دوفصلنامه محاسبات و سامانه های توزیع شده
سال پنجم، شماره۲، شماره پیاپی۱۰، صفحه ۲۳ تا ۴۲، سال ۱۴۰۱

تشخیص و پیشگیری از حملات نرخ پایین منع سرویس توزیع شده پروتکل HTTP در شبکه‌های نرم افزار محور مبتنی بر سوییچ‌های P4 با بکارگیری الگوریتم‌های یادگیری ماشین

رضا فلاحی کپورچالی[۱]، رضا محمدی*[۲]، محمد نصیری[۳]

[۱] کارشناسی ارشد شبکه‌های کامپیوتری، دانشکده فنی مهندسی، دپارتمان کامپیوتر، دانشگاه بوعلی سینا، همدان

[۲] استادیار گروه مهندسی کامپیوتر، دانشکده فنی مهندسی، دپارتمان کامپیوتر، دانشگاه بوعلی سینا، همدان

[۳] دانشیار گروه مهندسی کامپیوتر، دانشکده فنی مهندسی، دپارتمان کامپیوتر، دانشگاه بوعلی سینا، همدان

چکیده

معماری SDN به دلیل فراهم کردن دید انتزاعی در شبکه، امروزه محبوبیت بسیاری یافته است. در معماری SDN، به دلیل وجود کنترلر مرکزی، بیشتر بار پردازشی بر عهده کنترلر شبکه است. این مرکزیت پردازش، کنترلر شبکه را به هدفی بسیار مناسب برای حملات DDoS تبدیل کرده است. در چند دهه اخیر، روش‌های مختلفی برای مقابله با این حملات ارائه شده است؛ اما با افزایش پیچیدگی حملات، میزان ترافیک شبکه و در نتیجه میزان بار پردازشی روی کنترلر شبکه، محققان درصدد بکارگیری ادوات صفحه داده در پردازش‌های لازم برآمدند. یکی از کارآمدترین روش‌های ارائه شده، معرفی فناوری P4 است. با روی کار آمدن P4، می‌توان از توان پردازشی ادوات صفحه داده در فرآیند تشخیص و پیشگیری حملات DDoS در شبکه‌های SDN استفاده نمود؛ که نتیجه حاصل از آن، کاهش میزان بار پردازشی روی کنترلر شبکه و افزایش انعطاف‌پذیری در ادوات صفحه داده است. در این مقاله، به معرفی یک مدل تشخیص و پیشگیری از حملات slow-rate DDoS با بکارگیری سوییچ‌های P4 و استفاده از تکنیک‌های یادگیری ماشین پرداخته شده است. هدف اصلی ارائه این مدل، بکارگیری سوییچ‌های قابل برنامه‌ریزی P4 در روند تشخیص حملات، به جهت کمینه‌سازی سربار پردازشی کنترلر است. در پیاده‌سازی این مدل از کنترلر ONOS استفاده شده است. فرآیند استخراج مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین، نیازمند تخصیص توان پردازشی بخصوصی است که با بکارگیری سوییچ‌های قابل برنامه‌ریزی P4 و پردازش محلی بسته‌ها در سوییچ، این سربار پردازشی کمینه خواهد شد. در این پژوهش، مدل ارائه شده از جهت زمان تشخیص حمله، مصرف پهنای باند و سربار پردازشی کنترلر مورد ارزیابی قرار گرفته است. بر اساس نتایج به دست آمده، مدل ارائه شده نسبت به حالت معمول SDN، حدود ۶۰ ثانیه بهبود در تشخیص حمله و حدود ۵۰٪ کاهش سربار پردازشی و مصرف پهنای باند را به همراه داشته است. نتایج حاصل، نشانگر آن است که استفاده از ادوات P4 و قابلیت برنامه‌ریزی ادوات صفحه داده، تاثیر بسزایی در تشخیص حملات slow-rate DDoS و بار پردازشی کنترلر در شبکه‌های SDN خواهد داشت.

کلمات کلیدی: پردازشگرهای مستقل از پروتکل و قابل برنامه‌ریزی پکت، حملات نرخ پایین منع سرویس توزیع شده، شبکه‌های نرم‌افزار محور، یادگیری ماشین، اونوس، سرور وب

تاریخچه مقاله:

تاریخ ارسال: ۱۴۰۱/۱۰/۱۰
تاریخ اصلاحات: ۱۴۰۱/۱۱/۱۵
تاریخ پذیرش: ۱۴۰۱/۱۲/۲۵
تاریخ انتشار: ۱۴۰۱/۱۲/۲۹

Keywords:

P4
Slow-rate DDoS
Machine Learning
SDN
ONOS
Web server

* ایمیل نویسنده مسئول:
R.mohammadi@basu.ac.ir

Detection and prevention of slow-rate DDoS attacks on HTTP protocol in P4-based software defined networks using machine learning techniques

Reza Fallahi Kapourchaali[1], Reza Mohammadi[*2], Mohammad Nassiri[3]

[1] Master of Information Technology, Department of Computer Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran.
[2] Assistant Professor, Department of Computer Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran.
[3] Associate Professor, Department of Computer Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran.

Abstract

SDN architecture has become popular nowadays due to the abstract view that it provides. Due to the centralized network Controller in SDN, Most of the processing load is on the controller. This centralized controller has made this architecture a great target to DDoS attacks. Over the few past decades, many detection methods has been proposed; but with increased traffic and complexity of DDoS attacks, researchers aimed to utilize the data plane processing power. One of the most effective methods that has been proposed, is the P4 technology. With P4, we can utilize the processing power of the data plane devices in detection and prevention procedure of DDoS attacks on SDN; which will result the reduction of controller overhead and more flexibility data plane devices. In this research, we proposed a detection and prevention model that utilizes machine learning techniques along with implementation of P4 switches to detect slow-rate DDoS attacks on SDN. The ONOS controller has been used for implementation of this model. The goal of proposing this model, is to use programmable P4 switches in detection procedure, in order to minimize the controller overhead. The procedure of extracting feature values for machine learning models, will result processing overhead for the controller, but with implementing this procedure with P4 switches on data plane and local processing of packets in the switch, the controller overhead will be minimized. The proposed model has been analyzed in terms of detection time, bandwidth consumption and CPU utilization of the controller. In compare to the normal SDN, the results shows about 60 seconds improvement in detection time, about 50% less overhead on bandwidth consumption and CPU utilization in proposed method. The results show that implementation of P4 data plane, with programming the data plane devices, will have significant effects on detection of slow-rate DDoS attacks and processing load of the controller in SDN.

۱ – مقدمه

SDN یک معماری جدید از شبکه‌های اینترنتی است، که برای ساده سازی مدیریت شبکه معرفی شده است؛ این ساده سازی به وسیله جداسازی صفحه داده[1] از صفحه کنترل[2] به دست آمده، که مهم‌ترین ویژگی این معماری نیز به حساب می‌آید. هر یک از دو بخش اصلی صفحه داده و صفحه کنترل شامل زیربخش‌هایی هستند، که در معماری سنتی شبکه نیز وجود دارد، اما در این معماری به شکل ماژولار تغییر شکل داده و همین باعث انعطاف‌پذیری بالا در تولید و بکارگیری تجهیزات شبکه و درنهایت، مدیریت بهتر شبکه شده است.

در معماری سنتی شبکه، ادوات خاص منظوره همراه با توانایی پردازشی بخصوص و محدود، مورد استفاده قرار می‌گیرند که بخش صفحه کنترل و صفحه داده و به دنبال آن مفهوم کنترل و انتقال داده در شبکه، از یکدیگر جدایی ناپذیرند؛ اما با جدا شدن این دو بخش در معماری SDN، مفهوم کنترل و انتقال داده در شبکه دو مفهوم مجزا می‌باشند. در پی این جداسازی سطوح، ادوات معماری سنتی با ادواتی عام منظوره، ساده‌تر و البته ارزان قیمت‌تر جایگزین شده، که توان پردازشی خاصی ندارند؛ این ادوات تنها اجرا کننده سیاست‌هایی هستند که کنترلر شبکه برای آنها تعیین می‌کند، به همین دلیل از ادوات معماری SDN با عنوان "دستگاه‌های گنگ[3]" یاد می‌شود.

در معماری SDN تمام موارد مدیریتی شبکه برعهده کنترلر شبکه است. کنترلر شبکه که در بخش صفحه کنترل قرار می‌گیرد، یک نرم-افزار است که با بکارگیری تعداد خاصی از پروتکل‌ها با بخش صفحه داده ارتباط برقرار کرده و از این راه سیاست‌های اصلی شبکه را روی ادوات صفحه داده پیاده‌سازی می‌کند. نکته قابل توجه در این زمینه آن است که بار پردازشی زیادی در شبکه روی کنترلر بوده و ادوات تنها اجرا کننده نتایج این پردازش‌ها هستند؛ درواقع نقطه قوت و نقطه حساس این معماری هر دو در یک قسمت است؛ وجود کنترلر مرکزی شبکه.

مرکزیت کنترل در این معماری باعث شده که در صورت بروز مشکل در کنترلر شبکه، تمام شبکه تحت تاثیر قرار گیرد؛ همین مورد شبکه‌های SDN را به یک هدف عالی برای انجام انواع حملات سایبری تبدیل کرده است، که حملات DDoS از جمله مهم‌ترین آن‌ها هستند.

حملات DDoS به گونه ای طراحی شده‌اند، که با ایجاد تعداد بالایی از درخواست‌ها توسط کاربران غیرواقعی، یک سیستم را طوری درگیر پاسخگویی به خود کنند که سیستم ظرفیت پاسخگویی به کاربران واقعی را از دست داده و پس از مدتی به طور کامل مختل شود. کنترلر شبکه و معماری SDN در صورت رخداد چنین حملاتی بسیار آسیب‌پذیر خواهند بود که نتیجه آن، اختلال در کل شبکه مورد نظر است.

در راستای مقابله با این حملات پژوهشگران روش‌های مختلفی را ارائه کرده‌اند که در ادامه به آنها اشاره می‌شود، اما نکته قابل توجه آن است که در صورت پیاده‌سازی تمام اقدامات مقابله‌ای در کنترلر شبکه، سربار شدید پردازشی برای کنترلر به وجود خواهد آمد که در کیفیت کلی ارتباطات شبکه تاثیرگذار است. سربارپردازشی روی کنترلر، به دلیل وجود ادوات ساده و فاقد قدرت پردازشی به‌خصوص در معماری SDN، اجتناب ناپذیر است؛ این مساله موجب ارائه روش‌های مختلفی برای انتقال بخشی از پردازش شبکه از صفحه کنترل به صفحه داده شد، که یکی از این روش‌ها معرفی زبان P4 و ادوات مبتنی بر آن است.

فناوری P4 امکان تعیین روند پردازش بسته‌ها را در ادوات صفحه داده از طریق برنامه‌ریزی، فراهم می‌کند. ادوات مبتنی بر P4 دارای توان پردازشی مورد نیاز در بسیاری از کاربرد های شبکه، از جمله تشخیص و جلوگیری حملات، می‌باشند. با بکارگیری این ادوات می-توان بخش زیادی از فرآیند تشخیص و جلوگیری از حملات را در صفحه داده پیاده‌سازی نمود و از این رو مشکل سربار پردازشی روی کنترلر شبکه تا حد بسیاری برطرف می‌شود. در ادامه به بررسی برخی پژوهش‌های صورت گرفته در زمینه تشخیص حملات DDoS در شبکه های SDN می‌پردازیم.

۲ – پیشینه پژوهش

۲-۱- حملات slow-rate DDoS: حملات انکار سرویس توزیع-شده یا DDoS دسته‌ای از حملات مخرب هستند، که مبنای آن‌ها اشباع‌سازی منابع سیستم است. این حملات سیستم یا شبکه هدف خود را به طور دائم مشغول به خود می‌سازند، تا سیستم توانایی ارائه سرویس به کاربران واقعی خود را نداشته باشد؛ در ادامه، این حملات سیستم را وادار می‌سازند تا منابع خود را برای ارائه سرویس به مهاجم بکارگیرد و به طور دائم میزان تقاضا را افزایش داده تا در نهایت ظرفیت پاسخگویی سیستم، تکمیل شده و پس از مدتی سیستم به طور کامل مختل شود.

