زیادی را به شبکه تحمیل خواهدکرد. شیوه شناسایی حملات DDoS با کارایی بالا یکی از چالش‌های مهم و دشوار در شبکه‌های نرم‌افزارمحور است. با وجود تمام مزایا و قابلیت‌های این نوع معماری شبکه به دلیل این موضوع که قسمت کنترلی تمام مؤلفه‌های شبکه از جمله تجهیزاتی مانند سوئیچ‌ها در یک لایه کنترل‌کننده تجمیع شده امکان ایجاد یک نقطه‌شکست[۳] در شبکه محتمل است و هدف مطلوبی برای مهاجمان، جهت بکارگیری حملات منع خدمت توزیع‌شده با قدرت و اثرگذاری بیشتر قرار گرفته است[۹].

باوانی و همکارانش در[۱۰] چهار رویکر مبتنی بر آنتروپی، یادگیری ماشین[۴]، رویکردهای تجزیه و تحلیل ترافیک و نرخ اتصال را برای تشخیص حملات منع خدمت معرفی کردند. اما دو رویکرد مؤثر یعنی روش آماری مبتنی بر آنتروپی و یادگیری ماشین به جهت کارایی و تشخیص دقیق‌تر برای جلوگیری از حملات منع خدمت مورد توجه صاحب نظران حوزه شبکه قرار گرفته‌اند. به علت وجود اشکالات در روش‌های مطرح شده فعلی در SDN رویکردی که بتواند این حملات را با کارایی بالا تشخیص دهد حائز اهمیت است.

در این پژوهش بیان می‌شود که ضمن مقایسه تحقیقات صورت گرفته در گذشته روش پیشنهاد شده با استفاده از رویکردهای آماری و بررسی ویژگی‌های مؤثر از طریق الگوریتم یادگیری ماشین به صورت بلادرنگ توانایی تشخیص و شناسایی حملات را دارد. این روش ضمن سربار محاسباتی کمتر برای کنترل‌کننده و افزایش سرعت تشخیص به جهت انتخاب ویژگی‌های محدود از قابلیت الگوریتم‌های یادگیری ماشین جهت تشخیص حمله

استفاده کرده که توانایی تشخیص با عملکرد بهتر را تضمین می‌کند.

۲- پیشینه پژوهش

پژوهش‌های زیادی در رابطه با روش‌های تشخیص حملات منع خدمت توزیع‌شده در شبکه نرم‌افزارمحور صورت گرفته است در این بخش چند مورد از آن‌ها شرح داده می‌شوند:

عمر و همکاران[۱۱] با استفاده از روش آماری مبتنی بر آنتروپی ترافیک شبکه که دارای حدآستانه ثابت می‌باشد حمله را تشخیص می‌دهند. بسته‌های ورودی باید بر اساس زمان سپری شده یا تعداد آنها تقسیم شوند. هر پنجره حاوی یک جدول دو ستونی است. ستون اول آدرس IP مقصد جریان و ستون دوم میزان ترافیک شبکه آدرس مدنظر را نشان می‌دهد. جریان‌های ورودی برای یافتن قانون مرتبط بررسی می‌شوند. در این مقاله از آمارهای موجود در کنترل‌کننده استفاده شده و پنجره زمانی روی مقدار ۵۰ بسته تنظیم شده‌است. برای تشخیص حمله در کنترل‌کننده، IPهای مقصد از بسته‌های ورودی تحلیل و بررسی می‌شود. بررسی‌ها بیانگر این موضوع بود که این روش به صورت مؤثری توانایی تشخیص حملات را دارد، که در صورتی که مقدار محاسبه شده در پنج بار اندازه‌گیری متوالی بیشتر از آستانه باشد، به عنوان حمله دسته بندی می‌شود و به وسیله پورت‌ها و شناسه سوئیچ حمله را خنثی و مسدود می‌کند. روش پیشنهاد شده، این ضعف را نشان می‌دهد که توسط مهاجم از طریق ارسال ترافیک جعلی برای سایر مقاصد و جلوگیری از کاهش ناگهانی حدآستانه حمله را اجرا کند و دقت تشخیص حمله تا حد زیادی کاهش دهد.

گالیانو و همکاران[۱۲] از رویکرد آنتروپی به جهت شناسایی حملات منع سرویس توزیع‌شده استفاده می‌کند که دارای یک الگوی توزیع‌شده به جهت تشخیص حملات می‌باشد. در روش ارائه‌شده برای هر مؤلفهٔ لبه یعنی راه‌گزین‌های لایه زیرساخت در شبکه‌های نرم‌افزار محور، یک واحد تشخیص حمله منع خدمت توزیع‌شده در نظر گرفته‌شده است. بدین ترتیب وظیفه جمع‌آوری جریان و محاسبه آنتروپی که در روش‌های ارائه‌شده قبلی بر عهده کنترل‌کننده بود به این واحدها محول می‌شود و سربار محاسباتی کنترل‌کننده شبکه کاهش می‌یابد. در هر واحد بر اساس آدرس آی‌پی مقصد، آنتروپی محاسبه شده و در صورتی که آنتروپی هرکدام از راه‌گزین‌های شبکه بیش از حدآستانه تغییر کرده باشد، وقوع حمله احتمالی به کنترل‌کننده گزارش داده می‌شود. این روش با هدف کاهش سربار محاسباتی در کنترل‌کننده ایجاد شده است و تنها حملاتی را می‌تواند تشخیص دهد که آنتروپی ترافیک عبوری از حدآستانه مشخص‌شده در هر راه‌گزین فراتر رود. با توجه به ویژگی توزیع‌شدگی در حملات منع خدمت، لزوماً ترافیک حمله عبوری از یک راه‌گزین مشخص در شبکه عبور نمی‌کند و این روش بیشتر برای حملات منع خدمتی که به صورت توزیع‌شده نباشند توانایی تشخیص با سرعت بیشتر و سربار کمتر برای کنترل‌کننده را به همراه دارد.

همتی و همکاران در[۱۳] از حدآستانه پویا به جای حدآستانه ثابت استفاده شده است که می‌تواند به افزایش دقت تشخیص در شرایط و سناریوهای مختلف کمک کند. در الگوریتم پیشنهاد شده برای جلوگیری از ایجاد سربار زیاد برای کنترل‌کنندهٔ SDN از سه تابع سبک برای تشخیص استفاده کرده است که تابع اول آدرس‌های IP مقصد را دسته بندی می‌کند، تابع دوم آنتروپی را محاسبه کرده و جهت بررسی وجود یا عدم وجود حمله آن را با حدآستانه تطبیق می‌دهد و تابع سوم از طریق میانگین آنتروپی ترافیک حمله و عادی و همچنین انحراف استاندار آنتروپی ترافیک، حدآستانه را محاسبه می‌کند. این روش نیز تنها بر اساس آدرس IP مقصد احتمال وقوع حمله را بررسی کرده و در صورتی که مهاجم ترافیک حمله را بین مقاصد مختلف توزیع کند، دقت روش پیشنهاد شده کاهش می‌یابد.

در مقاله[۱۴] ریبیرو و همکاران یک تکنیک تشخیص نفوذ مبتنی‌بر ناهنجاری را برای محیط SDN ارائه می‌کنند که از الگوریتم یادگیری ماشین برای تشخیص استفاده می‌شود که با جریان داده پیوسته و مجموعه‌ای از دستورالعمل‌ها برای غلبه بر مشکلات ارائه شده در تشخیص ناهنجاری‌ها عمل می‌کند. در این پژوهش برای این که بتوان حملات را شناسایی نمود، باید مبنایی از امضای حملات موجود باشد. این دسته از پیشنهادات علیرغم سرعتی که در شناسایی حملات خاص به دلیل آگاهی از امضای حملات دارد، تنها می‌تواند حملاتی را کشف کند که با امضاهای موجود قبلی سازگار باشند و مدل یادگیری قبلا نمونه‌هایی از حمله‌ی صورت گرفته را مشاهده کرده باشد.

الزهرانی و همکاران در[۱۵] توانایی تشخیص نفوذ در شبکه را با بکارگیری الگوریتم‌های مختلف یادگیری ماشین بر روی SDN افزایش می‌دهند. الگوریتم‌های یادگیری ماشین مبتنی بر درخت (XGBoost، جنگل تصادفی و درخت تصمیم) برای افزایش نظارت و عملکرد شاخص دقت پیاده‌سازی شدند. روش پیشنهادی بر روی مجموعه داده‌ای به نام NSL-KDD برای انجام دو کار تشخیص وجود یا عدم وجود حمله و طبقه‌بندی نوع حمله، آموزش داده می‌شود. آموزش مدل تنها با پنج ویژگی از ۴۱ ویژگی موجود

صورت می‌گیرد و نوع حمله مورد تجزیه و تحلیل قرار می‌گیرد تا مشاهده شود به کدام دسته تعلق دارد. نتایج بررسی معیارهای عملکردی نشان‌دهنده برتری الگوریتم XGBoost نسبت به سایرین می‌باشد. تعیین نوع دقیق حمله می‌تواند موجب اقدامات کاهشی مناسب شود اما روش‌های یادگیری ماشین در کنترل‌کننده سربار محاسباتی بالایی ایجاد خواهندکرد.

خشاب و همکاران[۱۶] متفاوت از سایر رویکردهای موجود که تنها از ویژگی‌های جریان بومی برای تشخیص حمله استفاده می‌کنند، این مدل ویژگی‌های بومی را گسترش می‌دهد. این ویژگی‌های توسعه یافته علاوه بر ویژگی‌های بومی (یعنی تعداد بسته‌ها، بایت‌ها و مدت زمان ارسال آنها) توسط یک ماژول تشخیص برای طبقه‌بندی هر جریان به عنوان عادی یا غیرعادی با استفاده از الگوریتم یادگیری ماشین استفاده می‌شوند. معماری مکانیسم پیشنهادی آنها از چهار ماژول اصلی جمع کننده جریان، توسعه دهنده ویژگی، تشخیص ناهنجاری و کاهش ناهنجاری تشکیل شده است. نتایج آزمایش نشان داد که RF[۵] مناسب‌ترین مدل طبقه‌بندی برای این سناریو در شبکه است.

در مقاله[۱۷] دیپا و همکاران مجموعه‌ای از تکنیک‌ها را با اتخاذ الگوریتم‌های مختلف یادگیری ماشین برای تشخیص رفتار غیرعادی ترافیک شبکه پیشنهاد می‌کنند که روش یادگیری به صورت گروهی، دقت بهتری را نسبت به الگوریتم‌هایی که تنها از یک مدل یادگیری استفاده می‌کردند ارائه می‌دهد. در واقع ترکیب تکنیک‌های یادگیری ماشین به عنوان یک روش یادگیری گروهی برای شناسایی بسته‌های مخرب و نرمال استفاده می‌شود و بررسی‌های انجام شده نشان می‌دهد که ترکیب الگوریتم‌های SVM-SOM در مقایسه با مدل یادگیری ماشین ساده و همچنین سایر مدل‌های ترکیبی به دقت، نرخ تشخیص و نرخ هشدار مثبت کاذب بهتری دست یافت. به طور مشابه در کارهای دیگر هم با استفاده از چندین روش یادگیری به نتایج بهتری رسیدند. اما باید به این نکته اشاره کرد که ترکیب رویکردهای یادگیری ماشین می‌تواند باعث سربار محاسباتی بسیار بالا و کاهش سرعت تشخیص حمله شود.