[1] Data plane
[2] Control plane
[3] Dumb devices

حملات DDoS بسته به رفتاری که در شبکه دارند و سیستمی که مورد هدف قرار می‌دهند، در دسته‌بندی‌های مختلف قرار می‌گیرند. نوع خاصی از این حملات، slow-rate DDoS است که به شدت مخرب بوده و هدف آن بیشتر سرورهای HTTP هستند. در حملات DDoS به صورت پیش‌فرض، ماشین‌های مهاجم حجم بالایی از ترافیک را به شبکه تزریق می‌کنند. این حجم بالای ترافیک در بازه زمانی مشخص حین رصد شبکه، قابل تشخیص است. به بیان ساده‌تر از طریق تشخیص ناهنجاری در ترافیک شبکه می توان با حملات DDoS با نرخ بالا مقابله نمود. اما در مقابل، slow-rate DDoS همانطور که از نام آن پیداست، حجم ترافیک بالایی را تولید نکرده و مکانیزم متفاوت و هوشمندانه‌تری در حمله دارد؛ درslow-rate DDoS با شناسایی نقطه ضعف‌های موجود در برخی پروتکل‌ها، مقادیر خاصی از سرآیند بسته¬ها در شبکه توسط ماشین‌های مهاجم دستخوش تغییر می‌شود. در واقع در این درسته حملات، با سوءاستفاده از ضعف‌های موجود در پروتکل‌ها، سرورها به صورت دائم مشغول پاسخگویی به ماشین‌های مهاجم شده و از پاسخگویی به کاربران واقعی بازمی‌مانند.

از جمله ابزارهای انجام این دسته از حملات می‌توان به Slowloris، SlowHTTPTest، R.U.D.Y، GoldenEye و H.U.L.K اشاره کرد. حملات slow-rate DDoS به سرورهای HTTP یکی از مهم‌ترین حملات در این دسته هستند که سرورهای thread-based بیشترین آسیب‌پذیری را در برابر آن‌ها خواهند داشت. پیام‌های درخواست و پاسخ در ارتباط بین ماشین درخواست دهنده و سرور HTTP، دارای قالب خاصی هستند و Slowloris یک نمونه از حمله‌هایی است که از قالب یاد شده سوءاستفاده کرده، و درخواست-های ناقص و بخش‌بخش را با نرخ بسیار پایین و در تعداد بسیار بالا به سمت سرور ارسال می‌کند؛ با ادامه این روند، ظرفیت سرور بر اثر پردازش تعداد بالایی از درخواست‌های غیر واقعی اشباع شده و درنتیجه سرور از پاسخگویی باز خواهد ماند.

در تشخیص حملات DDoS اغلب روش‌های معرفی شده، بر پایه استفاده از تکنیک‌های آماری و یا با استفاده از الگوریتم‌های حوزه یادگیری ماشین ارائه شده‌اند. حجم ترافیک حملات با نرخ پایین از نظر ظاهری تقریبا مشابه کاربران واقعی بوده و شناسایی این دسته از حملات نسبت به حملات با نرخ بالا، به نسبت پیچیده‌تر است؛ در نتیجه برای هریک از دو روش تکنیک‌های آماری و الگوریتم‌های یادگیری ماشین، نیاز به تولید اطلاعات دقیق‌تر خواهد بود. با توجه به رفتار و ماهیت این دسته حملات و همچنین دقت بالا و کاربرد وسیع الگوریتم‌های حوزه یادگیری ماشین، در این پژوهش از الگوریتم-های یاد شده در تشخیص حملات استفاده شده است [13]

۲-۲- شبکه‌های نرم‌افزار محور : شبکه‌های نرم‌افزار محور یا SDNعنوان یک معماری نسبتا جدید است که شبکه را قادر می‌سازد تا کلیه اعمال کنترلی شبکه به وسیله یک کنترلر مرکزی (نه لزوما یک کنترلر، بلکه به معنای مرکزیت کنترل در شبکه) مدیریت شود. تمرکز اصلی این معماری بر جداسازی صفحه کنترل از صفحه داده است. در پی این هدف، ادوات خاص منظوره موجود در معماری سنتی (سوییچ، مسیریاب و ...) با ادوات عام منظوره فاقد قدرت پردازشی خاص جایگزین می‌شوند. این ادوات ساده‌تر، توسط یک کنترلر مرکزی مدیریت شده و تنها اجرا کننده قوانین تولید شده توسط آن هستند و از خود توان پردازشی به‌خصوصی ندارند.

معماری SDN، یک معماری سه لایه است که لایه‌های برنامه‌های کاربردی، کنترل و زیرساخت را شامل می‌شود. به این ترتیب سه سطح عملیاتی مدیریت، کنترل و داده در این لایه‌های یاد شده در نظر گرفته می‌شوند. همچنین بین لایه‌ها، دو رابط شمالی[۴] و جنوبی[۵] وجود داشته که ارتباط بین این لایه¬ها را فراهم می‌کنند؛ بر این اساس صفحه‌های کنترل و برنامه‌های کاربردی از طریق رابط شمالی و صفحه‌های کنترل و زیرساخت نیز از طریق رابط جنوبی به یکدیگر متصل شده‌اند. به دلیل وجود این سطوح مجزا و رابط‌های بین آن‌ها، قابلیت انعطاف-پذیری شبکه بالا بوده و همین امر جایگزینی پروتکل‌های ارتباطی و ادوات P4 با پروتکل‌های موجود و ادوات عام‌منظوره را ساده‌تر کرده است.

کنترلر شبکه یک محصول نرم‌افزاری است که نقش اصلی را در پیاده‌سازی قوانین تولید شده توسط برنامه‌های کاربردی بر روی ادوات شبکه، بر عهده دارد و به نوعی بازوی عملیاتی شبکه محسوب می‌شود. با توجه به جداسازی صفحه داده از صفحه کنترل و عدم وجود وابستگی، در انتخاب نوع کنترلر محدودیتی وجود نداشته و در این راستا شرکت‌ها و انجمن‌های مختلف، محصولات مختلفی را تولید و عرضه کرده‌اند.OpenDayLight ، Ryu و ONOS برخی از کنترلر های متن‌باز[۶] موجود هستند. ONOS یک پروژه متن‌باز بوده که هدف آن ایجاد یک سیستم عامل شبکه مبتنی بر SDN، برای ارائه دهندگان خدمات ارتباطی است. این کنترلر برای مقیاس‌پذیری، کارایی و دسترسی بالا طراحی شده است، مبتنی بر زبان Java بوده، دارای

[4] Northbound API
[5] Southbound API
[6] Open source

رابط کاربری گرافیکی نیز می‌باشد. یکی از مهم‌ترین ویژگی‌های آن پشتیبانی مستقیم از ادوات P4 است .[14-16]

۲-۳- فناوری P4: همانطور که پیش تر اشاره شد، مرکزیت کنترل در شبکه‌هایSDN باعث شده تا این معماری به هدفی عالی برای انجام حملاتDDoS تبدیل شود. این حملات بخش‌های مختلف مرتبط با کنترلر شبکه را هدف قرار می‌دهند چراکه با از کار افتادن کنترلر شبکه، کل شبکه تحت تاثیر قرار خواهد گرفت. از سمتی دیگر، تشخیص و جلوگیری به موقع حملات، خود نیازمند اختصاص توان پردازشی به‌خصوصی است که در مقیاس یک شبکه می‌تواند بسیار بالا باشد. پیاده‌سازی روش‌های تشخیص و جلوگیری از حملات در کنترلر شبکه، به دلیل تحمیل سربار پردازشی به کنترلر، خود موجب کاهش کیفیت عملکرد آن می‌شود. این امر باعث ایجاد محدودیت‌هایی در ارائه روش‌های پیشنهادی در مقابله با حملات و یا پیچیدگی بیش از حد روش‌ها شده است. به همین دلیل محققان تصمیم گرفتند تا بخشی از پردازش های لازم را از کنترلر، به ادوات در سطح داده انتقال دهند. معرفی فناوری P4 و ادوات مبتنی بر آن، یکی از کارآمدترین روش‌های ارائه شده در این زمینه است.

محققان برای شبیه‌سازی شبکه‌های نرم افزار محور ابزارهای مختلفی استفاده می‌کنند که هر یک ویژگی‌های خاصی در اختیار کاربران خود قرار می‌دهند، اما به‌صورت عام، در اکثر محیط‌های آزمایشی و پژوهشی به دلیل انعطاف‌پذیری و سرعت بالا از شبیه‌ساز Mininet برای شبیه‌سازی شبکه‌های SDN استفاده شده است. Mininet یک شبیه‌ساز شبکه مبتنی بر خط‌فرمان[7] است، که شبکه‌ای از میزبان‌های مجازی[8]، سوئیچ‌ها، کنترلرها و لینک‌های بین آن‌ها را ایجاد می‌کند. دلیل فراگیر بودن این شبیه‌ساز در امکاناتی است که در اختیار کاربر قرار می‌دهد. Mininet در پیاده‌سازی این پژوهش به دلیل پشتیبانی مستقیم این شبیه‌ساز از سوییچ‌های P4 مورد استفاده قرار گرفته است .[17]

با معرفی زبان P4 و ادوات مبتنی بر آن این امکان فراهم شده است، تا بخشی از پردازش بسته‌ها در صفحه داده انجام شود. این قابلیت، محدودیت مطرح شده در حالت پیش‌فرض شبکه‌های SDN را برطرف می‌کند چراکه ادوات در این حالت، از توان پردازشی به‌خصوصی برخوردار هستند. قابلیت پردازشی ادوات P4 از طریق برنامه‌ریزی آن‌ها به‌وسیله زبان P4 و توسط متخصصان شبکه فراهم می‌شود. زبان P4، برخلاف زبان‌هایی مانند پایتون و C، یک زبان برنامه‌نویسی خاص منظوره است که جهت تعیین روند پردازش بسته‌ها در ادوات SDNطراحی شده و با استفاده از قابلیت‌های آن می‌توان عملکرد ادوات را مطابق با نیازها در حوزه کاری مورد نظر تعیین نمود. بر این اساس می‌توان بسیاری از الگوریتم‌ها و روش‌های استخراج اطلاعات از شبکه را در یک سوییچ P4 پیاده‌سازی و در تشخیص ناهنجای از آن‌ها استفاده نمود؛ در این صورت کنترلر تنها دریافت کننده این اطلاعات است و سربار پردازشی مربوطه حذف خواهد شد. با توجه به جزئیات عنوان شده، از قابلیت‌های زبان و ادوات P4 می‌توان در کاربردهای مختلف مانند تشخیص حمله، تنظیم بار و رصد شبکه استفاده نمود، تا حجم بالایی از بار پردازشی کنترلر کاسته شود.

زبان P4 این قابلیت را دارد که قالب‌های سرآیند[9] بسته‌ها را به صورت سفارشی[10] تعریف کرده و این سرآیندها را به صورت پویا از بسته تجزیه کند. P4 علاوه بر این، از جداول سفارشی‌سازی شده و سایر ساختارهای مرتبط با پردازش بسته مانند شمارنده‌ها، ثبات‌ها و تغییر سرآیند بسته‌ها نیز پشتیبانی می‌کند. این امر باعث شده زبان P4 کاملا مستقل از پروتکل باشد؛ به این معنا که در صورت استقرار پروتکل جدید در شبکه، به راحتی می‌توان برنامه P4 را برای پشتیبانی از مقادیر سرآیند پروتکل جدید تغییر داد. بنابراین یکی دیگر از ویژگی های زبان P4 قابلیت پیکربندی مجدد است، به بیان ساده‌تر می‌توان در صورت نیاز بجای خرید سخت‌افزار جدید، برنامه جدید P4 را در سوییچ آن پیاده‌سازی نمود.

هربرنامه P4 برای یک معماری خاص نوشته می‌شود که ماشین هدف (سوییچ) نیز بر اساس همان معماری طراحی شده است؛ این یعنی برنامه نوشته شده به زبان P4 به معماری دستگاهی که توسط تولیدکنندگان ارائه می‌شود وابستگی دارد. برنامه P4 که برای یک معماری خاص نوشته شده است، تنها در ماشین‌های هدفی که از همان مدل معماری استفاده می‌کنند، قابل حمل و پیاده‌سازی است و عملکرد و قابلیت‌های هریک از این ماشین‌ها متفاوت هستند. در عمل روش‌های مختلفی برای پیاده‌سازی سوییچ‌های P4 در شبکه وجود دارد، اما اکثرا در موارد پژوهشی از شبیه‌ساز Mininet همراه با سوییچ های BMv2 که بر اساس معماری V1model است، استفاده شده است .[18,19]

۲-۴- پژوهش‌های پیشین: در ادامه به بررسی روش بکارگیری سوییچ‌های P4 در این پژوهش جهت تشخیص و پیشگیری حمله، پرداخته خواهد شد.