در مقاله[۱۸] لیو و همکاران با توجه به قابلیت‌های شبکه نرم‌افزارمحور روشی برای تشخیص حملات منع خدمت توزیع‌شده از ترکیب هر دو رویکرد آماری مبتنی بر آنتروپی و یادگیری ماشین بهره بردند. در این روش ضمن افزایش دقت، کاهش بار محاسباتی کنترل‌کننده را نیز به دنبال دارد. روش مبتنی بر آنتروپی ساده بوده و منابع محاسباتی کمتری از کنترل‌کننده را اشغال می‌کند اما دقت بالایی ندارند از طرف دیگر الگوریتم‌های یادگیری ماشین در صورت استفاده از مجموعه آموزشی متناسب دقت بسیارخوبی برای تشخیص دارند اما منابع محاسباتی زیادی را به خود اختصاص می‌دهند. در روش پیشنهادی ترافیک شبکه محلی در سوئیچ شناسایی می‌شود و در دو دسته نرمال و مخرب طبقه‌بندی می‌شود. از این طریق کنترل‌کننده اطلاعات جدول سوئیچ مشکوک را جمع آوری می‌کند و شش ویژگی از آن را استخراج کرده که الگوریتم یادگیری ماشین احتمال وقوع حمله را بررسی می‌کند. در این روش شبکه زیر سوئیچ ممکن است قربانی شود و تعیین حدآستانه دقیق جهت تشخیص چالش برانگیز می‌باشد همین امر می‌تواند دقت مدل تشخیصی مبتنی بر آنتروپی را پایین آورده و سربار محاسباتی در کنترل‌کننده را افزایش دهد.

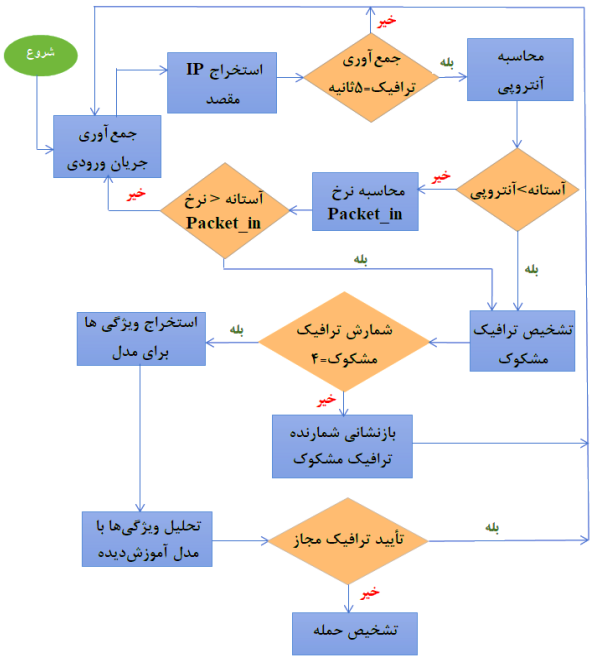
در پژوهش‌های گذشته که از رویکردهای آماری به تنهایی بهره‌برده‌اند، [۱۱, ۱۲] ضمن پذیرش مزیت سربار محاسباتی کمتر، مسئله تشخیص حمله با دقت بالا چالش برانگیز می‌باشد. حتی در [۱۳] نیز با وجود پویا سازی حدآستانه برای روش آنتروپی دقت این مدل نسبت به روش‌های مدل یادگیری کمتر است. همچنین ضمن پذیرش دقت بالا در تشخیص حملات با استفاده از روش‌های یادگیری ماشین سربار، محاسباتی زیادی برای کنترل‌کننده ایجاد می‌شود[۱۴-۱۷]. با این وجود در[۱۸] نیز با توجه به این که از روش آنتروپی به صورت محلی در راه‌گزین‌های شبکه استفاده می‌شود امکان ایجاد مثبت کاذب و منفی کاذب بالا می‌رود و حتی در صورت بکارگیری این روش در کنترل‌کننده مرکزی نیز، توسط مهاجم از طریق ارسال یکنواخت‌تر ترافیک مخرب قابلیت نفوذ دارد. این شیوه حمله باعث بی اثر شدن روش آنتروپی می‌شود. در روش پیشنهادی با استفاده از پتانسیل روش یادگیری ماشین در کنار استفاده از رویکردهای آماری در ترافیک شبکه، دقت تشخیص بیشتر شده و از هزینه زیاد جهت زیرساخت سامانه جلوگیری می‌شود.

۳- روش پیشنهادی

در این بخش الگوریتم پیشنهادی از طریق تقویت مکانیسم موجود، حملات DDoS را در شبکه‌های نرم‌افزارمحور با استفاده از تجزیه و تحلیل آماری و روش یادگیری ماشین تشخیص می‌دهد.

در معماری پژوهش‌های بررسی‌شده از یک کنترل‌کننده جهت مسیریابی و تشخیص حمله استفاده نموده‌اند که چالش‌های محاسباتی متعددی را در شبکه به همراه دارد. نخست آن‌که در شرایط حمله مهاجم با تولید عمدی حجم زیادی از بسته‌های که فاقد قانون مشابه‌ای در جدول جریان سوئیچ‌ها هستند باعث

افزایش نمایی درخواست‌های Packet-in به کنترل‌کننده شبکه می‌شوند. این عمل علاوه بر اشباع پهنای باند میان تجهیزات و کنترل‌کننده، میزان زیادی از منابع سیستمی را نیز مصرف می‌کند. همچنین ارسال مداوم آمار جریان و فرآیند تشخیص حمله مبتنی‌بر آمار و یادگیری ماشین به منابع پردازشی گسترده نیاز دارد. این تمرکز وظایف بر روی کنترل‌کننده واحد، موجب افزایش بار پردازشی، تاخیر در پاسخ‌گویی و کاهش کارایی خواهد شد.



شکل۲: فلوچارت روش پیشنهادی

برای رفع این مشکل استفاده از دو کنترل‌کننده برای توپولوژی شبکه پیشنهاد می‌شود که کنترل‌کننده اصلی عمل مسیریابی و مدیریت ترافیک را انجام می‌دهد و کنترل‌کننده دیگر جهت تشخیص حمله استفاده می‌شود. هدف از این پژوهش بکارگیری اصولی می‌باشد که از صرف انرژی زیاد پردازنده در کنترل‌کننده اصلی ممانعت به عمل آید. در این صورت کنترل‌کننده توانایی پاسخ‌گویی به درخواست‌ها و نصب قوانین را با کاهش تأخیر و سربار محاسباتی خواهد داشت. همچنین از طریق پرکردن خلأ تکنیک‌های آماری موجود با بکارگیری ویژگی‌های محدود این توانایی را دارد که همراه با الگوریتم یادگیری ماشین تشخیص کارآمد حمله را در شبکه داشته باشد. این کار باعث بهبود معیارهای عملکردی برای تشخیص حملات DDoS می‌شود. در ادامه به تفصیل جزئیات رویکرد پیشنهادی شرح داده می‌شود.

۳-۱- محاسبه آنتروپی

آنتروپی در تئوری اطلاعات، یک روش وزن‌دهی براساس معیار فراوانی از میزان اطلاعات یا میزان تصادفی بودن یک متغیر است. هراندازه که تصادفی بودن متغیرها بیشتر باشد، مقدار آنتروپی نیز بیشتر خواهدبود. بالعکس، در واقع هر چه قطعیت نسبت به متغیرهای اطلاعاتی بالاتر باشد، مقدار آنتروپی کمتر است. در یک حمله DDoS توقع این را داریم که ترافیک ارسالی از آی‌پی‌های منبع جعلی مختلف که توزیع‌شده است به یک آدرس آی‌پی مقصد مورد نظر تمرکز پیدا کند، که از طریق آنتروپی شانون[۶] قابل بررسی می‌باشد. دو مؤلفه مهم برای شناسایی با دقت بالا در حملات DDoS اندازه پنجره و حدآستانه می‌باشند که با بررسی تعدادی بسته یا یک دوره زمانی مطابق با ویژگی‌های جدول جریان برای تعیین حدآستانه به عنوان معیاری جهت تشخیص حمله استفاده خواهد شد.

روش‌هایی که بر پایه آنتروپی به شناسایی حملات منع خدمت توزیع‌شده می‌پردازند برای محاسبه از ویژگی‌های جدول جریان استفاده می‌کنند که در روش پیشنهادی، آنتروپی برای آدرس آی‌پی مقصد مطابق با آنتروپی شانون و رنی[۷] محاسبه می‌شود. رابطه (۲) نحوه محاسبه آنتروپی شانون را برای توزیع آدرس‌های IP مقصد در شبکه نمایش می‌دهد.

در رابطه (۱) m_i تعداد بسته‌های داده در ورودی جریان است که آدرسIP مقصد آن‌ها IP_i است. M مجموع کل تعداد بسته‌های داده در پنجره زمانی می‌باشد، این پارامتر حجم کلی داده را نشان می‌دهد. X متغیر گسسته‌ای که توزیع آن بررسی می‌شود (مانند آدرس‌های IP). x_i مقدار iام متغیر گسسته X است که اشاره به یک آدرس IP منحصربه فرد دارد. $p(x_i)$ سهم نسبی هر آدرسIP مقصد در کل توزیع کل بسته‌ها به مقاصد مختلف را نشان می‌دهد و همیشه بین ۰ و ۱ است و n تعداد آدرس‌های IP منحصربه‌فرد در یک دوره زمانی است.

$$p(x_i) = \frac{m_i}{M} \qquad (۱)$$

$$H(X) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \qquad (۲)$$

همچنین در صورت محاسبه آنتروپی شبکه به وسیله، آنتروپی تعمیم یافته از رابطه (۳) قابل محاسبه خواهد بود. α پارامتر مرتبه آنتروپی است، که مقادیر مختلفی می‌گیرد و رفتار آنتروپی را

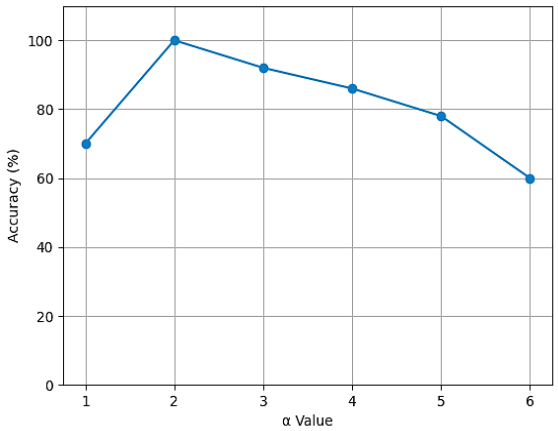
تنظیم می‌نماید.

$$H_\alpha(X) = \frac{1}{1-\alpha}\log_2(\sum_{i=1}^{n} p(x_i)^\alpha) \quad (۳)$$

این رابطه تعمیمی از آنتروپی شانون به شمار می‌رود که با معرفی پارامتر α توانایی بیشتری در ایجاد اختلاف بین ترافیک نرمال و حمله را فراهم می‌کند. زیرا رویدادهایی که احتمال بالاتری دارند (مانند زمانی که تمرکز ترافیک رخ می‌دهد) تأثیر بیشتری بر میزان آنتروپی خواهند داشت. این موضوع زمانی ملموس‌تر می‌شود که $\alpha > 1$ باشد.

آنتروپی تعمیم‌یافته به دلیل ایجاد تأثیر بیشتر رویدادهای با احتمال بالا، حساسیت بیشتری در تشخیص حملات دارد که همین امر باعث تشخیص با دقت بالاتر حتی در درصدهای پایین حمله می‌شود. انتخاب آستانه برای تمایز بین ترافیک نرمال و حمله ساده‌تر است، زیرا فاصله آنتروپی بین این دو حالت با افزایش α بیشتر می‌شود.