[7] Command-line
[8] Virtual hosts
[9] Header
[10] Custom-made

Santos و همکاران[1] در سال ۲۰۱۹ در پژوهشی با توجه به اهمیت معماری SDN و وجود خطر حمله‌های جدید از نوع DDoS، سعی بر آن داشته‌اند تا چهار الگوریتم یادگیری ماشین شامل Random Forest، CART، SVM و MLP را در شرایط مختلف تحت آزمایش قرار داده تا میزان کارایی آن‌ها را در تشخیص حملات DDoS براساس مؤلفه‌های مختلف ارزیابی نمایند. بخشی از این پژوهش به بررسی مؤلفه‌ها و انتخاب ویژگی‌های مناسب برای حل این مسأله پرداخته شده که یکی از مراحل تأثیرگذار در نتیجه نهایی، در حل مسائل با استفاده از الگوریتم‌های یادگیری ماشین است. به عنوان نتیجه، تعدادی از ویژگی‌ها بر اساس میزان اهمیت آن‌ها در تشخیص حملات، به عنوان مؤثرترین ویژگی‌ها معرفی شده و بازدهی هر یک از چهار الگوریتم یاد شده، مورد ارزیابی قرار گرفته است. نکته قابل توجه در این پژوهش آن است، که در صورت پیاده‌سازی مرحله استخراج مقادیر ویژگی‌های الگوریتم یادگیری ماشین در کنترلر شبکه، سربار پردازشی بسیار زیادی به کنترلر شبکه تحمیل می‌شود. این خود یک گلوگاه برای کل شبکه است چراکه استخراج مقادیر ویژگی برای الگوریتم‌های یادگیری ماشین، نیازمند اندازه‌گیری دائمی مقادیر بوده و بخش قابل توجهی از توان پردازشی کنترلر را به خود اختصاص خواهد داد.

Diaz و همکاران[2] در سال ۲۰۲۰ یک چارچوب منعطف برای تشخیص حملات DDoS در شبکه‌های SDN با استفاده از الگوریتم-های یادگیری ماشین ارائه کردند، که در آن تمرکز بر روی حملات DDoS با نرخ پایین بوده و در آن مدلی ماژولار را با بکارگیری کنترلر ONOS، جهت تشخیص و جلوگیری از این دسته حملات مورد استفاده قرار داده‌اند. مدل ارائه شده در این پژوهش بسیار کارآمد بوده چراکه بخش‌های مختلف آن فاقد وابستگی به یکدیگر است و از این رو شاهد انعطاف‌پذیری بسیاری در حین پیاده‌سازی خواهیم بود.

در سال ۲۰۲۰ Sahoo و همکاران[3] با استفاده از الگوریتم‌های یادگیری ماشین و روش های کاربردی حوزه هوش مصنوعی، روشی برای بهبود تشخیص ترافیک حملات DDoS در شبکه‌های نرم‌افزار محور ارائه کردند. در این پژوهش الگوریتم اصلی طبقه‌بندی[11]، SVM بوده که جهت افزایش هرچه بیشتر دقت آن به صورت توأم با الگوریتم ژنتیک و KPCA بکارگرفته شده است. از الگوریتم ژنتیک جهت بهینه-سازی مولفه‌های الگوریتم SVM و از KPCA جهت کاهش ابعاد بردار ویژگی‌ها و همچنین از N-RBF برای کاهش زمان آموزش الگوریتم استفاده شده است. در نهایت مدل آموزش داده شده SVM+GA+KPCA به طبقه‌بندی داده‌ها خواهد پرداخت و درصورتی که داده‌های ترافیکی به عنوان حمله طبقه‌بندی شوند، سیستم وارد مرحله جلوگیری از حمله خواهد شد. بدیهیست بهبود کیفیت تشخیص الگوریتم‌های یادگیری ماشین، رابطه مستقیم با نتیجه نهایی و کیفیت تشخیص خواهد داشت اما چگونگی بهبود الگوریتم‌ها و از طرفی پیچیدگی و پردازش مورد نیاز آن‌ها در مسائل مختلف، متفاوت خواهد بود و امری غیرقطعی است.

Mohammadi و همکاران[4] در سال ۲۰۱۹ طی پژوهشی روشی را برای جلوگیری از حملات SYN flood در SDN معرفی کردند، که در قالب یک افزونه برای کنترلر عمل می‌کند. این نوع از حملات از قابلیت‌های موجود در پروتکل TCP که جهت اطمینان انتقال در شبکه طراحی شده است، سوءاستفاده کرده و هدف آنها اشباع‌سازی ظرفیت شبکه است. در این عنوان کنترلر شبکه، با استفاده از روش‌های آماری، نقش واسط را بین مبدا و مقصد بسته ایفا نموده و در صورت اطمینان از عدم وجود حمله، اقدام به نصب قانون در ادوات خواهد نمود. این روش، روشی مناسب برای جلوگیری از حمله و پیشگیری از وقوع خطا در تشخیص حمله، می‌باشد.

Musumeci و همکاران[5] در سال ۲۰۲۲ در پژوهشی سعی بر آن داشته‌اند تا با بکارگیری سوییچ‌های مبتنی بر P4و الگوریتم‌های یادگیری ماشین، جلوگیری از حملات DDoS را توسط خود سوییچ‌ها انجام دهند. در این روش از چهار الگوریتم SVM، Random Forest، KNN و ANN استفاده شده است. هریک از الگوریتم‌های یاد شده دارای تعدادی مولفه هستند، که در طول آزمایش‌های متعدد مورد ارزیابی قرار گرفته و بهترین مقادیر برای آن‌ها معرفی شده است. همانطور که اشاره شد، فرآیند استخراج مقادیر ویژگی برای کنترلر شبکه سربار پردازشی بالایی خواهد داشت، ویژگی اصلی در این پژوهش، استفاده از سوییچ‌های P4 برای استخراج مقادیر ویژگی‌های مورد نیاز الگوریتم‌های یادگیری ماشین است. در کنار این پیاده‌سازی دو معماری متفاوت برای استخراج ویژگی نیز معرفی شده است، که بر اساس آن سوییچ‌های مبتنی بر P4 مقادیر ویژگی‌های مورد نیاز الگوریتم‌های یادگیری ماشین را استخراج کرده و به کنترلر شبکه می-فرستند. با استفاده از این روش، کنترلر شبکه تنها دریافت کننده مقادیر ویژگی‌ها بوده و مجبور به اختصاص توان پردازشی به این بخش نخواهد بود. این امر باعث کاهش سربار پردازشی روی کنترلر و بهبود کارایی کل شبکه در تشخیص حمله شده که در نتایج حاصل از این پژوهش قابل مشاهده است.

[11] Classification

در سال ۲۰۱۹ Febro و همکاران[6] در پژوهش خود سعی در شناخت و پیشگیری نوع خاصی از حملات DDoS باعنوان SIP-Invite DDoS را داشته‌اند. در این عنوان تمرکز بر تشخیص و پیشگیری حمله در نزدیکی مبدا حمله است و برای این امر از قابلیت های زبان P4 بهره گرفته شده است. تشخیص حمله در نزدیکی مبدا موجب پیشگیری از هدر رفت پهنای باند و خسارت در طول مسیر خواهد شد. در این پژوهش پیاده‌سازی رصد ترافیک[12] شبکه در نزدیکی مبدا و تشخیص ناهنجاری‌های موجود، با بکارگیری سوییچ-های مبتنی بر P4 حاصل شده است. روند کلی کار از سه قسمت اصلی حسگرها، بخش مدیریت سیاست‌ها و بخش تشخیص حمله تشکیل شده است. هر پورت در سوییچ در واقع طوری برنامه¬ریزی شده است، که مانند یک حسگر عمل کرده و سرآیند بسته‌های ورودی را بررسی خواهد کرد. برای تشخیص و سیاست‌گذاری نیز کنترلر موظف است، تمام وظایف مربوط به مدیریت حسگرها و تشخیص حمله را انجام دهد. در این روش از ترکیب یک جدول (که بسته‌های SIP را نگهداری می‌کند) و یک شمارنده، تشخیص حمله انجام خواهد شد. کنترلر در صورت تشخیص حمله از طریق دستکاری در قوانین جدول سوییچ، عمل انسداد ترافیک حمله از پورت مورد نظر را انجام می‌دهد.

Khooi و همکاران[7] در سال ۲۰۲۰، با توجه به شرایط خاصی که حملات DDoS برای ارائه دهندگان خدمات ارتباطی به وجود آورده‌اند، با بکارگیری ادوات مبتنی بر P4 یک معماری توزیع‌شده درون شبکه‌ای برای تشخیص و پیشگیری از حملات ارائه کردند. معماری ارائه شده یک معماری دفاعی کاملا اتوماتیک است که در آن تعداد درخواست‌ها و پاسخ‌ها در شبکه، رصد و نگهداری خواهند شد. نقاط انجام این رصدها در شبکه، بسیار مهم بوده و در نتیجه نهایی و میزان توان پردازشی لازم برای انجام آن، نقش بسزایی دارد. در معماری معرفی شده، تعداد پیام‌های درخواست در مسیریابهای دسترسی[13] و تعداد پیام‌های پاسخ در مسیریاب‌های مرزی[14] رصد می‌شوند. پس از آن بر اساس نتایج به دست آمده تشخیص و عملیات لازم برای جلوگیری از حمله صورت می‌گیرد.

Balarezo و همکاران[8] در سال ۲۰۲۰ یک مدل حمله slow-rate DDoS را معرفی کردند و جنبه‌های مختلف آن را مورد بررسی قرار دادند. تمرکز محققان بر پروتکل ارتباطی TCP و مکانیزم کنترل ازدحام این پروتکل بوده و بر این اساس در ادامه ارائه این مدل از پروتکل Openflow و همچنین فناوری P4 استفاده شده است. ادوات مبتنی بر P4 از طریق P4Runtime با صفحه کنترل ارتباط برقرار می‌کنند که مبتنی بر gRPC/http2 عمل می‌کند و همچنین پروتکل ارتباطی ادوات مبتنی بر Openflow نیز TCP است؛ تمرکز این مدل نیز بر دو پروتکل یادشده می‌باشد. شناسایی ابعاد مختلف این مدل حمله، در طراحی سیستم تشخیص و جلوگیری از حملات slow-rate DDoS بسیار کاربردی خواهد بود.

در سال ۲۰۲۰ Dimolianis و همکاران[9] یک روش جلوگیری از حملات DDoS را با بکارگیری خاصیت برنامه‌پذیری صفحه داده که از طریق زبان P4 و ادوات مبتنی بر آن حاصل شده است، معرفی کردند. در این روش سوییچ‌های مبتنی بر زبان P4 بکارگرفته شده‌اند که برای تشخیص ناهنجاری، اطلاعات خاصی از سرآیند بسته‌ها را استخراج و بررسی کرده و در مراحل بعد از این اطلاعات در تشخیص ناهنجاری استفاده می‌شود. این سوییچ‌ها در نقاط حیاتی شبکه پیاده-سازی شده و نتایج حاصل از رصدهای خود را به سیستم پیشگیری از حمله خارجی، در قالب هشدارهایی اعلام خواهند نمود.

با توجه به موارد عنوان شده، در بیشتر موارد، تشخیص و پیشگیری از حملات DDoS نیازمند پردازش به‌خصوصی در شبکه است. میزان این پردازش بسته به روش تشخیص و نوع دریافت اطلاعات از شبکه، متفاوت خواهد بود. برای ارائه یک روش تشخیص و جلوگیری کارآمد، نیاز است تا اطلاعات خاصی از نقاط حیاتی در شبکه دریافت شده و جلوگیری از حمله نیز در نقطه‌ای از شبکه انجام شود که مانع هدر رفت پهنای باند و سایر منابع شبکه شود. این امر نیازمند آن است که حملات در ابتدایی‌ترین نقاط شروع حمله و با کمترین سربار پردازشی ممکن برای کنترلر، تشخیص داده شوند.