در پژوهش انجام شده مشاهده شد که در صورتی که مقدار α به ازای مقادیر بزرگتر از دو ($\alpha > 2$) قرار داده شود، دقت محاسبات تحت تأثیر خواهدگرفت و در صورت ایجاد تغییرات کوچک و یا نویزهای جزئی ایجاد شده ترافیک به عنوان حمله تشخیص داده شدند و علاوه بر این موضوع به ازای مقادیر بزرگتر α سربار محاسباتی نیز افزایش می‌یابد. بر این اساس، به‌منظور بهینه‌سازی تعادل بین دقت و کارایی، مقدار $\alpha = 2$ به عنوان پارامتر بهینه انتخاب گردید.



شکل ۳: میزان دقت آنتروپی رنی به ازای نرخ حمله ۱۰٪ و مقادیر متفاوت α

در تمرکز ترافیک برای حملات مثل DDoS که ترافیک روی یک یا چند آدرس IP متمرکز می‌شود، از α منفی استفاده نمی‌کنیم، زیرا این مقادیر به جای توجه به بخش‌های با ترافیک زیاد، به قسمت‌های پراکنده و کم‌حجم وزن می‌دهند و تمرکز حمله را نادیده می‌گیرند؛ در نتیجه، تشخیص دقیق حمله با این تنظیمات دشوار می‌شود و دقت سیستم کاهش می‌یابد. میزان دقت آنتروپی رنی با استفاده از αهای متفاوت و نرخ حمله ۱۰٪ بررسی شد که در صورت انتخاب $\alpha = 2$ امکان تمایز ترافیک نرمال از حمله به طور کامل ایجاد می‌شود و در صورت افزایش نرخ حمله میزان این تمایز به وضوح آشکار می‌شود.

جهت تفسیرپذیری بهتر و مقایسه نتایج آنتروپی‌های بکار گرفته‌شده می‌توان آنها را با استفاده از رابطه (۴) نرمال سازی کرد. هدف از نرمال‌سازی حذف تأثیر مقیاس‌ها یا دامنه‌های مختلف داده‌ها بوده تا بتوان داده‌ها را در یک چارچوب یکسان مقایسه کرد. در زمینه آنتروپی، نرمال‌سازی باعث می‌شود مقادیر آنتروپی مستقل از تعداد عناصر (مثلاً تعداد آدرس‌های IP مقصد) شوند و در محدوده‌ای مشخص (بین ۰ و ۱) قرار گیرند. $\log_2 n$ حداکثر مقدار ممکن آنتروپی در توزیع نرمال می‌باشد.

$$H_N(x) = \frac{H_{(x)}}{\log_2 n} \quad (۴)$$

با توجه به این موضوع که امکان ایجاد اختلالات در شبکه یا از کار افتادن برخی از دستگاه‌ها وجود دارد که باعث ایجاد بی‌نظمی موقتی در ترافیک مبادله‌شده می‌شود. براساس این اصل که حمله‌کننده به صورت مستمر و ضمن چندین جریان حمله را ترتیب خواهدداد، از این موضوع استفاده شده و محدودیتی برای بررسی تعداد پنجره‌ها اعمال شده است. که مطابق با آزمایشات مختلف با مقادیر متفاوت پنجره زمانی ۵ ثانیه‌ای و تکرار ۴ مرتبه‌ای برای ترافیک مشکوک در نظر گرفته می‌شود. این کار باعث شده که در صورت کاهش موقت آنتروپی از تشخیص اشتباه جلوگیری شود و نرخ مثبت کاذب کاهش یابد.

۳-۲- محاسبه Packet-in در بازه زمانی معین

در شبکه نرم‌افزارمحور، پیام‌های Packet-in نقش اساسی در تعامل بین لایه داده یعنی سوئیچ‌های OpenFlow و لایه کنترل ایفا می‌کنند. این پیام‌ها زمانی تولید خواهندشد که یک بسته ورودی در سوئیچ با هیچ ورودی جریانی در جدول جریان انطباق نداشته باشد. در این حالت، بسته به صورت یک پیام Packet-in به کنترل‌کننده ارسال می‌شود تا تصمیم‌گیری لازم مثل ایجاد ورودی جریان جدید یا دور انداختن بسته اتخاذ شود. در حملات DDoS مهاجم با ارسال حجم زیادی از بسته‌ها به مقصدی خاص یعنی

حمله متمرکز یا چندین مقصد، منابع شبکه را مصرف کرده و باعث ایجاد اختلال در خدمات و عملکرد سیستم یا سازمان مورد نظر می‌شوند. روش‌های سنتی تشخیص DDoS، مانند آنتروپی، در سناریوهای حمله متمرکز کارآمد هستند، اما در حملات توزیع‌شده به چندین مقصد، به دلیل عدم کاهش کافی آنتروپی، ممکن است ناکارآمد باشند. از سوی دیگر، نرخ پیام‌های Packet-in که در اثر عدم تطابق بسته‌ها با جداول جریان سوئیچ به کنترل‌کننده ارسال می‌شوند به‌شدت افزایش پیدا می‌کند این رفتار از طریق تعداد شبکه‌ای از بات‌ها برای ارسال ترافیک با مقدار بسیار زیاد به سمت مقاصد تعیین شده می‌باشد، که می‌تواند نشان‌دهندهٔ ناهنجاری شبکه در سناریوهای حمله توزیع‌شده باشد. در این حالت علاوه بر اشباع جداول جریان سوئیچ‌ها به علت ایجاد قوانین جریان جدید، موجب افزایش سربار کنترل‌کننده به علت مسیریابی برای بسته‌ها می‌شود. بر همین اساس، روش پیشنهادی ترکیبی از آنتروپی و نرخ Packet-in را به عنوان روش‌های پیش‌تشخیص به‌کار گرفته است تا ترافیک مشکوک را شناسایی کرده و سپس برای تحلیل دقیق‌تر به یک مدل یادگیری ماشین ارجاع دهد. این رویکرد نه‌تنها دقت تشخیص را بهبود می‌بخشد، بلکه با کاهش بار محاسباتی کنترل‌کننده، کارایی سیستم را افزایش می‌دهد.

برای محاسبه تعداد کل پیام‌های Packet-in در یک پنجره زمانی $N_{Packet-in}$ مجموع کل تعداد بسته‌های Packet-in تولید شده که در بازه زمانی تعیین شده از سوئیچ‌های OpenFlow به کنترل‌کننده ارسال شده‌اند. $C_{Packet-in}(s_i)$ تعداد پیام‌های تولید شده در بازه زمانی مقرر از مبداهای متفاوت و s_i آدرس IP مبدأ منحصربه‌فرد i–ام در بازه مورد نظر و n تعداد کل مبدأهای منحصربه‌فر می‌باشد.

$$N_{Packet-in} = \sum_{i=1}^{n} C_{Packet-in}(s_i) \qquad (۵)$$

۳-۲-۱- پویاسازی حدآستانه Packet-in

بکارگیری آستانه‌های ثابت برای تشخیص حمله می‌تواند به دلیل تغییرات طبیعی در الگوهای ترافیک، مانند افزایش ترافیک مشروع در ساعات اوج[۸] باعث ایجاد مثبت‌های کاذب (تشخیص نادرست ترافیک عادی به‌عنوان حمله) یا منفی‌های کاذب (عدم تشخیص حملات واقعی) شود. پویاسازی آستانه امکان سازگاری با این تغییرات را فراهم می‌کند.

روش نابرابری چبیشف[۹] یک روش آماری قوی برای تنظیم آستانه پویا است. از این رویکرد می‌توان برای تنظیم آستانه پویا جهت شناسایی حملات DDoS در شبکه استفاده کرد که از میانگین (μ) حسابی تعداد پیام‌های Packet-in مرتبط با هر IP مبدأ منحصربه‌فرد و انحراف معیارِ (σ) انحراف معیار تعداد پیام‌های Packet-in مرتبط با هر IP مبدأ، که پراکندگی داده‌ها حول میانگین در شرایط عادی را نشان می‌دهد بهره می‌برد. این روش می‌تواند تغییرات غیرعادی مانند افزایش ناگهانی نرخ Packet-in ناشی از حملات DDoS را شناسایی کند. آستانه پویا از رابطه (۸) محاسبه می‌شود، که در آن k ضریبی برای تنظیم حساسیت است بر اساس تجربه یا آزمایش تنظیم می‌شود که در این پژوهش جهت حساسیت بیشتر به ترافیک مشکوک $k = 3$ انتخاب می‌شود و از حدآستانه پایین جهت از دست ندادن ترافیک مشکوک استفاده شده‌است. این روش به دلیل توانایی در سازگاری با تغییرات طبیعی ترافیک و سادگی محاسباتی، برای محیط‌های SDN که نیاز به نظارت بلادرنگ دارند، بسیار مناسب است.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} C_{Packet-in}(s_i) \qquad (۶)$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(C_{Packet-in}(s_i) - \mu\right)^2} \qquad (۷)$$

$$T = \mu - k\sigma \qquad (۸)$$

۳-۳- الگوریتم یادگیری ماشین

در این پژوهش، برای تشخیص حملات منع خدمت توزیع‌شده در شبکه‌های نرم‌افزارمحور، از الگوریتم ماشین بردار پشتیبان با کرنل غیرخطی RBF استفاده شده است. SVM یک روش یادگیری ماشین نظارت‌شده است که با یافتن ابرصفحه‌ای با حداکثر حاشیه جداسازی، داده‌های ترافیک شبکه را به دو کلاس ترافیک عادی و ترافیک تحت حمله طبقه‌بندی می‌کند. داده‌های مورد استفاده از جدول جریان سوئیچ‌های OpenFlow استخراج شده‌اند و شامل پنج ویژگی که در ادامه بیان می‌شوند هستند. این ویژگی‌ها در بازه‌های زمانی تعیین‌شده جمع‌آوری شده‌اند. با توجه به ماهیت غیرخطی توزیع این ویژگی‌ها در حملات DDoS، مانند حملات ICMP Flood، UDP Flood و TCP SYN Flood، کرنل RBF به دلیل توانایی در نگاشت داده‌ها به فضای ویژگی با ابعاد بالاتر و جداسازی غیرخطی، به‌عنوان گزینه‌ای مناسب انتخاب می‌شود.

کرنل RBF شباهت بین نقاط داده را بر اساس فاصله اقلیدسی محاسبه می‌کند و امکان جداسازی کلاس‌های غیرخطی را فراهم می‌سازد. این کار را انجام می‌دهد تا داده‌های پیچیده و غیرخطی،

مانند الگوهای ترافیک شبکه در حملات DDoS، را به فضایی با ابعاد بالاتر نگاشت دهد که در آن جداسازی خطی ممکن شود، و همچنین از محاسبات مستقیم و پرهزینه مختصات در فضای بالاتر جلوگیری کند.

فرمول کرنل از طریق رابطه زیر تعریف می‌شود:

$$K(x_i - x_j) = \exp(-\gamma . ||x_i - x||^2) \qquad (۹)$$

در رابطه (۹) $K(x_i - x)$ مقدار کرنل بین وکتور پشتیبان x_i که یکی از نقاط داده نزدیک به مرز جداسازی و نقطه جدید x را محاسبه می‌کند که در بازه $[1 , 0]$ قرار دارد، exp تابع نمایی، که کاهش نمایی فاصله را مدل می‌کند، γ پارامتر تنظیم‌کننده عرض تابع گاوسی، که بر حساسیت مدل به تفاوت‌های محلی تأثیر می‌گذارد هر چه مقدار آن کوچک‌تر باشد (مثلا ۰,۰۰۱) مدل را به الگوهای کلی‌تر متمایل کرده، در صورتی‌که مقادیر بزرگ (مثلا۱۰) به نقاط نزدیک حساس‌تر هستند. مقدار بهینه با استفاده از روش Grid Search مقدار ۰,۱ انتخاب می‌شود و $||x_i - x||$ فاصله اقلیدسی بین بردار x_i و x است.