Panda و Badotra [10] در سال ۲۰۱۹ در یک پژوهش با استفاده از سیستم تشخیص نفوذ Snort سعی برآن داشته‌اند تا در سناریوهای مختلف و تحت حملات مختلف DDoS، میزان کارایی این سیستم تشخیص نفوذ را مورد ارزیابی قرار دهند. برای شبیه‌سازی، از شبیه‌ساز Mininet همراه با کنترلرهای ONOS و ODL استفاده شده است. طی آزمایشات انجام شده در این پژوهش، کنترلر ODL همراه با سیستم تشخیص نفوذ Snort، زمان کمتری را برای تشخیص حمله و همچنین زمان بیشتری را تا از کار افتادگی کامل کنترلر به همراه داشته است. شناسایی عملکرد کنترلرهای یاد شده همراه با ارزیابی عملکرد سیستم‌های تشخیص نفوذ، تاثیر بسزایی در شناخت بهتر مساله و ارائه راه حل‌های بهینه خواهد داشت.

[12] Traffic monitoring
[13] Access routers
[14] Edge routers

Mahrach و Haqiq [11]در سال ۲۰۲۰ سعی در ارائه یک مکانیزم جلوگیری از حملات syn-flood در شبکه های نرم‌افزار محور از طریق بکارگیری بخش داده با استفاده از قابلیت های فناوری P4 را داشته‌اند. در مکانیزم ارائه شده به دلیلی محدودیت میزان حافظه موجود در بخش داده، از تکنیک syn cookie استفاده شده است، که بدون ذخیره حالت می‌باشد و به فضای ذخیره حالات ارتباط TCP نیاز ندارد.

Simsek و همکاران [12] در سال ۲۰۱۹ در پژوهشی سعی کردند تا با بکارگیری قابلیت های زبان P4 از بخش داده برای تشخیص حملات انکار سرویس استفاده کنند تا از این طریق میزان بار و حجم کاری روی پردازنده مرکزی را کاهش دهند. پیش فرض این پژوهش بر آن است که بیشتر حملاتی که بر روی کنترلر انجام می‌شوند از طریق جعل آدرس فیزیکی و IP هستند.

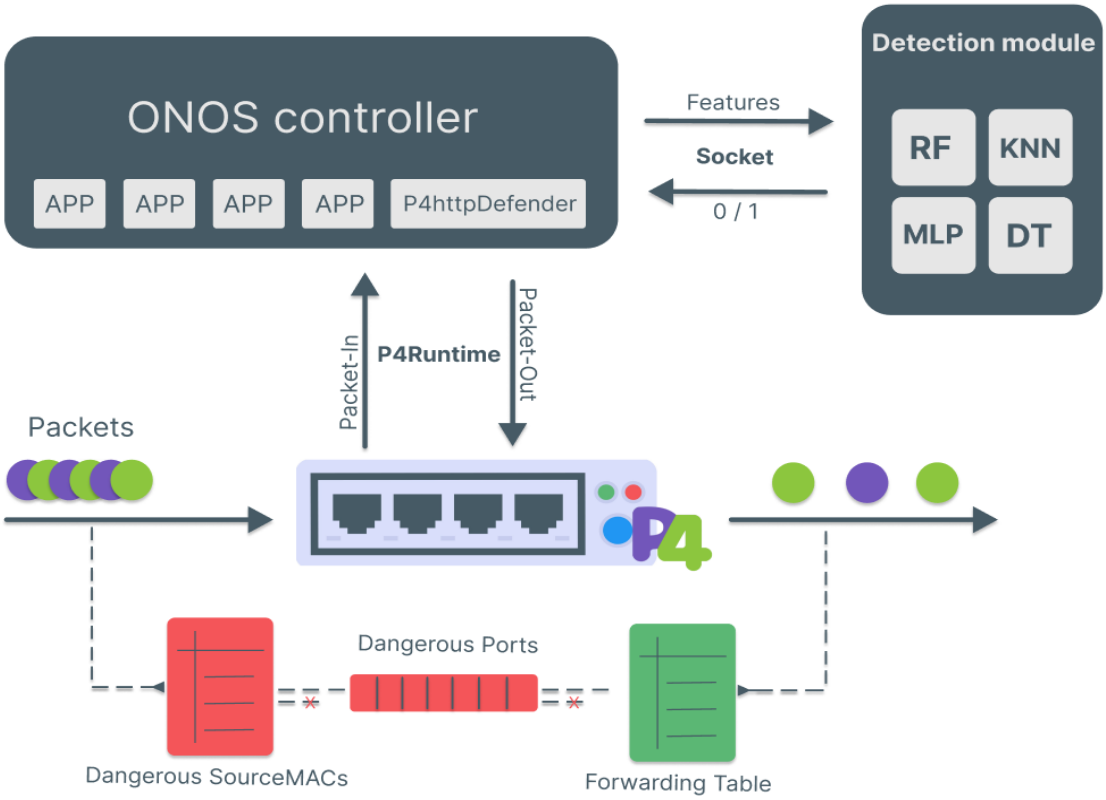
تمرکز این پژوهش بر روی تشخیص و جلوگیری از حملات slow-rate DDoS متمرکز بر سرورهای HTTP، در معماری شبکه های نرم‌افزار محور است؛ که برای این امر از کنترلر ONOS همراه با سوییچ‌های مبتنی بر P4 و الگوریتم‌های یادگیری ماشین استفاده شده است. در این راستا سوییچ‌های P4 اطلاعات شبکه را رصد کرده و مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین را استخراج می‌کنند. در صورت بروز شرایط مشکوک به حمله، ویژگی‌های حاصل از مرحله قبل همراه با اطلاعات ماشین‌های مشکوک به حمله، در قالب پیامی به سمت کنترلر شبکه ارسال می‌شوند. کنترلر شبکه در این حالت نقشی در استخراج ویژگی نداشته و از این جهت سربار پردازشی وجود نخواهد داشت. کنترلر پس از دریافت ویژگی‌ها، این مقادیر را به یک ماژول تشخیص حمله خارجی که از مدل‌های یادگیری ماشین تشکیل شده، ارسال می‌کند؛ بنابراین در تشخیص حمله و اجرای مدل-های یادگیری ماشین نیز کنترلر سربار پردازشی را به همراه نخواهد داشت. ماژول تشخیص حمله با استفاده از مقادیر ویژگی‌های استخراج شده توسط سوییچ‌های P4، تشخیص حمله را انجام داده و نتیجه را به کنترلر شبکه ارسال می‌کند. کنترلر شبکه بر اساس بازخوردی که از مرحله قبل به دست آمده، درصورت وجود حمله، قانونی را برای مسدودسازی پورت متصل به ماشین مهاجم، در سوییچ مورد نظر، نصب می کند. به این صورت هرگونه ترافیک حاصل از پورت متصل به ماشین مهاجم، توسط سوییچ P4 مسدود خواهد شد. لازم به ذکر است تأخیر ناشی از مبادله پیام بین ماژول تشخیص حمله و کنترلر بسیار ناچیز بوده و در مقابل سربار پردازشی حاصل از پیاده‌سازی الگوریتم‌های یادگیری ماشین در خود کنترلر، قابل چشم پوشی است؛ چراکه پیچیدگی الگوریتم‌های یادگیری ماشین و میزان تغییرات احتمالی آن‌ها با توجه به تغییرات حملات در آینده، نامشخص بوده و وجود یک ماژول تشخیص حمله خارجی و عدم وابستگی آن به کنترلر شبکه، انعطاف‌پذیری بیشتری را فراهم می‌آورد.

۳ – مدل تشخیص حمله ارائه شده

حملات slow-rate DDoS، با سوءاستفاده از مدخل‌های امنیتی موجود در پروتکل‌های ارتباطی، سیستم قربانی را مورد هدف قرار می-دهند. این دسته یکی از مخرب‌ترین حملات بر سرورهای HTTP هستند و سرورهای thread-based بیشترین آسیب‌پذیری را در برابر این حملات خواهند داشت. پیام‌های درخواست و پاسخ مبادله شده بین سرور و ماشین درخواست دهنده، دارای چرخه‌ای مشخص و قالبی خاص هستند، که ماشین مهاجم با ایجاد تغییر در روند معمول آن‌ها اقدام به حمله می‌کند. در پی این حملات، تعداد بالایی از درخواست‌ها با تغییر حالت معمول قالب پیام و یا چرخه یک پیام معمولی، به سمت سرور ارسال می‌شوند و منابع سرور بر اثر ناتوانی سرور در اتمام فرآیند پاسخگویی به درخواست‌ها، اشباع شده و سرور از پاسخگویی بازخواهد ماند. استفاده از تکنیک‌های یادگیری ماشین، روشی مناسب جهت تشخیص حملات یاد شده می‌باشد. در این پژوهش، با استفاده از قابلیت برنامه‌ریزی سوییچ‌های P4 و تکنیک‌های یادگیری ماشین، مدلی منعطف جهت تشخیص این دسته از حملات ارائه شده است. مدل ارائه شده در این پژوهش، از سه بخش تشکیل شده است که در شکل ۱ قابل مشاهده است. مطابق شکل ۱، مؤلفه‌های کنترلر، ماژول تشخیص حمله خارجی و سوییچ P4 به‌صورت هماهنگ با یکدیگر در تعامل هستند. راه ارتباطی بین ماژول تشخیص حمله و کنترلر ONOS، از طریق سوکت فراهم شده است؛ سوییچ‌های P4 و کنترلر ONOS نیز از طریق پروتکل ارتباطی P4Runtime با یکدیگر در ارتباط هستند.

در پیاده‌سازی ماژول تشخیص حمله، از چهار الگوریتم CART، KNN، RF و MLP جهت طبقه‌بندی استفاده شده است که توسط کتابخانه Scikit-learn پیاده‌سازی شده است[20] . تشخیص توسط این الگوریتم‌ها، بر اساس مقادیر موجود در دو مرحله آموزش الگوریتم و ارائه ویژگی به آن‌ها انجام می‌شود. در مرحله آموزش، از یک مجموعه داده مناسب با تعداد ۱۶۶۵۳۰ نمونه و تعداد ۳۳ ویژگی استفاده شده است، که هریک از ویژگی‌ها متناسب با رفتار حملات slow-rate DDoS در شبکه انتخاب شده است [21]. جدول ۱ نشانگر مجموعه ویژگی‌های بکار رفته در فرآیند آموزش الگوریتم‌های یاد شده می‌باشد.

پس از آموزش الگوریتم‌های یادگیری ماشین با تنظیم مؤلفه‌های مناسب و مجموعه داده یاد شده، مدل‌های یادگیری ماشین حاصل در



(شکل -۱): شمای کلی مدل تشخیص حمله ارائه شده

ماژول تشخیص حمله، آماده تشخیص حملات خواهند بود. بدین صورت تنها کافیست تا مقادیر ویژگی‌های مورد نظر به مدل‌های یادگیری ماشین تحویل داده شوند، تا عملیات تشخیص توسط آن‌ها صورت گیرد. با بکارگیری زبان P4، فرآیند استخراج مقادیر ویژگی‌ها در سوییچ پیاده‌سازی خواهد شد. بنابراین مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین، جهت تشخیص، به صورت مستقیم توسط سوییچ‌های P4 و در بخش داده استخراج می‌شوند. سوییچ‌های P4، مطابق روند پردازشی تعیین شده در برنامه P4، همزمان علاوه بر مسیریابی و هدایت بسته‌ها، به صورت مداوم درحال رصد بسته‌های HTTPموجود در شبکه می‌باشند. در فرآیند استخراج مقادیر ویژگی-ها، سوییچ‌های P4، مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین جهت تشخیص حمله را استخراج می‌کنند. استخراج این مقادیر بر اساس مجموعه داده یادشده و بسته‌های مبادله شده مبتنی بر پروتکل HTTP می‌باشد. در مواقعی که شبکه تحت حمله باشد، سرور HTTPدر اثر حمله، پس از بازه زمانی مشخص، از پاسخگویی به درخواست‌ها باز خواهد ماند. در چنین شرایطی، اختلاف بین درخواست‌های HTTP و پاسخ‌های بازگشته از سمت سرور، به صورت مدام افزایش خواهد یافت. سوییچ‌های P4 در این مدل طوری برنامه-ریزی شده‌اند، که در صورت تجاوز این مقدار از بازه‌ای خاص، برای سوییچ شرایط مشکوک به حمله تلقی خواهد شد. در صورتی که سوییچ بر اساس بسته‌های مبادله شده HTTP، شرایط مشکوک را تشخیص دهد، به بسته برچسب مشکوک به حمله اختصاص خواهد داد، و بسته به همراه مقادیر ویژگی‌ها و مشخصات ماشین‌های مشکوک، جهت تشخیص حمله به کنترلر ONOS ارسال خواهد شد. در کنترلر ONOS، درصورتی که بسته برچسب مشکوک را به همراه داشته باشد، توسط برنامه تعبیه شده در کنترلر، فرآیندی آغاز می‌شود که در آن ویژگی‌های میزبان مشکوک، به ماژول تشخیص حمله ارسال خواهد شد. ماژول تشخیص حمله، از طریق مدل‌های یادگیری ماشین، اقدام به طبقه‌بندی ویژگی‌های ارائه شده خواهد نمود. نتیجه حاصل از بخش تشخیص حمله، یک خروجی دودویی است. درصورت پیشبینی حمله از سوی مدل‌های یادگیری ماشین، کنترلر ONOS اقدام به نصب قانونی مرتبط با فرآیند مسدودسازی پورت متصل به ماشین مهاجم خواهد نمود. در روند مسدودسازی، کنترلر ONOS با در اختیار داشتن نتیجه تشخیص و اطلاعاتی از ماشین‌های مهاجم، در جدول مخصوص مسدودسازی ترافیک حمله از سوییچ متصل به مبدا حمله، قانونی را نصب کرده تا مانع ارسال و دریافت اطلاعات از پورت متصل به ماشین

(جدول -۱): ویژگی‌های بکار گرفته شده در روند تشخیص حمله

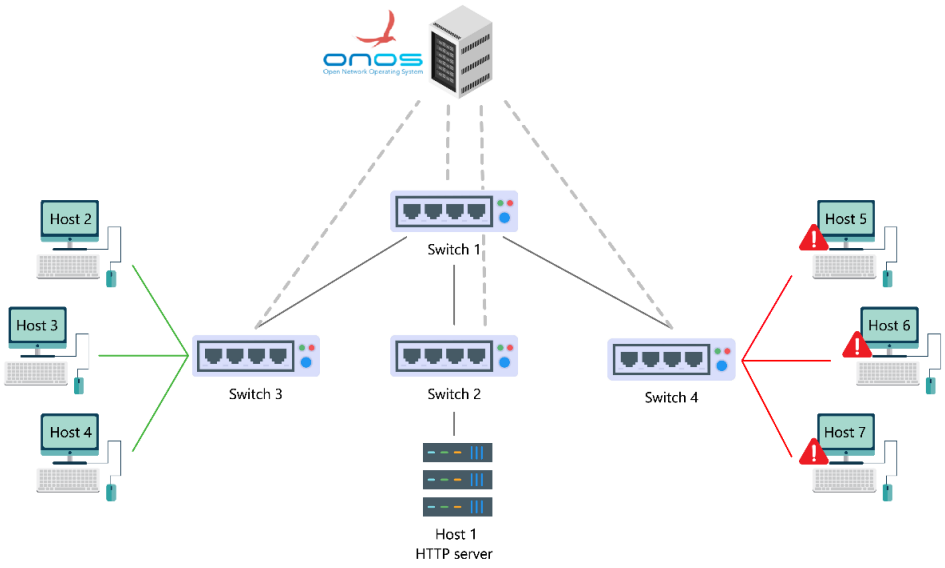
شماره	ویژگی	شماره	ویژگی	شماره	ویژگی
۱	Flow Duration	۱۲	Fwd Packet Length Std	۲۳	Fwd IAT Min
۲	Total Fwd Packets	۱۳	Bwd Packet Length Std	۲۴	Fwd IAT Max
۳	Total Backward Packets	۱۴	Fwd Packets/s	۲۵	Bwd IAT Max
۴	Total Length of Fwd Packets	۱۵	Bwd Packets/s	۲۶	Fwd IAT Std
۵	Total Length of Bwd Packets	۱۶	Flow Bytes/s	۲۷	Bwd IAT Std
۶	Fwd Packet Length Mean	۱۷	Flow Packets/s	۲۸	Flow IAT Mean
۷	Bwd Packet Length Mean	۱۸	Fwd IAT Mean	۲۹	Packet Length Mean
۸	Bwd Packet Length Max	۱۹	Bwd IAT Mean	۳۰	Min Packet Length
۹	Bwd Packet Length Min	۲۰	Fwd IAT Total	۳۱	Max Packet Length
۱۰	Fwd Packet Length Max	۲۱	Bwd IAT Total	۳۲	Packet Length Variance
۱۱	Fwd Packet Length Min	۲۲	Bwd IAT Min	۳۳	Packet Length Std

مهاجم شود. به این صورت تمام بسته‌های ورودی در سوییچ، در صورت وجود تطابق اطلاعات بسته در جدول یادشده، مسدود خواهند شد و مابقی بسته‌ها طبق روند معمول به مسیر مناسب هدایت خواهند شد.

کنترلر در این مدل سربار پردازشی بسیار کمتری را متحمل خواهد شد، چراکه تنها واسط ارتباطی بین سوییچ‌های P4 و ماژول تشخیص حمله بوده و در روند تشخیص حمله با سربار پردازشی کمتری مواجه خواهد بود. به دلیل عدم وجود وابستگی بین بخش تشخیص حمله و کنترلر ONOS، انعطاف‌پذیری در این مدل بالا است. در صورت نیاز به هرگونه تغییرات احتمالی در بخش تشخیص حمله، نیاز به هیچگونه تغییر در کنترلر نبوده؛ این ویژگی در پیاده‌سازی الگوریتم‌های بهینه‌تر، پیاده‌سازی روش‌های متفاوت برای تشخیص و درنتیجه بهبود کیفیت تشخیص حملات، بسیاری از محدودیت‌ها را برطرف خواهد نمود.

۴- شبیه سازی شبکه و حمله

روش تشخیص حمله ارائه شده در این پژوهش، با استفاده از شبیه-ساز Mininet پیاده‌سازی شده است. جزئیات توپولوژی شبکه در شکل ۲ به تصویر کشیده شده است. در این شبیه‌سازی از چهار سوییچ P4 استفاده شده است، که هریک از نوع BMv2 و مبتنی بر معماری V1model می‌باشند. علاوه بر این، تعداد ۷ ماشین میزبان در شبکه پیاده سازی شده است، که یک ماشین میزبان در شبکه نقش سرور HTTP را ایفا کرده و سایر میزبان‌ها درخواست‌هایی را به سمت سرور ارسال می‌کنند. در این میان، سه ماشین میزبان اقدام به تولید ترافیک سالم نموده، و سه میزبان دیگر نیز با استفاده از ابزار SlowHTTPTest، اقدام به انجام حملات slow-rate DDoS خواهند نمود. در هریک از سوییچ‌های P4، دو جدول تعبیه شده است،



(شکل-۲): توپولوژی شبکه شبیه‌سازی شده

که یک مورد برای مسیریابی و دیگری برای مسدودسازی ترافیک حمله بکار گرفته می‌شود. در نهایت به جهت جلوگیری از تکنیک‌های جعل آدرس، با استفاده از یک ثبات، پورت‌های ورودی مخرب نیز مسدود خواهند شد. همانطور که اشاره شد، استخراج مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین توسط سوییچ‌های P4، و براساس روند تعیین شده برای آن‌ها در فایل برنامه P4، انجام خواهد شد.

در این پژوهش به جهت مقایسه تاثیر استفاده از توان پردازشی ادوات برنامه‌ریزی شده P4، با حالت مبتنی بر استاندارد Openflow، دو سناریو پیاده‌سازی شده است، که در شکل ۳ و شکل ۴ قابل مشاهده است.

در سناریوی اول، سوییچ‌های برنامه‌ریزی شده P4 وظیفه هدایت بسته‌ها، استخراج ویژگی، تشخیص شرایط مشکوک به حمله و جلوگیری از ترافیک حملات را بر عهده دارند. این در حالی است که در سناریوی دوم، سوییچ تنها وظیفه هدایت بسته‌ها و جلوگیری از ترافیک حملات را بر عهده داشته، و تشخیص شرایط مشکوک به حمله و استخراج مقادیر ویژگی توسط کنترلر انجام خواهد گرفت.

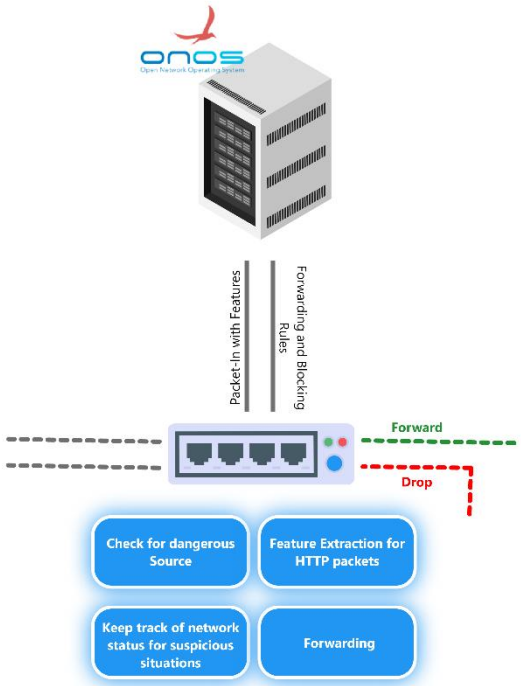
۵- ارزیابی نتایج به دست آمده

۵-۱- ارزیابی عملکرد مدل‌های یادگیری ماشین: جدول ۲ نمایانگر مقادیر ویژگی‌ها و مولفه‌هایی است که مدل‌های یادگیری ماشین براساس آن آموزش داده شده‌اند. انتخاب مقادیر مولفه‌ها، با در نظر گرفتن نتایج حاصل از پژوهش‌های پیشین و ارزیابی مقادیر مختلف بر روی مجموعه داده مورد استفاده در این پژوهش، انجام گرفته است.

جزئیات مقادیر مرتبط با زمان آموزش الگوریتم‌ها، در جدول ۳ موجود است. با توجه به مقادیر به دست آمده، در میان چهار مدل یادگیری ماشین مدل KNN به دلیل سازوکار متفاوت، فاقد زمان آموزش بوده و مقدار زمان بسیار پایینی را نشان خواهد داد، که تنها در آماده‌سازی مدل توسط سیستم سپری شده است.

(جدول-۲): پارامترهای تنظیم شده در آموزش مدل‌های یادگیری ماشین

مدل	پارامترهای تنظیم شده در مدل
MLP	hidden_layer_sizes = (2,10) , activation = relu
KNN	n_neighbors = 5 , weights = uniform
Random Forest	n_estimators = 10 , criterion = gini , max_features = None
Decision Tree (CART)	criterion = gini , splitter = best , max_depth = None

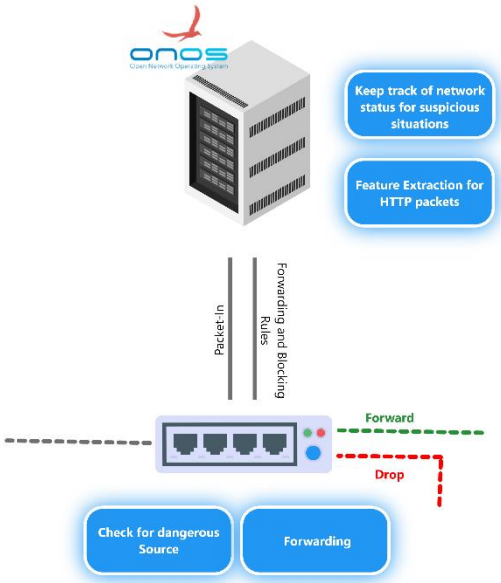


(شکل-۳): نمایی از نحوه عملکرد سناریو اول (با برنامه‌ریزی سوییچ‌های P4)

با صرف نظر از KNN، مدل Decision Tree (CART) کمترین زمان آموزش و مدل Random Forest با حدود ۱۶ ثانیه بیشترین زمان آموزش را به همراه داشته است. در این میان، می‌توان الگوریتم Decision Tree (CART) را متعادل‌ترین الگوریتم معرفی نمود، که زمان آموزش آن زیر ۳ ثانیه می‌باشد. همچنین در حالت کلی، بیشترین و کمترین زمان آموزش را نیز به ترتیب الگوریتم‌های Random Forestو KNN به همراه داشته‌اند.