نُرم مربعی فاصله اقلیدسی بین x_i و x که با فرمول زیر محاسبه می‌شود.

$$||x_i - x||^2 = \sum_{k=1}^{d} (x_{i,k} - x_k)^2 \qquad (۱۰)$$

در این رابطه $x_{i,k}$ مقدار مؤلفه k–ام بردار x_i و x_k مقدار مؤلفه k–ام بردار x می‌باشد و d نیز تعداد ویژگی‌ها که برابر پنج می‌باشد را بیان می‌کند.

تابع تصمیم‌گیری SVM برای طبقه‌بندی یک نقطه جدید x با استفاده از رابطه این تصمیم‌گیری را انجام می‌دهد.

$$f(x) = sign(\sum_{i=1}^{N} \alpha_i . y_i . K(x_i - x) + b) \qquad (۱۱)$$

در این رابطه $f(x)$ خروجی تابع تصمیم‌گیری، با استفاده از تابع sign به کلاس حمله یا عادی نگاشت می‌شود. N نماد تعداد بردارهای پشتیبان می‌باشد، α_i ضرایب لاگرانژ برای وکتور پشتیبان i، که از حل مسئله بهینه‌سازی به دست می‌آید و نشان‌دهنده اهمیت هر وکتور است، y_i برچسب کلاس برای نمونه i، می‌باشد، b بایاس، که موقعیت ابرصفحه را تنظیم می‌کند.

در الگوریتم SVM مسئله بهینه‌سازی دوگان برای یافتن ضرایب لاگرانژ (α_i) حل می‌شود. این مسئله به‌ویژه در استفاده از کرنل‌ها اهمیت دارد، زیرا به محاسبات در فضای ابعادی بالاتر بدون نگاشت صریح داده‌ها اجازه می‌دهد.

$$\max_{\alpha}(\sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i . \alpha_j . y_i . y_j . K(x_i - x_j)) \qquad (۱۲)$$

با محدودیت‌های

$$\sum_{i=1}^{N} \alpha_i . y_i = 0$$

$$0 \le \alpha_i \le C, \qquad \forall_i = 1,2, \dots , N$$

در این رابطه C پارامتر منظم‌سازی[۱۰] که تعادل بین حداکثر کردن حاشیه جداسازی و کاهش خطاهای طبقه‌بندی را کنترل می‌کند. مقادیر کوچک C حاشیه بزرگ‌تری ایجاد می‌کنند و ممکن است برخی خطاها را نادیده بگیرند، در حالی که مقادیر بزرگ‌تر مدل را به داده‌های آموزشی بیش از حد نزدیک می‌کنند. برای پیدا کردن مقدار بهینه، از روش Grid Search استفاده شده است تا پارامتر C بهینه شود. در این روش، مقادیر مختلف C با اعتبارسنجی متقاطع ۵–fold آزمایش شدند بهترین عملکرد مربوط به، $C = 1$ انتخاب شده است. نشان‌دهنده هدف بهینه‌سازی است که تابع هدف باید بیشینه شود. عبارت $\max_{\alpha}$ در فرمول مسئله بهینه‌سازی دوگان SVM، نشان‌دهنده هدف بهینه‌سازی است که هدف آن بیشینه کردن تابع هدف نسبت به متغیرهای$[\alpha_1 , \alpha_2 , \dots , \alpha_N]$ هستند، در اینجا، α یک بردار شامل ضرایب لاگرانژ برای هر نمونه آموزشی i از ۱ تا N است. این بیشینه‌سازی به منظور یافتن مقادیر بهینه α_i انجام می‌شود که تابع هدف را به حداکثر برساند. این فرآیند با استفاده از روش‌های بهینه‌سازی ریاضی حل می‌شود و نتیجه، مجموعه‌ای از α_i است که ابرصفحه جداسازی را تعریف می‌کند.

الگوریتم ماشین بردار پشتیبان[۱۱] از روش‌های یادگیری نظارتی در حوزه طبقه‌بندی و رگرسیون می‌باشد که به علت قابلیت آن در ایجاد مرزهای تصمیم‌گیری بهینه، به‌طور گسترده در مسائل امنیتی شبکه، از جمله تشخیص حملات DDoS، مورد استفاده قرار می‌گیرد. SVM با هدف ایجاد حداکثر حاشیه[۱۲] بین کلاس‌های مختلف، مانند ترافیک نرمال و حمله، کاربرد دارد و این حاشیه به‌عنوان فاصله میان نزدیک‌ترین نمونه‌های هرکلاس و مرز تصمیم‌گیری معرفی خواهندشد. این قابلیت، SVM را به مدلی با تعمیم‌پذیری بالا تبدیل می‌کند که حتی با وجود داده‌های نویزی، عملکرد بهتری ارائه می‌دهد. فرآیند آموزش SVM شامل یادگیری الگوهای داده‌های آموزشی با استفاده از ویژگی‌هایی که در ادامه تعریف می‌شوند خواهدبود. پس از آموزش، مدل با داده‌های

آزمایشی مورد ارزیابی قرار می‌گیرند تا معیارهایی مانند دقت و سایر شاخص‌های عملکردی سنجیده شوند. در کاربردهای شبکه‌ای، SVM با تحلیل ویژگی‌های ترافیک، حملات DDoS را با سرعت و دقت بالا شناسایی کرده و امکان واکنش سریع به تهدیدات را فراهم می‌آورد. کرنل تابع پایه شعاعی به‌عنوان یکی از مؤثرترین کرنل‌ها در SVM، نقش کلیدی در بهبود عملکرد مدل برای داده‌های غیرخطی، مانند ترافیک شبکه دارد. این کرنل با نگاشت داده‌ها به فضایی با ابعاد بالاتر، تفکیک‌پذیری کلاس‌ها را در مواردی که الگوهای ترافیک پیچیده و غیرخطی می‌باشند، تضمین می‌کند. RBF[۱۳] با تمرکز بر فاصله‌های محلی بین نمونه‌ها، این توانایی را دارد که تغییرات در ویژگی‌های ترافیک، مانند افزایش غیرعادی و ناگهانی نرخ جریان‌های شبکه و یا کاهش جریان‌های تعاملی در حملات DDoS، را با دقت بالا مدل‌سازی کند.

۳-۳-۱- استخراج ویژگی‌های مدل یادگیری

حملات منع خدمت توزیع‌شده با مختل کردن عملکرد عادی شبکه باعث توقف عملکرد فعالیت‌های نرمال در شبکه می‌شوند. اختلالات ایجاد شده به واسطه ارسال حجم زیادی از داده‌ها به سمت سوئیچ‌ها می‌باشد. مقادیر آماری این داده‌ها در شرایط نرمال و ارسال ترافیک مخرب از سوی مهاجم با یکدیگر تفاوت دارند. بنابراین با توجه به همین موضوع استفاده از ویژگی‌های آماری که بتواند میان ترافیک عادی و مخرب تمایز ایجاد کند حائز اهمیت می‌باشد که در ادامه به برخی از آنها اشاره خواهیم نمود.

الف) سرعت آدرس‌های IP منبع

ویژگی SSIP[۱۴] با تحلیل میانگین نرخ ظهور دستگاه‌های منبع متفاوت در واحد زمان، ابزاری موثر برای تشخیص حملات DDoS در شبکه‌های کامپیوتری فراهم می‌کند. در حالی که در شرایط عادی SSIP به صورت کنترل‌شده حفظ می‌شود، در حملات DDoS شاهد افزایش سریع و غیرعادی آن خواهیم بود. بنابراین، نظارت مستمر بر SSIP می‌تواند به شناسایی سریع حملات و اتخاذ تدابیر لازم جهت کاهش آسیب‌های ناشی از حمله کمک کند. در این رابطه $\sum \mathrm{IP_{src}}$ تعداد کل آدرس‌های IP مبدا منحصر به فرد در بازه زمانی ΔT می‌باشد.

$$\mathrm{SSIP} = \frac{\sum \mathrm{IP_{src}}}{\Delta T} \qquad (۱۳)$$

البته، باید به این نکته توجه نمود که افزایش SSIP به تنهایی لزوماً به معنای وقوع حمله نخواهد بود و باید با سایر معیارها و الگوهای ترافیکی شبکه مورد بررسی قرار گیرد تا مانع از هشدارهای اشتباه شود.

ب) انحراف معیار تعداد بسته‌ها در ورودی‌های جریان

ویژگی SDFP[۱۵] با اندازه‌گیری انحراف معیار تعداد بسته‌ها در ورودی‌های جریان، امکان تشخیص تفاوت در الگوی ترافیک حمله و نرمال را فراهم می‌آورد. در حملات DDoS، به دلیل استفاده از شبکه‌ای از دستگاه‌های آلوده و همچنین تشابه میان ترافیک ارسالی از سمت این میزبان‌ها در سیستم، انحراف معیار تعداد بسته‌ها به سرعت کاهش می‌یابد، در حالی که در شرایط عادی، SDFP به صورت کنترل‌شده و پایدار حفظ می‌شود. بنابراین، نظارت بر این ویژگی آماری می‌تواند به شناسایی سریع حمله و اتخاذ تدابیر لازم کمک کند. در این فرمول mean_packets میانگین تعداد بسته‌ها در جریان‌های بازه زمانی تعیین‌شده می‌باشد. $\mathrm{packets_i}$ تعداد بسته‌های جریان i در بازه زمانی است و N تعداد کل ورودی‌های جریان در این بازه زمانی می‌باشد.

$$\mathrm{SDFP} = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{packets_i} - \mathrm{mean_packets})^2} \qquad (۱۴)$$

پ) انحراف معیار تعداد بایت‌ها در ورودی‌های جریان

SDFB[۱۶] نشان‌دهنده انحراف معیار تعداد بایت‌ها در ورودی‌های جریان است. هنگامی که یک حمله در حال انجام است، بسته‌ها معمولاً از نظر اندازه کوچک‌تر از بسته‌های نرمال در زمان عبور ترافیک عادی هستند. این به مهاجم اجازه می‌دهد تا تعداد بیشتری از آنها را در واحد زمان ارسال کند. بنابراین در هنگام ایجاد حمله منع خدمت این شاخص کاهش پیدا می‌کند. که در آن $\mathrm{bytes_i}$ تعداد بایت‌های جریان ورودی i در بازه زمانی ΔT است و N نیز تعداد ورودی‌های جریان در این بازه زمانی می‌باشد.

$$\mathrm{SDFB} = \sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(\mathrm{bytes_i} - \mathrm{mean_bytes})^2} \qquad (۱۵)$$

ت) سرعت ورودی جریان

شاخص SFE[۱۷] نشان‌دهنده تعداد ورودی‌های جریان ثبت شده سوئیچ، در بازه زمانی تعیین شده می‌باشد. در یک موقعیت حمله، ورودی‌های جریان به دلیل ارسال حجم بالایی از بسته‌ها با آدرس‌ها و پورت‌های تصادفی به شدت افزایش می‌یابد. همین امر موجب سربار پردازشی برای کنترل‌کننده و اشباع جدول جریان سوئیچ‌های شبکه می‌شود. در رابطه (۱۶) N تعداد کل ورودی‌های

جریان و ΔT دوره نمونه برداری می‌باشد.

$$SFE = \frac{N}{\Delta T} \quad (۱۶)$$

ث)رابطه بین جریان‌های تعاملی و جریان‌های کل

در این معیار پارامترهای Int_IP تعداد جریان‌های تعاملی و N تعداد کل ورودی‌های جریان است. این ویژگی معیاری است که نشان می‌دهد چه نسبتی از جریان‌های شبکه به صورت متقابل هستند. یک جریان متقابل زمانی رخ می‌دهد که دو جریان i و j وجود داشته باشند که آدرس IP مبدأ یکی، آدرس IP مقصد دیگری باشد و بالعکس. به عبارت دیگر، یک ارتباط دوطرفه و رفت و برگشتی بین دو آدرس IP برقرار باشد.

$$Src_{IP_i} = Dst_{IP_j}$$
$$Dst_{IP_i} = Src_{IP_j} \quad (۱۷)$$

در شرایط عادی، این ویژگی تقریبا برای همهٔ بسته‌های موجود در شبکه رخ می‌دهد و نزدیک به یک است، اما در حالت حمله به دلیل عدم تقارن جریان‌ها، حجم زیادی از درخواست‌ها بدون پاسخ خواهد ماند. در موقعیت‌های حمله، مهاجم قربانی خود را با بسته‌هایی بمباران می‌کند که قادر به پاسخگویی به آنها نباشد. با افزایش N و ثابت ماندن تعداد Int_IP این ویژگی به شدت کاهش می‌یابد.

$$RPF = \frac{Int_IP}{N} \quad (۱۸)$$

۴- نتایج ارزیابی

محیط شبیه‌سازی این پژوهش با استفاده از نرم‌افزار مجازی‌ساز VMware Workstation بر روی یک سیستم میزبان مجهز به پردازنده Intel Core i5-12450H پیاده‌سازی شده است. ماشین مجازی (VM) اختصاصی با تخصیص ۴ هسته پردازشی و ۸ گیگابایت حافظه رم (RAM) راه‌اندازی شده و از ۵۰ گیگابایت فضای ذخیره‌سازی هارددیسک برای نصب سیستم‌عامل Ubuntu 20.04.6 LTS استفاده شده است. این سیستم‌عامل به‌عنوان پلتفرم اصلی برای اجرای ابزار Mininet نسخه ۲,۳,۰ و کنترل‌کننده Ryu نسخه ۴,۳۴ به‌کار گرفته شده است. همچنین ابزار hping3 برای تولید ترافیک در شبکه استفاده می‌شود که از طریق آن ترافیک حمله و نرمال ایجاد خواهندشد و در ادامه از دو کنترل‌کننده مستقل Ryu که از زبان برنامه نویسی python 3.8 پشتیبانی می‌کنند جهت مسیریابی Packet-inها و تشخیص حمله استفاده می‌شود. کنترلی که جهت تشخیص حمله از آن استفاده شده، با توجه به روش‌های تشخیصی طراحی در آن باید توانایی شناسایی حملات DDoS را به صورت متمرکز و توزیع‌شده، همراه با نرخ‌های متفاوت داشته باشد.

۴-۱- جمع آوری جریان

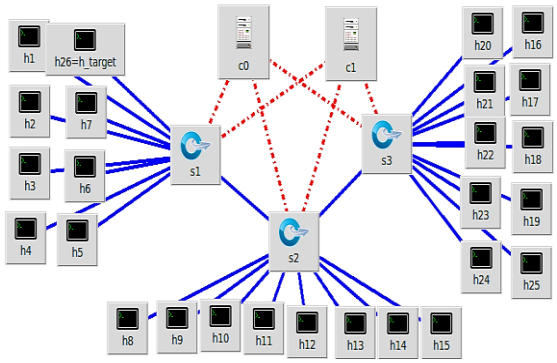
ابزار hping3 یک ابزار قدرتمند و انعطاف‌پذیر در حوزه شبکه است که برای تولید و تحلیل ترافیک شبکه، تست امنیت، بررسی فایروال‌ها، ارزیابی عملکرد شبکه و شبیه‌سازی انواع ترافیک‌ها از جمله ترافیک عادی و حمله استفاده می‌شود و از پروتکل‌های TCP، UDP و ICMP پشتیبانی می‌کند. در این مطالعه، از دو اسکریپت مشابه با hping3 برای تولید ترافیک عادی با تعداد و اندازه بسته‌های تصادفی (تفاوت اصلی در سرعت ارسال) و یک اسکریپت برای ترافیک حمله با آدرس‌های IP تصادفی و حجم بالا استفاده شده است. مجموعه داده جمع‌آوری‌شده شامل ۶۰۰۰۰ نمونه جریان ترافیک است. ۸۰٪ این مقدار (۴۸۰۰۰) جهت آموزش و ۲۰٪ باقی مانده جهت ارزیابی استفاده می‌شود. برای شبیه‌سازی حمله، ۱۵۰۰۰رکورد حمله شامل انواع حمله مانند ICMP Flood، UDP Flood و TCP SYN Flood شدند. رکوردهای نرمال شامل ۷۵٪ ترافیک TCP و ۱۸٪ ترافیک UDP و ۷٪ ترافیک ICMP که ترکیبی مناسب از ارتباطات معمول شبکه را ارائه می‌دهند. ترکیب نهایی مجموعه داده‌های آزمایشی در جدول ۳ نشان داده شده است.

۴-۲- توپولوژی شبکه

برای طراحی و پیاده‌سازی توپولوژی شبکه، از ابزار MiniEdit استفاده شده است. این ابزار به صورت گرافیکی امکان ساخت توپولوژی‌های SDN در بستر Mininet را فراهم می‌نماید. شبکه دارای ۲۶ میزبان و سه سوئیچ جریان باز[۱۸] و دو کنترل‌کننده Ryu است. تمام هاست‌ها به سوئیچ‌ها و هر یک از سوئیچ‌ها به هر دو کنترل‌کننده متصل هستند.

جدول ۱. مجموعه داده جهت آموزش و ارزیابی مدل یادگیری

مجموعه داده	مجموع جریان نرمال	مجموع جریان حمله	مجموع
مجموعه آموزشی	۳۶۰۰۰	۱۲۰۰۰	۴۸۰۰۰
مجموعه ارزیابی	۹۰۰۰	۳۰۰۰	۱۲۰۰۰
مجموع	۴۵۰۰۰	۱۵۰۰۰	۶۰۰۰۰



شکل۴: توپولوژی شبکه پیشنهادی

- کنترل‌کننده C0 :

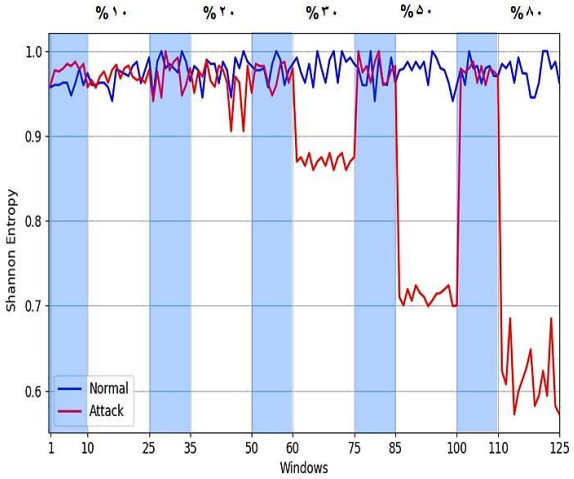
کنترل‌کننده C0 به عنوان کنترل‌کننده اصلی عمل می‌کند. وظیفه مسیریابی بسته‌های جدیدی که در سوئیچ قوانینی برای آنها وجود ندارد را برعهده دارد و زمانی‌که یک بسته Packet-in دریافت کند، جریان ورودی جدید در سوئیچ بر اساس آدرس MAC مبدا و مقصد، آدرس IP مبدا و مقصد و نوع پروتکل بسته نصب می‌کند تا از پرسش مجدد کنترل‌کننده جهت مسیریابی بسته‌های مشابه جلوگیری به عمل آید.

- کنترل‌کننده C1 :

وظیفه کنترل‌کننده C1 مدیریت درخواست‌ها و تعامل با سوئیچ‌ها از طریق API تعیین‌شده است. این کنترل‌کننده به‌صورت مستقل از C0 اجرا و در یک فرآیند مدیریتی جدید راه‌اندازی می‌شود. این جداسازی به این دلیل می‌باشد که در صورتی که C0 در شرایط حمله یا به وسیله ترافیک بالا تحت فشار قرار گیرد، این کنترل‌کننده همچنان می‌تواند عملیات شناسایی ترافیک مخرب را بااستفاده از روش‌های آماری و یادگیری انجام دهد. در واقع به عنوان واسطه‌ای عمل می‌کند که داده‌های آماری سوئیچ را به وسیله پروتکل OpenFlow جمع‌آوری و به ماژول تشخیص حمله ارائه می‌دهد. این داده‌ها برای محاسبه ویژگی‌های کلیدی مورد استفاده در رویکردهای پیش‌تشخیص آنتروپی و نرخ Packet-in و روش تشخیصی یادگیری ماشین جهت شناسایی حمله بکار گرفته می‌شوند. جهت کاهش بار محاسباتی کنترل‌کننده C1 از ارسال سه دسته پیام، بسته‌های بدون جریان مطابق در سوئیچ، تغییرات وضعیت پورت‌ها و حذف جریان‌ها برای کنترل‌کننده تشخیص حمله ارسال نمی‌شود و تنها آمار پنج دسته ویژگی تعداد بسته، تعداد بایت، آدرس IP مبدا، آدرس IP مقصد و تعداد بسته‌هایی که قانونی در جدول جریان سوییچ برای آنها وجود ندارد، از طریق سوئیچ ارسال می‌شود. با این طراحی حتی در صورت عدم پاسخگویی کنترل‌کننده مسیریاب، کنترل‌کننده C1 با ایجاد قوانینی در سوئیچ‌ها برای مسدودسازی ترافیک مشکوک و ارسال هشدار به سیستم مدیریت شبکه، از گسترش حمله جلوگیری می‌کند.

۴-۳- مقایسه عملکرد آنتروپی شانون و رنی

در این مطالعه، اثربخشی دو نوع آنتروپی، یعنی آنتروپی شانون و آنتروپی رنی را برای تشخیص ترافیک حمله در سیستم‌های شبکه ارزیابی کردیم. تجزیه و تحلیل بر اساس مجموعه داده‌ای است که شامل ترافیک نرمال و حمله در نرخ‌های مختلف حمله (۱۰٪، ۲۰٪، ۵۰٪ و ۸۰٪) می‌باشد.

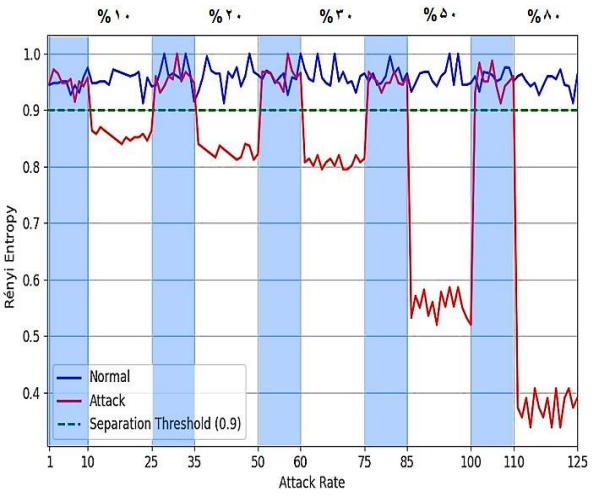


شکل۵: رفتار آنتروپی شانون در نرخ‌های متفاوت حمله

نتایج از طریق چندین شکل، از جمله نمودارهای رفتار آنتروپی (شکل‌های۴ و ۵) و ماتریس‌های سردرگمی (شکل ۶)، مجموعاً عملکرد برتر آنتروپی Rényi را در تشخیص حمله نشان می‌دهند. شکل ۴ رفتار آنتروپی شانون را برای ترافیک نرمال و حمله در نرخ‌های مختلف حمله نشان می‌دهد که بازه‌های ۱۰تایی ترافیک بیان‌کننده ترافیک نرمال و بازه‌های ۱۵تایی ترافیک حمله با نرخ‌های ۱۰٪، ۲۰٪، ۳۰٪، ۵۰٪ و ۸۰٪ را نشان می‌دهد.