در ارزیابی عملکرد و دقت مدل‌های یاد شده، از مقیاس‌های AUC-ROC score، Recall، Precision، Accuracy و F1-Score استفاده شده است. نحوه محاسبه هریک از این مقادیر در روابط زیر شرح داده شده است. نمونه‌های مثبتی که مدل یادگیری ماشین آن‌ها را به درستی، مثبت تشخیص داده است با عنوان TP[15] شناخته می‌شوند. نمونه‌های منفی که مدل یادگیری ماشین آن‌ها را به اشتباه، مثبت تشخیص داده است با عنوان FP[16] شناخته می‌شوند. همچنین نمونه‌های منفی که مدل یادگیری ماشین آن‌ها را به درستی، منفی تشخیص داده است با عنوان TN[17] شناخته می‌شوند. و در نهایت

[15] True Positives
[16] False Positives
[17] True Negatives



(شکل-۴): نمایی از نحوه عملکرد سناریوی دوم (مطابق رفتار Openflow)

نمونه‌های مثبتی که مدل یادگیری ماشین آن‌ها را به اشتباه، منفی تشخیص داده است نیز با عنوان FN[18] شناخته می‌شوند. بر اساس توضیحات پیشین، Precision کسری از نمونه‌های مثبت واقعی از میان کل نمونه‌هایی است که توسط مدل، مثبت تشخیص داده شده‌اند. پس از آن، Recall، کسری از نمونه‌های مثبت تشخیص داده شده توسط مدل یادگیری ماشین، از کل نمونه‌های مثبت واقعی می‌باشد. معیار Accuracy نشانگر تعداد نمونه‌هایی است که توسط مدل یادگیری ماشین به درستی طبقه‌بندی شده‌اند. معیار AUC-ROC scoreمیزان کارایی و مؤثر بودن مدل را بر اساس نسبت FPR[19] به TPR[20] را نشان می‌دهد و یک معیار اندازه گیری از دقت کلی مدل یادگیری ماشین است. معیار F1-Score، بیانگر میانگین دو معیار Precision و Recall می‌باشد. این معیار اندازه‌گیری نشانگر آن است که مدل یادگیری ماشین بر اساس مجموعه داده موجود، چه تعداد نمونه را به درستی طبقه‌بندی کرده است. معیارهای یاد شده از طریق روابط ۱ تا ۶ قابل محاسبه هستند.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

[18] False Negatives
[19] False Positive Rate
[20] True Positive Rate

$$Recall = \frac{TP}{TP + FN} \quad (۲)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (۳)$$

$$FPR = \frac{FP}{TN + FP} \quad (۴)$$

$$TPR = \frac{TP}{TP + FN} \quad (۵)$$

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (۶)$$

(جدول-۳): زمان آموزش الوریتم‌های یادگیری ماشین (میلی ثانیه)

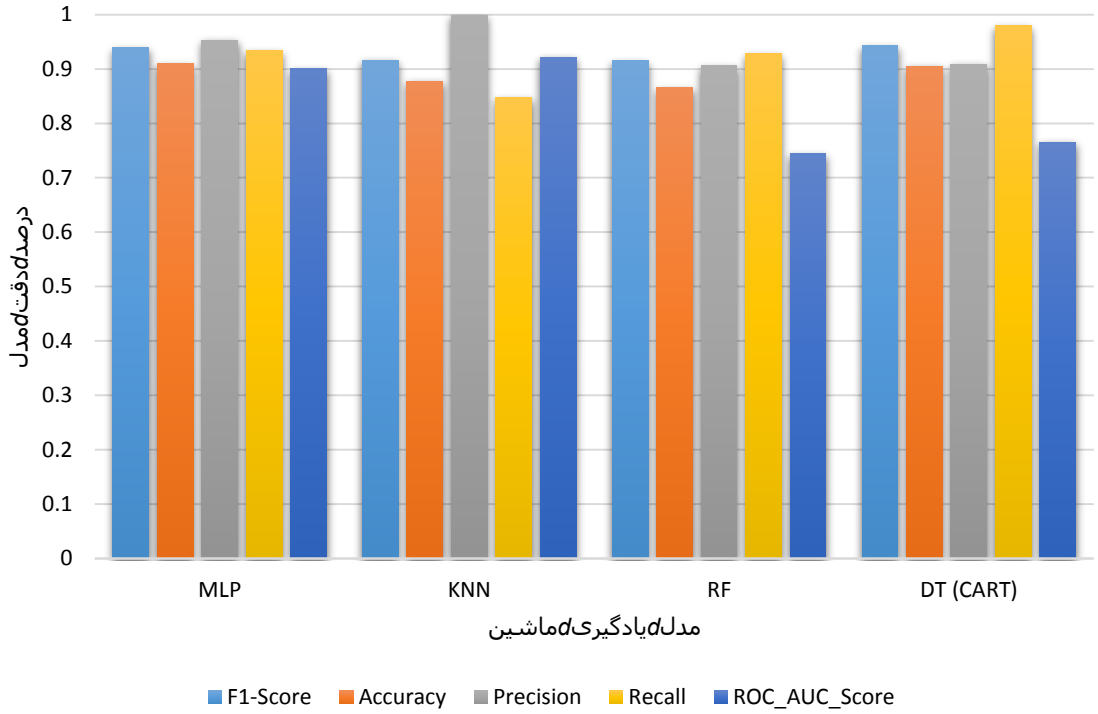
مدل یادگیری ماشین	کمینه زمان آموزش	میانگین زمان آموزش	بیشینه زمان آموزش
MLP	۴۵۵۱	۸۵۷۵/۵	۱۰۳۷۰
KNN	۱۶	۱۷	۱۸
Random Forest	۱۵۴۲۴	۱۵۹۸۵/۳	۱۶۶۲۹
CART	۲۶۰۶	۲۷۳۹/۳	۲۸۸۳

در ادامه پس از آماده‌سازی ماژول تشخیص حمله، شبکه مطابق توضیحات پیشین، مورد حمله قرار گرفته و مدل‌های موجود با در اختیار داشتن ویژگی‌های استخراج شده توسط سوییچ‌های P4، اقدام به طبقه‌بندی داده‌ها نموده‌اند. نتایج حاصل از ارزیابی مدل‌ها با معیار-های یادشده در جدول ۴ و نمودار شکل ۵ نمایش داده شده است.

بر اساس مقادیر حاصل، مدل KNN بهترین عملکرد را در معیار AUC ROC فراهم نموده و پس از آن به ترتیب MLP، Decision Tree (CART) و Random Forest قرار خواهند گرفت. در معیارRecall ، به ترتیب دو مدل Decision Tree (CART) و MLP نمونه‌های مثبت بیشتری را در میان تشخیص‌های خود به همراه داشته‌اند. بر اساس معیار Precision به دست آمده، مدل‌های MLPو KNN تعداد نمونه‌های مثبت بیشتری را از کل نمونه‌های مثبت موجود تشخیص داده‌اند. با توجه به معیار Accuracy و F1-Score، دو مدل MLP و Decision Tree (CART) بهترین دقت را در بین چهار مدل یاد شده به همراه داشته‌اند. لازم به ذکر است، مدل KNN به دلیل سازوکار متفاوتی که داراست، در تشخیص حمله و به طور کلی کاربردهایی که حساس به زمان هستند، مناسب نمی-باشد؛ چراکه برای هر نمونه در حین طبقه‌بندی، تمام نمونه‌های موجود در مجموعه‌داده را مورد بررسی قرار خواهد داد (به دلیل نداشتن مرحله آموزش) که این امر در حین اجرا، سربار زمانی ایجاد خواهد نمود. البته که در یافتن پاسخ و صرفه‌جویی در زمان تست و ارزیابی اولیه مسائل، یک نماینده مناسب از الگوریتم‌های حوزه یادگیری ماشین می‌باشد.

(جدول-۴): میزان دقت مدل‌های یادگیری ماشین در تشخیص حملات

مدل یادگیری ماشین	AUC ROC	Recall	Precision	Accuracy	F1-Score
MLP	۰/۹۰	۰/۹۳	۰/۹۵	۰/۹۱	۰/۹۳
KNN	۰/۹۲	۰/۸۴	۰/۹۹	۰/۸۷	۰/۹۱
Random Forest	۰/۷۴	۰/۹۲	۰/۹۰	۰/۸۶	۰/۹۱
Decision Tree (CART)	۰/۷۶	۰/۹۷	۰/۹۰	۰/۹۰	۰/۹۴



(شکل-۵): نمودار ارزیابی عملکرد مدل‌های یادگیری ماشین موجود در ماژول تشخیص حمله

۵-۲- ارزیابی زمان تشخیص حمله: مطابق توضیحات پیشین، جهت مقایسه عملکرد و تاثیر استفاده از قابلیت برنامه‌ریزی سوییچ‌های P4، دو سناریو شبیه‌سازی شده است که سناریوی اول بر اساس برنامه ریزی سوییچ‌های P4 و سناریوی دوم براساس رفتار Openflow می-باشد. همانطور که اشاره شد، در هریک از سناریوها، سه میزبان با استفاده از ابزار SlowHTTPTest اقدام به حمله به سرور HTTP خواهند نمود. نرخ ارسال حمله بر اساس follow-up data per second می‌باشد. بر این اساس چهار نرخ حمله در نظر گرفته شده است که مطابق جدول ۴ می‌باشد. میزبان‌های مهاجم بر اساس نرخ‌های گفته شده اقدام به انجام حملات slow-rate DDoS خواهند نمود و در هریک از سناریوها، پس از تشخیص و پیشگیری از حملات، زمان تشخیص محاسبه خواهد شد. جزئیات مربوط به زمان تشخیص حمله در سناریوهای اول و دوم به ترتیب در جدول ۵ و جدول ۶ موجود است. نرخ حمله در این جداول بر اساس follow-up data per second می‌باشد.

بر اساس مقادیر زمان تشخیص در جدول ۵، مدل Decision Tree (CART) با حدود ۱۹ ثانیه، کمترین زمان تشخیص را در بین مدل‌ها به همراه داشته و این زمان تشخیص پایین، در نرخ حملات کندتر نیز مشاهده می‌شود؛ بیشینه مقدار زمان تشخیص آن حدود ۲۴ ثانیه بوده و این در حالی است، که بیشنیه مقدار مدل Decision Tree (CART) نیز از بیشینه مقدار سایر مدل‌ها کمتر می‌باشد. پس از آن، مدل MLP بهترین زمان تشخیص کلی را به‌همراه داشته، طوری که در سه بازه نرخ حمله نهایی مدت زمان تشخیص حمله آن حدود ۲۱ ثانیه بوده است و در بدترین حالت نیز با حدود ۲۶ ثانیه، عملکردی میانه، از نظر بیشینه زمان هر چهار مدل به همراه داشته است. پس از دو مدل یاد شده، در ارزیابی مدل KNN و RF می‌توان زمان تشخیص مدل RF را بهتر از KNN دانست، چراکه KNN بیشینه زمان تشخیص را درمیان این چهار مدل به همراه داشته و نیز در نرخ حمله-های متفاوت RF زمان تشخیص کمتری را داشته است. این مورد به دلیل سازوکار مدل KNN است که پیش‌تر به آن اشاره شد و همانطور که انتظار می‌رفت، این سازوکار در زمان تشخیص حملات تاثیرگذار بوده است.