مقادیر آنتروپی شانون برای ترافیک عادی نسبتاً پایدار باقی می‌ماند و دربازه ۰,۹۴ و ۱,۰۰ متغیر است، در حالی که ترافیک حمله کاهش محسوسی را نشان می‌دهد، به ویژه در نرخ‌های حمله بالاتر (۳۰٪، ۵۰٪ و ۸۰٪)، که مقادیر تا ۰,۵۷ کاهش می‌یابد. با این حال، در نرخ حمله کمتر (۱۰٪ و ۲۰٪)، همپوشانی بین مقادیر آنتروپی معمولی و حمله قابل توجه نیست و نمی‌توان آستانه‌ای جهت تشخیص ناهنجاری به وجود آمده مدنظر قرار داد، که نشان دهنده چالش‌های بالقوه در تشخیص ترافیک حمله از ترافیک نرمال با استفاده از آنتروپی شانون است. در مقابل، شکل ۵

آنتروپی Rényi ($\alpha = 2$) را برای شرایط ترافیکی یکسان نشان می‌دهد. آنتروپی Rényi برای ترافیک عادی به طور مداوم بالا است و از ۰,۹۰ تا ۱,۰۰ متغیر بوده، در حالی که ترافیک حمله کاهش شدیدتری را نشان می‌دهد، که تا مقادیر ۰,۳۴ با نرخ حمله ۸۰٪ کاهش می‌یابد. خاطر نشان می‌گردد، حتی در نرخ‌های حمله پایین‌تر (۱۰٪ و ۲۰٪)، فاصله بین ترافیک عادی و حمله در مقایسه با آنتروپی شانون واضح‌تر بوده، و امکان انتخاب حدآستانه‌ای جهت تشخیص حمله ایجاد می‌شود.

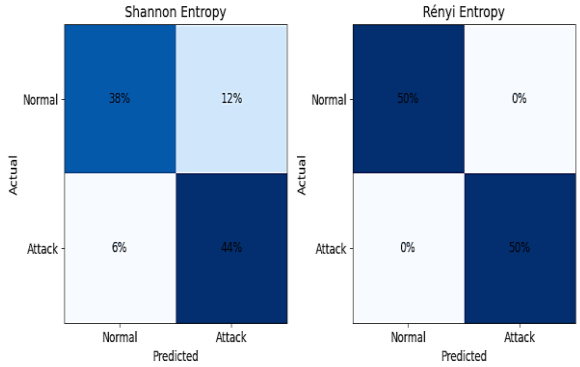


شکل۶: رفتار آنتروپی رنی در نرخ‌های متفاوت حمله

برای تقویت بیشتر تشخیص، ما از رویکرد آنتروپی Rényi برای تشخیص حمله به چند مقصد(توزیع‌شده)، استفاده کردیم. این رویکرد اصلاح شده آستانه جدایی را مقدار ۰,۹ معرفی می‌کند، که به طور موثر ترافیک عادی را از ترافیک حمله در تمام نرخ‌های حمله متمایز می‌کند. نتایج نشان می‌دهد که آنتروپی Rényi توانایی بیشتری در تفکیک ترافیک حمله و نرمال را در نرخ‌های متفاوت دارا می‌باشد. با استفاده از آستانه تعیین شده می‌توان میان ترافیک نرمال و حمله تمایز ایجاد کرد و حملات (متمرکز) با نرخ پایین را نیز تشخیص داد.

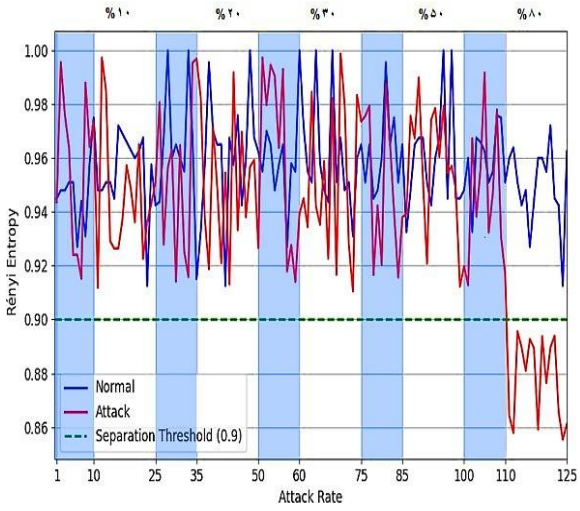
با مقایسه عملکرد دو آنتروپی به وسیله ماتریس در هم‌ریختگی[۱۹] ارائه شده در شکل ۶ این برتری تایید می‌شود. برای آنتروپی شانون، نرخ مثبت واقعی ۴۴٪ برای تشخیص حمله و نرخ منفی واقعی ۳۸٪ برای ترافیک عادی را نشان می‌دهد که منجر به دقت کلی ۸۲٪ می‌شود. با این حال، نرخ مثبت کاذب (۱۲٪) و نرخ منفی کاذب (۶٪) نشان‌دهنده نرخ خطای بالاتر، به ویژه در طبقه‌بندی اشتباه ترافیک عادی به عنوان حمله است. در مقابل، آنتروپی Rényi با دقت ۱۰۰٪ بدون هیچ گونه خطایی میان ترافیک نرمال و حمله تمایز ایجاد می‌کند.

اما ذکر این نکته ضروری است که در سناریوی ترافیک توزیع‌شده، آنتروپی رنی نمی‌تواند به‌طور مؤثر تمایز قابل‌اعتمادی بین ترافیک



شکل۷: مقایسه ماتریس درهم‌ریختگی آنتروپی شانون و رنی

نرمال و حمله ایجاد کند. دلیل این امر به ماهیت توزیع‌شده ترافیک برمی‌گردد که باعث می‌شود مقادیر آنتروپی رنی برای ترافیک حمله، به‌ویژه در نرخ‌های پایین‌تر، کاهش قابل‌توجه‌ای نداشته باشد و مقدار آن از حدآستانه تعیین شده کمتر نشود. این همپوشانی، به‌خصوص در نرخ‌های پایین‌تر حمله، تشخیص دقیق حملات را دشوار می‌کند.



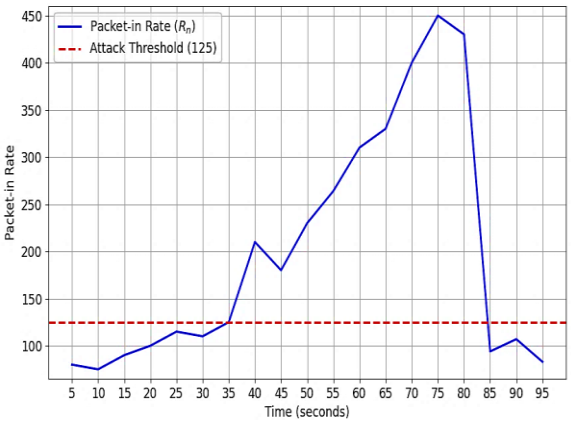
شکل۸: ایجاد حمله توزیع‌شده بین مقاصد مختلف

۴-۴- تحلیل ترافیک با روش پیش‌تشخیص Packet-in

برای رفع این مشکل و بهبود تشخیص حملات در سناریوهای ترافیک توزیع‌شده، رویکرد پیشنهادی استفاده از نرخ پیام‌های Packet-in است که در شکل ۱۰ نشان داده شده است. نرخ Packet-in به تعداد پیام‌های ارسالی به کنترل‌کننده در شبکه‌های نرم‌افزارمحور در واحد زمان اشاره دارد. این معیار به‌عنوان یک شاخص اساسی برای شناسایی رفتار غیرعادی در شبکه استفاده می‌شود، زیرا حملات DDoS معمولاً باعث افزایش

قابل‌توجه این معیار می‌شوند.

در نمودار نرخ Packet-in، محور افقی زمان در بازه‌های ۵ ثانیه‌ای و محور عمودی نرخ Packet-in را نشان می‌دهد و یک آستانه حمله به‌صورت خط‌چین افقی قرمز مشخص شده است که در ابتدا متناسب با مقیاس شبکه و میزان تجربی انتخاب می شود ودر مراحل بعدی با توجه به رابطه (۸) در شرایط عادی بروزرسانی می‌شود. در برخی بازه‌های زمانی، نرخ Packet-in به‌طور قابل‌توجهی از آستانه فراتر می‌رود که نشان‌دهنده فعالیت غیرعادی یا حمله است. برخلاف آنتروپی رنی در سناریوی توزیع‌شده، نرخ Packet-in به دلیل حساسیت به تغییرات ناگهانی در حجم ترافیک کنترلی، قادر است رفتارهای حمله را با دقت بیشتری شناسایی کند. این معیار به‌طور ویژه در شبکه‌های نرم‌افزارمحور که کنترل‌کننده مرکزی ترافیک را مدیریت می‌نماید، مؤثر واقع شده، زیرا حملات توزیع‌شده معمولاً الگوهای مشخصی از افزایش نرخ Packet-in به‌همراه خود ایجاد می‌کنند.



شکل۹: نمودار نرخ پیام‌های Packet-in در دوره‌های ۵ ثانیه‌ای

در مورد استفاده از آنتروپی رنی و نرخ پیام‌های Packet-in برای تشخیص حملات در سناریوهای ترافیک توزیع‌شده، این دو معیار به‌عنوان روش‌های پیش‌تشخیص پیشنهاد می‌شوند. این روش‌های پیش‌تشخیص با هدف شناسایی اولیه ترافیک مشکوک و کاهش سربار محاسباتی مدل‌های یادگیری ماشین طراحی شده‌اند.در این رویکرد، آنتروپی رنی و نرخ Packet-in به‌عنوان فیلترهای اولیه عمل می‌کنند تا ترافیک شبکه را به‌صورت سریع و با هزینه محاسباتی پایین بررسی کنند. در صورتی که هر یک از این معیارها ترافیک را به‌عنوان مشکوک تشخیص دهند، ترافیک برای تحلیل دقیق‌تر به یک مدل یادگیری ماشین ارسال می‌شود که بر اساس ویژگی‌های آماری استخراج شده از ترافیک، نوع آن را با دقت بالا تعیین می‌کند.

۴-۵- بررسی تغییرات ویژگی‌های مدل یادگیری ماشین

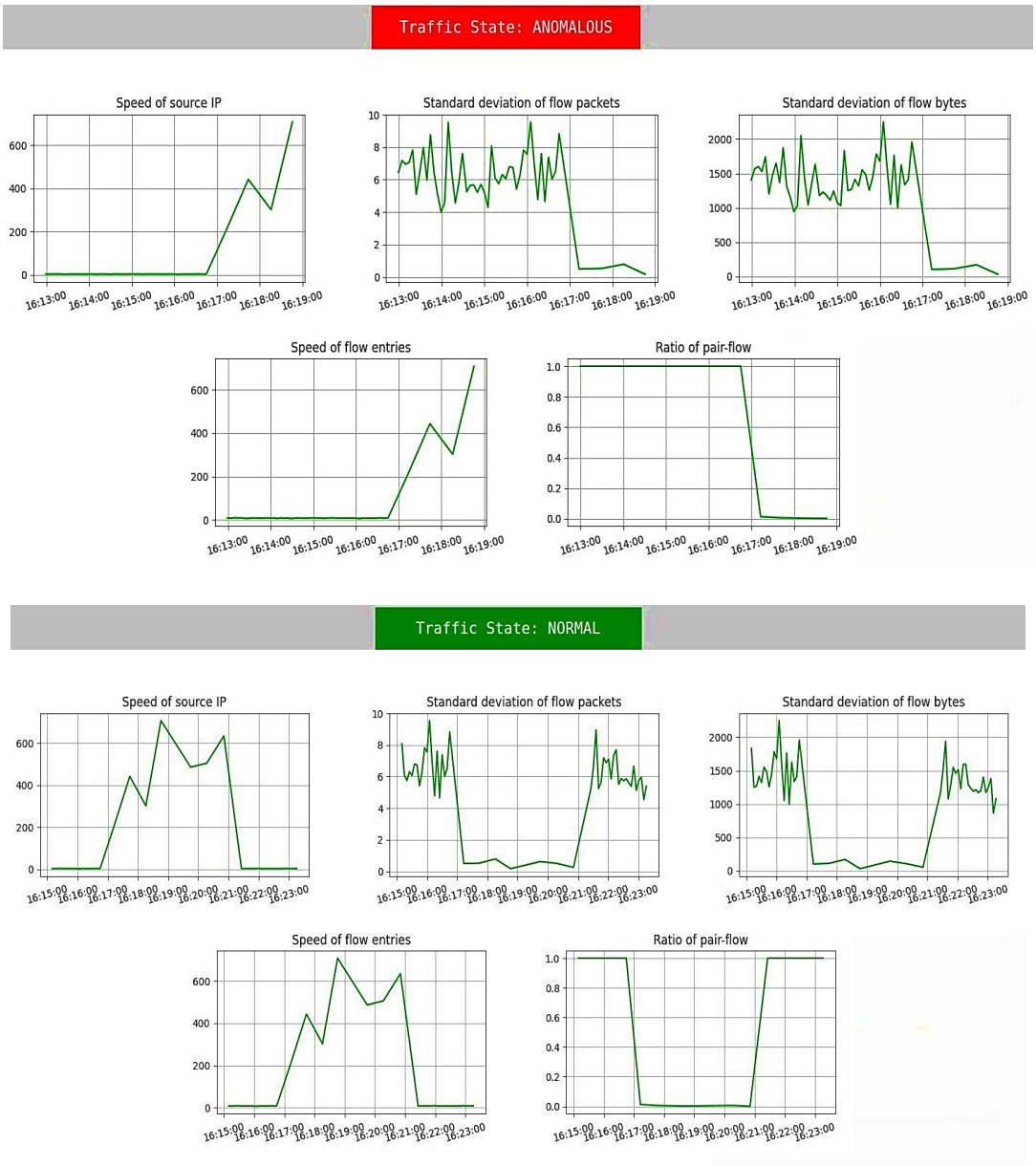
همان گونه که درقسمت استخراج ویژگی‌ها برای مدل یادگیری ماشین توضیح داده شد مدل یادگیری از ۵ ویژگی جهت شناسایی ترافیک نرمال از حمله استفاده می‌کند. مطابق با شکل ۱۰ و برچسب‌های عنوان وضعیت مبنی بر ناهنجار یا نرمال نحوه رفتار این ویژگی‌ها در زمان عبور ترافیک حمله از نرمال مشخص شده‌است.

ویژگی سرعت آدرس‌های IP منبع در شرایط نرمال تعداد آدرس‌های IP مبدأ منحصر به فرد معمولاً محدود و پایدار است، زیرا کاربران عادی از تعداد محدودی منبع ترافیک ایجاد می‌کنند. این امر باعث می‌شود SSIP در سطح پایینی، مثلاً ۱۰-۱۵ IP در هر ثانیه حفظ شود و تغییرات آن به صورت کنترل‌شده باشد. در حملات منع خدمت توزیع‌شده مهاجمان از تعداد زیادی دستگاه آلوده با آدرس‌های IP متفاوت استفاده می‌کنند، که منجر به افزایش سریع و غیرعادی این ویژگی می‌شود.

در حملات منع خدمت توزیع‌شده مهاجم از شبکه‌ای از دستگاه‌های آلوده استفاده می‌کنند که ترافیک مشابهی تولید می‌کند همین امر باعث ایجاد تعداد ثابت بسته‌ها و بایت‌ها مشابه در هر جریان می‌شود که این تشابه سبب کاهش پراکندگی ویژگی‌های SDFP و SDFB می‌گردد.

ویژگی سرعت ورودی جریان در شرایط نرمال تعداد محدودی در حدود ۱۰-۲۵ جریان در ثانیه است اما در شرایط حمله برای ایجاد سربار پردازشی جهت مسیریابی و اشباع جداول جریان در سوئیچ‌های OpenFlow این نرخ به شدت افزایش می‌یابد.

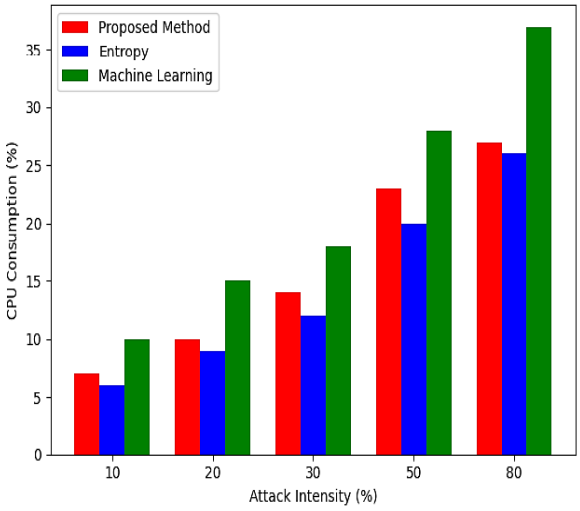
در ویژگی رابطه بین جریان‌های تعاملی در ترافیک عادی شبکه بیشتر جریان‌ها به صورت تعاملی هستند، زیرا هر درخواست با پاسخی همراه است. این امر باعث می‌شود RPF نزدیک به ۱ باشد، که نشان‌دهنده تقارن و تعادل در جریان‌ها است. در شرایط مهاجمان قربانی را با حجم بالایی از بسته‌ها بمباران می‌کنند که معمولاً پاسخ‌دهی به آن‌ها ممکن نیست (مانند SYN packets بدون ACK یا پینگ‌های بدون پاسخ). این عدم تقارن باعث می‌شود، RPF به شدت کاهش پیدا کرده و به حدود ۰,۱-۰,۳ برسد.



شکل۱۰: تشخیص نوع ترافیک با استفاده از ویژگی‌های تعیین شده توسط مدل ML

۴-۶- مقایسه میزان استفاده از CPU و زمان تشخیص

جهت بررسی کارایی روش پیشنهادی، میانگین مصرف پردازنده کنترل‌کننده SDN برای سه روش تشخیص حمله سیل DDoS، شامل روش مبتنی‌بر یادگیری ماشین، روش آماری مبتنی‌بر آنتروپی و روش پیشنهادی در سطوح مختلف شدت حمله (۱۰٪، ۲۰٪، ۳۰٪، ۵۰٪ و ۸۰٪) بررسی می‌گردد. در این پژوهش نرخ جریان پیوسته ۵ مگابیت در ثانیه در طول مدت ۲۰ آزمایش متوالی حفظ شده‌است. مدل یادگیری ماشینی SVM با کرنل RBF که برای تحلیل کل ترافیک شبکه به‌کار گرفته شده است به دلیل نیاز به پردازش جامع داده‌ها و بهینه‌سازی پیچیده کرنل در SVM، بالاترین میزان مصرف پردازنده را در تمامی سطوح عملکردی به نمایش می‌گذارد. روش پیشنهادی، اگرچه از منظر سربار محاسباتی اندکی بیشتر از روش آماری مبتنی بر آنتروپی عمل می‌کند، اما همان‌گونه که بیان شد، در روش‌های مبتنی بر آنتروپی مهاجم با ایجاد ترافیک جعلی به سایر مقاصد می‌تواند مانع کاهش آنتروپی تا حدآستانه شود که این امر موجب ایجاد خطا در تشخیص با دقت بالا در شناسایی حمله می‌شود. از این رو، با وجود پذیرش افزایش اندک سربار محاسباتی در روش پیشنهادی(۱تا ۳ درصد)، نسبت به روش آماری مبتنی‌بر آنتروپی، دقت بالاتر و پایداری آن در شناسایی الگوهای پیچیده حملات، موجب برتری این روش نسبت به دو روش دیگر است.



شکل۱۱: میزان سربار محاسباتی برای روش‌های تشخیص حمله در درصدهای متفاوت حمله

جدول ۲ نتایج مقایسه زمان‌های تشخیص حمله را برای سه روش مختلف ارائه داده‌است. روش تشخیص پیشنهاد شده در مقابل سناریوی توزیع‌شدگی ترافیک تشخیص بهتری را به همراه دارد. اما به دلیل اضافه شدن روش پیش‌تشخیص نرخ Packet-in به آنتروپی سربار محاسباتی و زمان پردازش مقدار کمی نسبت به روش آنتروپی افزایش نشان می‌دهد که با توجه به درصد بالاتر تشخیص حمله معقول و قابل توجیه است.

جدول ۲: مقایسه زمان تشخیص حمله

روش تشخیص	زمان تشخیص
روش آنتروپی	۴,۷۳
روش یادگیری ماشین SVM	۷,۶۵
روش پیشنهادی	۵,۰۲

۴-۷- مقایسه روش‌های تشخیص حمله بر اساس شدت

سه روش تشخیص حمله شامل روش آماری مبتنی بر آنتروپی، الگوریتم یادگیری ماشین SVM، و روش پیشنهادی با شدت‌های مختلف حمله در جدول ۳ نمایش داده شدند. همچنین ۴ شاخص ارزیابی محاسبه شدند که به ازای هر ۵ نرخ حمله محاسبات انجام شد که روش پیشنهادی کمترین خطای مثبت و منفی را ثبت کرده است. در مقابل روش آنتروپی در درصد های مختلف حمله دقت پایین‌تری داشته و نرخ خطا در این روش بالاتر می‌باشد.

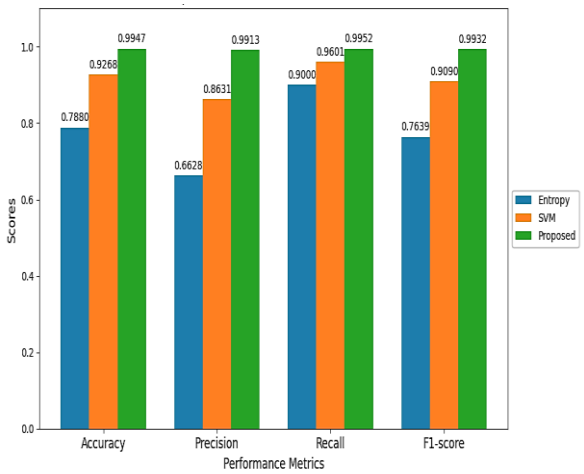
روش یادگیری ماشین SVM نیز در سطوح مختلف حمله عملکرد بهتری نسبت به روش آنتروپی ارائه می‌دهد اما همچنان نسبت به روش پیشنهادشده دقت کمتری دارد.