(جدول-۵): زمان تشخیص حمله در سناریوی اول بر حسب میلی ثانیه (با برنامه‌ریزی سوییچ‌های P4)

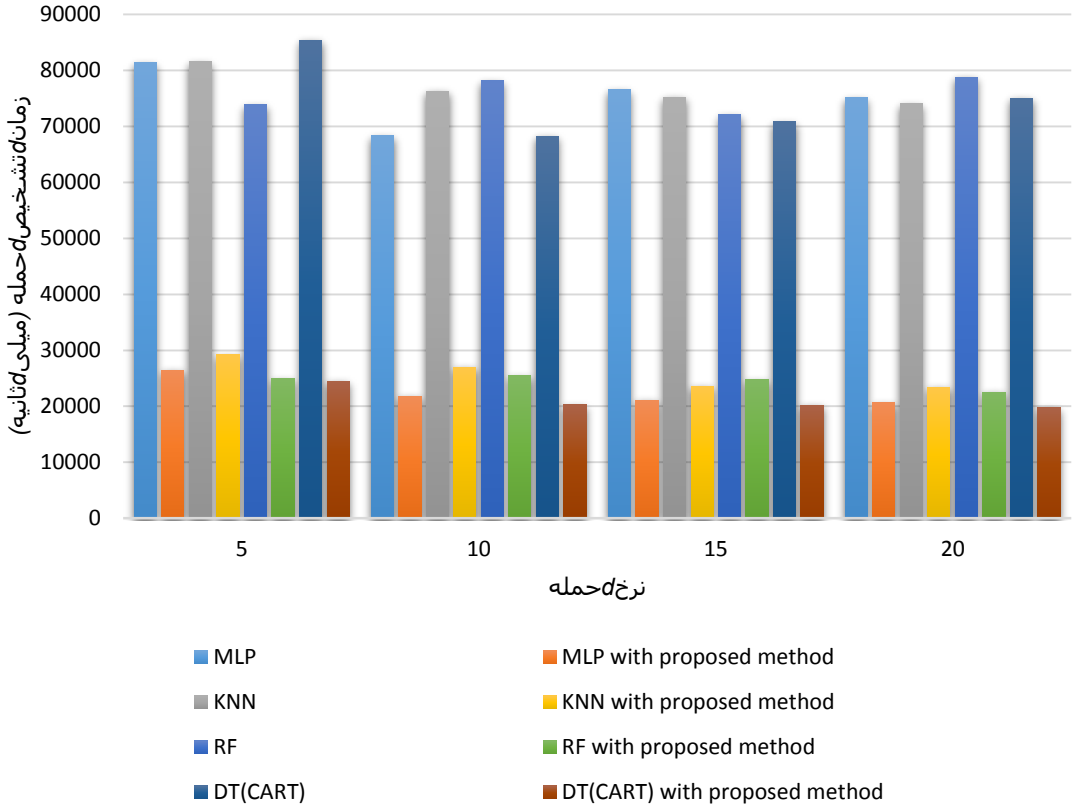
مدل	نرخ ۵	نرخ ۱۰	نرخ ۱۵	نرخ ۲۰
MLP	۲۶۴۸۹/۸۳	۲۱۸۰۸/۱۶	۲۱۱۰۵/۲	۲۰۶۷۷/۲
KNN	۲۹۲۱۴/۲	۲۷۰۱۰/۴۶	۲۳۶۰۱/۴	۲۳۳۸۵/۹
RF	۲۵۰۲۶/۲۶	۲۵۴۳۷/۲	۲۴۷۷۳/۴۶	۲۲۵۱۷/۹۳
CART	۲۴۴۸۸/۰۳	۲۰۳۰۶/۹۳	۲۰۱۸۲/۳۳	۱۹۸۰۷/۶۶

اما در سناریوی دوم، مطابق جدول ۶، زمان تشخیص سیستم اختلاف بسیاری دارد که نشانگر تاثیر استفاده از سوییچ‌های P4 و استفاده از توان پردازشی سوییچ‌ها می‌باشد. مطابق جدول ۵، در سناریوی دوم کمینه مدت زمان تشخیص حمله، با اختلاف کمی توسط مدل Decision Tree (CART) انجام پذیرفته است. حال آنکه برخلاف سناریوی اول، بیشینه زمان تشخیص نیز متعلق به خود مدل Decision Tree (CART) بوده، هرچند که بیشینه زمان تشخیص سه مدل دیگر نیز با این مقدار اختلاف اندکی دارند. پس از Decision Tree (CART)، مجدد می‌توان MLP را به عنوان مدل دوم با کمترین زمان تشخیص حمله در حالت کلی دانست؛ چراکه از جهت زمان تشخیص حمله در تمام نرخ‌های حمله، مشابه Decision Tree (CART) بوده و نتایج این دو مدل بسیار نزدیک به یکدیگر است و اختلاف کمی دارند. از میان دو مدل KNN و RF نیز با توجه به آن که بیشینه زمان تشخیص KNN با بیشینه زمان تشخیص RF اختلاف بیشتری داشته و در سایر قسمت‌ها نزدیک به یکدیگر بوده‌اند، می‌توان RF را دارای عملکرد بهتری، نسبت به KNN عنوان کرد. توضیحات عنوان شده در نمودار شکل ۶ به تصویر کشیده شده است.

(جدول-۶): زمان تشخیص حمله در سناریوی دوم بر حسب میلی ثانیه (مطابق رفتار Openflow)

مدل	نرخ ۵	نرخ ۱۰	نرخ ۱۵	نرخ ۲۰
MLP	۸۱۴۳۰/۵	۶۸۳۵۰/۴	۷۶۵۳۳/۵۳	۷۵۰۶۵/۰۶
KNN	۸۱۵۸۷/۷۶	۷۶۳۱۱/۲۳	۷۵۱۸۷/۷۶	۷۴۰۹۶/۶۳
RF	۷۳۸۳۹/۳۳	۷۸۱۲۶/۷	۷۲۰۴۷/۰۶	۷۸۸۰۸/۹
CART	۸۵۴۱۸/۷۶	۶۸۳۰۹/۱	۷۰۷۹۳/۱۳	۷۵۰۴۵/۰۳

براساس نتایج به‌دست آمده از نمودار شکل، ۶ می‌توان نتیجه گرفت که در سناریوی اول با افزایش نرخ، در حملات slow-rate DDoS، یا به بیان دیگر با افزایش فاصله زمانی بین درخواست‌ها از سرور HTTP، مدت زمان تشخیص حمله کاهش پیدا خواهد کرد. در مقایسه دو سناریوی اول و دوم، با اختلاف حدود ۶۰ ثانیه‌ای سناریوی اول در تمامی موارد عملکرد بهتری از خود نشان داده است. نتایج حاصل نشانگر میزان تاثیر بکارگیری فناوری P4 در تشخیص حملات موجود در شبکه‌های SDN می‌باشد. کاهش زمان تشخیص در سناریوی اول به دلیل انجام پردازش‌های اولیه و لازم در تشخیص حملات و استخراج مقادیر ویژگی‌های مورد نیاز مدل‌های یادگیری ماشین توسط سوییچ‌های P4 است، که باعث شده بخش قابل توجهی از فرآیند تشخیص حمله بدون نیاز به پردازش‌های کنترلر انجام شود.



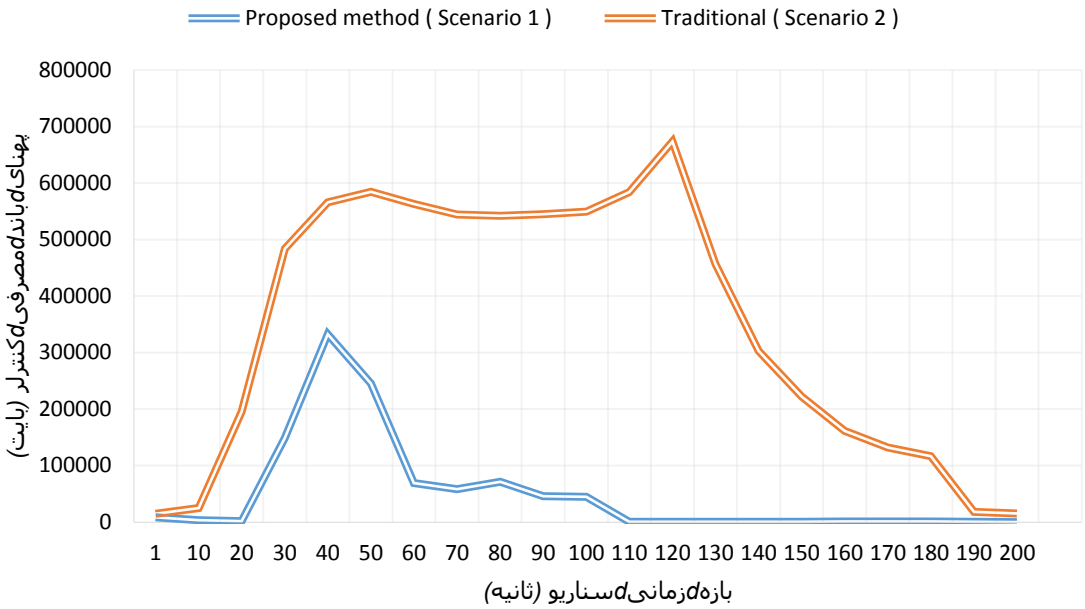
(شکل-۶): نمودار مقایسه زمان تشخیص حمله در سناریوی اول و دوم

۵-۳- ارزیابی مصرف پهنای باند: در این بخش به بررسی پهنای باند مصرفی کنترلر پرداخته خواهد شد. همانطور که اشاره شد، تمرکز این پژوهش بر روی حملات slow-rate DDoS و از نوع slowloris است. و در این نوع حملات، ترافیک پروتکل TCP و HTTP بسیار بالا خواهد بود چراکه تعداد درخواست‌های بالا اما به‌صورت ناقص و بخش‌بخش توسط مهاجم از سرور درخواست خواهد شد.

در سناریوی اول که از سوییچ‌های P4 جهت بررسی بسته‌ها و تشخیص شرایط مشکوک به حمله استفاده شده است، حجم بالایی از حملات در خود سوییچ مسیریابی شده و به کنترلر فرستاده نخواهد شد، مگر آنکه شرایط حساس و مشکوک به حمله پدید آید. نمودار شکل ۶، میزان مصرف پهنای باند کنترلر ONOS را، طی حمله و فرآیند تشخیص حمله سناریوهای اول و دوم نشان می‌دهد.

در سناریوی دوم سوییچ‌های P4، هیچ نقشی در روند تشخیص حمله و یا تشخیص شرایط مشکوک به حمله نداشته و به جهت نیاز فرآیند استخراج مقادیر ویژگی، تمامی بسته‌ها به کنترلر فرستاده شده تا مقادیر ویژگی‌های لازم استخراج شوند. همچنین کنترلر باید دید کلی نسبت به وضعیت شبکه و شرایط آن داشته باشد، تا در حین حمله بتواند ویژگی‌هایی که استخراج کرده است را جهت تشخیص حمله، تحویل مدل‌های یادگیری ماشین دهد. با توجه به نمودار موجود در شکل ۶، مصرف پهنای باند در سناریوی اول بسیار کمتر از سناریوی دوم بوده است. می‌توان دریافت که با توجه به بکارگیری توان پردازشی سوییچ‌های P4 در سناریو اول، به دلیل کاهش مدت زمان تشخیص حمله و همچنین انجام برخی پردازش‌های لازم به‌صورت محلی توسط سوییچ P4، میزان پهنای باند مصرفی بسیار کمتر از حالت موجود در سناریوی دوم می‌باشد. بر این اساس به دلیل انجام پردازش‌های ابتدایی به‌صورت محلی در سوییچ، بسته‌ها با تاخیر زمانی بیشتری، نسبت به سناریوی دوم، به کنترلر ارسال شده‌اند که موجب کاهش مصرف پهنای باند کنترلر شده است. از طرفی با توجه به زمان تشخیص کوتاه‌تر در سناریوی اول، بیشینه مقدار پهنای باند مصرفی در این سناریو بسیار کمتر از سناریوی دوم بوده و پس از تشخیص حمله، ترافیک و پهنای باند اشغال شده توسط مهاجمان در مدت زمان سریع‌تری آزاد خواهد شد.