جدول ۳: ارزیابی روش‌های تشخیصی با شاخص‌های طبقه‌بندی

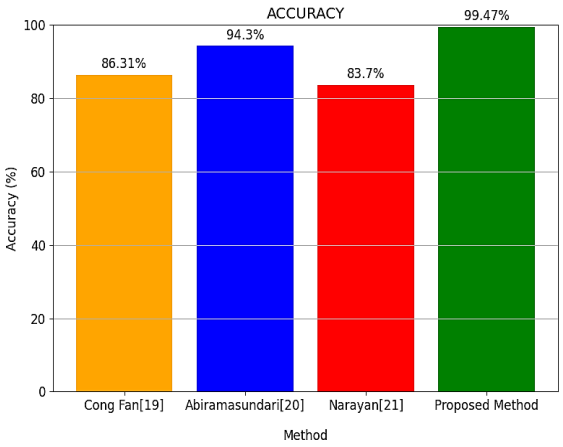
روش تشخیص حمله	نرخ حمله	FN	TN	FP	TP
آنتروپی	٪۱۰	۱۲۰	۷۵۶۰	۳۲۴۰	۱۰۸۰
یادگیری ماشینSVM		۴۸	۹۷۲۰	۱۰۸۰	۱۱۵۲
روش پیشنهادی		۶	۱۰۷۴۶	۵۴	۱۱۹۴
آنتروپی	٪۲۰	۲۴۰	۶۷۲۰	۲۸۸۰	۲۱۶۰
یادگیری ماشینSVM		۹۶	۸۶۴۰	۹۶۰	۲۳۰۴
روش پیشنهادی		۱۲	۹۵۵۲	۴۸	۲۳۸۸
آنتروپی	٪۳۰	۳۶۰	۵۸۸۰	۲۵۲۰	۳۲۴۰
یادگیری ماشینSVM		۱۴۴	۷۵۶۰	۸۴۰	۳۴۵۶
روش پیشنهادی		۱۸	۸۳۵۸	۴۲	۳۵۸۲
آنتروپی	٪۵۰	۶۰۰	۴۲۰۰	۱۸۰۰	۵۴۰۰
یادگیری ماشینSVM		۲۴۰	۵۴۰۰	۶۰۰	۵۷۶۰
روش پیشنهادی		۳۰	۵۹۷۰	۳۰	۵۹۷۰
آنتروپی	٪۸۰	۹۶۰	۲۴۰۰	۰	۸۶۴۰
یادگیری ماشینSVM		۳۸۴	۲۴۰۰	۰	۹۲۱۶
روش پیشنهادی		۴۸	۲۳۷۰	۳۰	۹۵۵۲

۴-۸- عملکرد روش پیشنهادی و مقایسه با مطالعات

شاخص‌های عملکردی و ماتریس‌های درهم‌ریختگی حاصل از این رویکرد ترکیبی نشان‌دهنده توانایی بالای آن در تشخیص حملات با حداقل خطا هستند که می‌تواند به جای بررسی کل ترافیک شبکه توسط مدل‌های یادگیری تنها ترافیک مشکوک را که میزان آن بسیار کمتر از ترافیک کلی شبکه است با دقت بالا جهت تشخیص دقیق‌تر بررسی کند. بر همین اساس مطابق شکل ۱۱ میزان دقت ۹۹/۴۷ را برای این روش بدست آمده‌است.



شکل۱۲: مقایسه شاخص‌های عملکردی سه روش تشخیص حمله



شکل۱۳: مقایسه میزان دقت روش پیشنهادی

در روش پیشنهادی به بررسی سناریوهای متفاوت از حمله پرداخته شده‌است که ضمن تشخیص حمله به یک مقصد، حمله‌هایی که قصد فریب روش آنتروپی را دارند نیز شناسایی خواهند شد. روش پیشنهادی می‌تواند به صورت مؤثری حملات را شناسایی کند که در مقایسه با سه پژوهش دیگر نتایج نشان دهنده نرخ تشخیص بهتر برای روش پیشنهادی می‌باشد. در پژوهش[۱۹] کنگ فن با استفاده از روش آنتروپی نرخ دقت ۸۶/۳۱ درصد را کسب می‌نماید و در پژوهش [۲۰] با استفاده از روش یادگیری ماشین SVM در مجموعه داده‌های متفاوت به طور میانگین نرخ دقت ۹۴/۳ درصد به‌دست می‌آورد. پژوهش[۲۱] از روش آنتروپی مبدأ به‌عنوان روش پیش‌تشخیص و از رویکرد یادگیری ماشین به‌عنوان روش تشخیصی استفاده می‌کند اما چون همان‌گونه که گفته شد مهاجم می‌تواند در روش آنتروپی، ترافیک جعلی را به گونه‌ای توزیع کند که تا حدآستانه تغییرنکند امکان ایجاد خطا در رویکرد را دارد و از طرف دیگرکه چون در این پژوهش روش پیش‌تشخیص این قابلیت را دارد که در مورد ترافیک نرمال اظهار نظر کند میزان خطا افزایش می‌یابد و این شاخص تا ۸۳/۷ درصد کاهش می‌یابد. اما پژوهش حاضر این شاخص عملکردی را به میزان ۹۹/۴۷ درصد افزایش می‌دهد که می‌تواند حملات را با در نظر گرفتن سناریوهای احتمالی تشخیص دهد.

۵- نتیجه گیری

در این پژوهش از دو کنترل‌کننده، یکی جهت مسیریابی بسته‌ها و دیگری جهت تشخیص حمله استفاده شده‌است. از روش‌های آنتروپی و نرخ Packet-in برای پیش‌تشخیص استفاده شدند. روش آنتروپی رنی این توانایی را دارد که حملات متمرکز را تا حد خوبی تشخیص دهد اما در صورتی که حمله بین مقاصد مختلف توزیع شود این روش دقت خود را دست می‌دهد و از روش پیش‌تشخیص نرخ Packet-in برای شناسایی حملات DDoS استفاده می‌شود. در صورتی که یکی از روش‌های پیش‌تشخیص وضعیت ترافیک را مشکوک در نظر بگیرد، جهت بررسی بیشتر به مدل یادگیری ماشین ارسال می‌شود. تحت این شرایط و بر اساس آزمایش‌ها و نتایج ارزیابی، سیستم به دقت متوسط ۹۹,۴۷ درصد را کسب می‌نماید که می‌توان با استفاده از روش‌های آماری بدون ایجاد سربار محاسباتی زیاد عمل پیش‌تشخیص را انجام داد و تنها ترافیک مشکوک را برای مدل یادگیری ماشین که سربار محاسباتی بیشتری دارد اما با دقت بالایی حملات DDoS را تشخیص می‌دهند استفاده شود. برای رویکرد آینده استفاده از یادگیری فدرال پیشنهاد می‌شود. این روش تشخیصی در شبکه‌های SDN با آموزش مدل‌های ML روی داده‌های محلی بدون انتقال داده‌های خام، حریم خصوصی را حفظ کرده و هزینه‌های محاسباتی را توزیع می‌کند. این روش دقت را با مدل‌های محلی و جهانی افزایش می‌دهد، اما چالش‌هایی مانند حملات خرابکارانه و نیاز به مدیریت پیچیدگی را به همراه دارد که باید در تحقیقات آینده بررسی شوند.

مراجع

[1] W. Zhijun, X. Qing, W. Jingjie, Y. Meng, and L. Liang, "Low-rate DDoS attack detection based on factorization machine in software defined network," IEEE Access, vol. 8, pp. 17404-17418, 2020.

[2] S. G. Rawat et al., "A survey of ddos attacks detection schemes in SDN environment," in 2023 International Conference on Computer, Information and Telecommunication Systems (CITS), 2023: IEEE, pp. 01-06.

[3] A. Hirsi et al., "Comprehensive analysis of ddos anomaly detection in software-defined networks," IEEE Access, 2025.

[4] Q.-Y. Zuo, M. Chen, G.-S. Zhao, C.-Y. Xing, G.-M. Zhang, and P. Jiang, "Research on OpenFlow-based SDN technologies," Journal of Software, vol. 24, no. 5, pp. 1078-1097, 2013.

[5] J. Naous, D. Erickson, G. A. Covington, G. Appenzeller, and N. McKeown, "Implementing an OpenFlow switch on the NetFPGA platform," in Proceedings of the 4th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 2008, pp. 1-9.

[6] U. B. Clinton, N. Hoque, and K. Robindro Singh, "Classification of DDoS attack traffic on SDN network environment using deep learning," Cybersecurity, vol. 7, no. 1, p. 23, 2024.

[7] X. Wang et al., "Deep reinforcement learning: A survey," IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 4, pp. 5064-5078, 2022.

[8] Y. Wang and L. Liu, "Centralized Network Management in SDN-based Communication Network of substation," in 2019 IEEE International Conference on Industrial Internet (ICII), 2019: IEEE, pp. 87-91.

[9] G. G. a. F. A. K.Kalkan, "Defense Mechanisms against DDoS Attacks in SDN Environment," IEEE Communications Magazine, vol. 55, pp. 175-179, 2017.

[10] N. Z. Bawany, Jawwad A. Shamsi, and Khaled Salah, "DDoS attack detection and mitigation using SDN: methods, practices, and solutions," Arabian Journal for Science and Engineering, vol. 42 pp. 425-441, 2017.

[11] T. Omar, A. Ho, and B. Urbina, "Detection of DDoS in SDN environment using entropy-based detection," in IEEE International Symposium on Technologies for Homeland Security (HST), 2019, pp. 1-4.

[12] J. Galeano-Brajones, J. Carmona-Murillo, J. F. Valenzuela-Valdés, and F. Luna-Valero, "Detection and mitigation of DoS and DDoS attacks in IoT-based stateful SDN: An experimental approach," Sensors, vol. 20, no. 3, p. 816, 2020.

[13] Z. Hemmati, G. Mirjalily, and Z. Mohtajollah, "Entropy-based DDoS Attack Detection in SDN using Dynamic Threshold," in 2021 7th International Conference on Signal Processing and Intelligent Systems (ICSPIS), 2021: IEEE, pp. 1-5.

[14] A. d. R. L. Ribeiro, R. Y. C. Santos, and A. C. A. Nascimento, "Anomaly detection technique for intrusion detection in sdn environment using continuous data stream machine learning algorithms," in 2021 IEEE international systems conference (SysCon), 2021: IEEE, pp. 1-7.

[15] A. O. Alzahrani and M. J. Alenazi, "Designing a network intrusion detection system based on machine learning for software defined networks," Future Internet, vol. 13, no. 5, p. 111, 2021.

[16] F. Khashab, J. Moubarak, A. Feghali, and C. Bassil, "DDoS attack detection and mitigation in SDN using machine learning," in 2021 IEEE 7th International Conference on Network Softwarization (NetSoft), 2021: IEEE, pp. 395-401.

[17] V. Deepa, K. M. Sudar, and P. Deepalakshmi, "Design of ensemble learning methods for DDoS detection in SDN environment," in 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 2019: IEEE, pp. 1-6.

[18] Z. Liu, Y. He, W. Wang, and B. Zhang, "DDoS attack detection scheme based on entropy and PSO-BP neural network in SDN," China Communications, vol. 16, no. 7, pp. 144-155, 2019.

[19] N. M. K. Cong Fan, Shi Chen, He Jiang, Yiwen Zhou, Carlene Campbell, "Detection of DDoS Attacks in Software Defined Networking Using Entropy," Applied Sciences, vol. 12, 2022.

[20] S. Abiramasundari and V. Ramaswamy, "Distributed denial-of-service (DDOS) attack detection using supervised machine learning algorithms," Scientific Reports, vol. 15, no. 1, p. 13098, 2025.

[21] D. Narayan, W. Heena, and K. Amit, "A Collaborative Approach to Detecting DDoS Attacks in SDN Using Entropy and Deep Learning," Journal of Telecommunications and Information Technology, 2024.

پاورقی‌ها:

[1] Distributed Denial of Service
[2] Software defined networking
[3] Point of Failure
[4] Machine learning
[5] Random Forest
[6] Shannon entropy
[7] Rényi
[8] Peak
[9] Chebyshev's Inequality
[10] Regularization parameter
[11] Support Vector Machine
[12] Margin
[13] Radial Basis Function
[14] Speed of source IP
[15] Standard deviation of flow packets
[16] Standard deviation of flow bytes
[17] Speed of flow entries
[18] OpenFlow
[19] Confusion matrix