موارد عنوان شده، مزیت بکارگیری توان پردازشی سوییچ‌های P4 و قابلیت برنامه‌ریزی این ادوات را نشان می‌دهد، که در نتیجه موجب



(شکل-۷): نمودار مقایسه مصرف پهنای باند کنترلر ONOS در سناریوهای اول و دوم

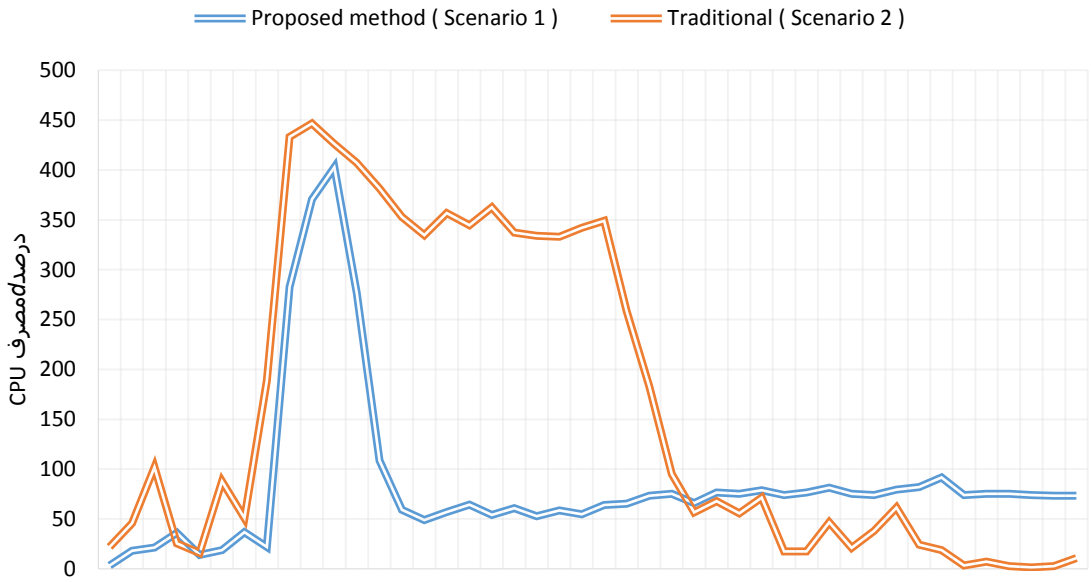
برتری مدل تشخیص حمله ارائه شده در این پژوهش نسبت به روش سنتی تشخیص حمله، که به‌طور کامل مبتنی بر کنترلر بوده، خواهد شد. همانطور که انتظار می‌رفت، مصرف پهنای باند کنترلر شبکه در مدل ارائه شده، در حین حملات slow-rate DDoS، بسیار پایین‌تر بوده است. بر این اساس، پهنای باند اشباع شده در کنترلر توسط سناریوی اول بسیار کمتر خواهد بود و همچنین پهنای باند اشغالی در مدت زمان کمتری آزاد خواهد شد.

۵-۴- ارزیابی سربار پردازشی کنترلر: با توجه به توضیحات پیشین، در سناریوی اول سوییچ‌های P4، طوری برنامه‌ریزی شده‌اند، که بیشتر پردازش‌های محلی مورد نیاز را انجام داده و از این طریق، بسته‌های کمتری جهت پردازش بیشتر به کنترلر فرستاده می‌شوند. اما در سناریوی دوم به دلیل نیاز کنترلر به دید کلی از وضعیت شبکه، جهت استخراج مقادیر ویژگی، تمامی بسته‌ها به کنترلر ارسال شده و انتظار می‌رود کنترلر سربار پردازشی بیشتری را در مقایسه با سناریوی اول متحمل شود. نتایج حاصل از ارزیابی این دو سناریو در نمودار موجود در شکل ۷ به تصویر کشیده شده است، که نشان‌دهنده میزان سربار پردازشی بر روی کنترلر می‌باشد.

با توجه به نمودار شکل ۷، میزان سربار پردازشی موجود بر روی کنترلر، در مدل ارائه شده در این پژوهش، بسیار کمتر از حالت پیش فرض مبتنی بر Openflow بوده است. در زمان وقوع حمله، سناریوی اول به دلیل تشخیص سریع‌تر حمله و نیز پردازش بسته‌ها و انجام فرآیند استخراج مقادیر ویژگی‌ها، به‌صورت محلی در خود سوییچ P4، کنترلر شبکه را به میزان کمتری درگیر پردازش‌های لازم در تشخیص حملات کرده و به این جهت سربار پردازشی کنترلر شبکه بسیار کم بوده است. سناریوی اول در بازه حدود ۲۰ ثانیه بیشترین سربار پردازشی را با بیشینه مقدار حدود ۴۰۰ درصد فراهم آورده است، که این مقدار، به سرعت پس از تشخیص حمله کاسته شده است و منابع پردازشی به سرعت آزاد شده‌اند.

این در حالی است که در سناریوی دوم، کنترلر در حدود ۷۵ ثانیه سربار پردازشی بسیاری را با بیشینه مقدار حدود ۴۵۰ درصد، متحمل شده است و روند آزادسازی منابع نیز با اختلاف کمی، کندتر از سناریوی اول می‌باشد. در سناریوی دوم به دلیل زمان تشخیص بیشتر و درگیری کنترلر در تمام طول دوره حمله، که به سبب انجام فرآیند استخراج مقادیر ویژگی از شبکه موجب شده است، میزبان سربار پردازشی بسیار بالا می‌باشد. فرآیند استخراج مقادیر ویژگی و دریافت اطلاعات از وضعیت شبکه، موجب شده سناریوی دوم در بیشینه میزان مصرف پردازنده نیز سربار پردازشی بیشتری را، در مقایسه با سناریوی اول، فراهم آورد.

با توجه به نتایج حاصل شده از این بخش می‌توان به اهمیت بکارگیری ادوات P4 و توان پردازشی آن‌ها که از طریق برنامه‌پذیری حاصل شده است، پی برد. همانطور که انتظار می‌رفت با استفاده از سوییچ‌های P4 فرآیندهای پردازشی موجود در شبکه که موجب ایجاد



(شکل-۸): نمودار مقایسه میزان مصرف بار پردازشی توسط پردازنده در سناریوی اول و سناریوی دوم

سربار برای کنترلر شده بودند، برطرف خواهند شد و این مساله موجب رفع محدودیت‌های موجود، در پیاده‌سازی بسیاری از سازوکارهای موجود در تشخیص حملات و بسیاری از زمینه‌های عملیاتی موجود در شبکه‌های نرم‌افزار محور خواهد شد.

۶- نتیجه‌گیری و پژوهش‌های آتی

با توجه به روش‌های ارائه شده پیشین در زمینه تشخیص حملات slow-rate DDoS، فرآیند تشخیص حمله در سطح شبکه نیاز به توان پردازشی بخصوصی داشته که حاصل رصد و استخراج اطلاعات از شبکه است؛ پیاده‌سازی این فرآیند در کنترلر شبکه، در کیفیت عملکرد کنترلر و درنتیجه کل شبکه تحت اختیار آن، تأثیرگذار خواهد بود. به همین جهت فناوری P4 و ادوات مبتنی بر آن معرفی شدند تا با برنامه-ریزی ادوات در صفحه داده، از توان پردازشی صفحه داده در معماری SDN استفاده شود.

در این مقاله، روشی برای جلوگیری از حملات slow-rate DDoS در سرورهای مبتنی بر پروتکل HTTP ارائه شد. این روش براساس یک مدل بسیار منعطف و با بکارگیری سوییچ‌های P4 و استفاده از کنترلر ONOS و الگوریتم‌های یادگیری ماشین، ارائه شده است. با بکارگیری توان پردازشی صفحه داده از طریق برنامه‌ریزی سوییچ‌های P4 و مدلی منعطف از چهار الگوریتم Decision Tree (CART)، Random Forest، MLP و KNN در تشخیص حمله، فرآیند استخراج اطلاعات و تشخیص حمله کاملاً خارج از کنترلر انجام شده و در نتیجه کنترلر شبکه کمترین سربار پردازشی را در تشخیص حمله و جلوگیری از آن متحمل خواهد شد.

در مدل ارائه شده با بکارگیری توان پردازشی صفحه داده، بسیاری از محدودیت‌های موجود در روش‌های پیشین، برطرف شده و با توجه به نتایج به‌دست آمده، سیستم و مدل ارائه شده در تشخیص حملات slow-rate DDoS از دقت مناسبی برخوردار بوده است؛ که در کنار بهینگی، از جهت پیاده‌سازی نیز وابستگی میان نوع کنترلر و بخش تشخیص حمله وجود نداشته است. با توجه به برنامه‌ریزی سوییچ‌های P4، کیفیت ویژگی‌های استخراج شده بسیار مناسب بوده و به صورت غیرمستقیم در مراحل آموزش الگوریتم‌های یادگیری ماشین نیز تأثیرگذار می‌باشد. این امر که از طریق برنامه‌ریزی سوییچ‌های P4 انجام شده است، نشانگر میزان اهمیت ادوات P4 است که زمینه را برای استفاده از این فناوری در سایر موارد مدیریتی شبکه نیز فراهم می‌آورد.

۶ – مراجع

[1] Santos R, Souza D, Santo W, Ribeiro A, Moreno E, 2019, Machine learning algorithms to detect DDoS attacks in SDN, Concurrency Computat Pract Exper. 2019;e5402

[2] J. A. Pérez-Díaz, I. A. Valdovinos, K. -K. R. Choo and D. Zhu, 2020, A Flexible SDN-Based Architecture for Identifying and Mitigating Low-Rate DDoS Attacks Using Machine Learning, IEEE Access, vol. 8, pp. 155859-155872.

[3] K. S. Sahoo et al., "An Evolutionary SVM Model for DDOS Attack Detection in Software Defined Networks", IEEE Access, vol. 8, pp. 132502-132513, 2020

[4] Mohammadi R, Conti M, Lal C, Kulhari SC, 2019 , SYN-Guard: An effective counter for SYN flooding attack in software-defined networking. Int J Commun Syst. 2019;e4061.

[5] Musumeci, F., Fidanci, A.C., Paolucci, F. et al. , 2022, Machine-Learning-Enabled DDoS Attacks Detectionin P4 Programmable Networks, J Netw Syst Manage 30, 21 (2022).

[6] A. Febro, H. Xiao and J. Spring, 2019, "Distributed SIP DDoS Defense with P4," 2019 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1-8.

[7] X. I. Khooi, L. Csikor, D. M. Divakaran and M. S. Kang, "DIDA: Distributed In-Network Defense Architecture Against Amplified Reflection DDoS Attacks," 2020 6th IEEE Conference on Network Softwarization (NetSoft), 2020, pp. 277-281

[8] J. F. Balarezo, S. Wang, K. G. Chavez, A. Al-Hourani, J. Fu and K. Sithamparanathan, "Low-rate TCP DDoS Attack Model in the Southbound Channel of Software Defined Networks," 2020 14th International Conference on Signal Processing and Communication Systems (ICSPCS), 2020, pp. 1-10

[9] M. Dimolianis, A. Pavlidis and V. Maglaris, "A Multi-Feature DDoS Detection Schema on P4 Network Hardware," 2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), 2020, pp. 1-6

[10] Badotra, S., Panda, S.N. , 2021, SNORT based early DDoS detection system using Opendaylight and open networking operating system in software defined networking, cluster computing , vol. 24, pp. 501-513.

[11] Safaa MAHRACH, Abdelkrim HAQIQ, 2020, " DDoS flooding attack mitigation in software defined networks", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 11, No. 1

[12] Goksel Simsek, Hakan Bostan, Alper Kaan Sarica, Egemen Sarikaya, Alperen Keles, Pelin Angin, Hande Alemdar, Ertan Onur, 2020, DroPPPP: A P4 Approach to Mitigating DoS Attacks in SDN, Information Security Applications. WISA 2019. Lecture Notes in Computer Science, vol 11897, Springer, Cham.

[13] Dhruba Kumar Bhattacharyya, Jugal Kumar Kalita , 2016, DDoS Attacks : evolution, Detection, Prevention, Reaction, and Tolerance , CRC Press

[14] https://opennetworking.org

[15] https://opennetworking.org/onos/

[16] https://wiki.onosproject.org

[17] http://mininet.org

[18] https://p4.org/p4-spec/docs

[19] github.com/p4lang/behavioral-model

[20]https://www.unb.ca/cic/dataset/dos-dataset.html

[21] https://scikit-learn.org/

رضا فلاحی کپورچالی متولد ۱۳۷۶، مدرک کارشناسی ارشد خود را در رشته مهندسی کامپیوتر گرایش شبکه‌های کامپیوتری در سال ۱۴۰۱، از دانشگاه بوعلی سینا همدان دریافت کرده است. زمینه‌های پژوهشی مورد علاقه ایشان عبارتند از : امنیت شبکه، فناوری P4، شبکه‌های بی‌سیم، شبکه‌های نرم‌افزار محور و اینترنت اشیاء صنعتی.

نشانی رایانامه ایشان عبارت است از :
rezfa.kap@gmail.com

رضا محمدی مدرک کارشناسی ارشد و دکتری خود را در رشته شبکه‌های کامپیوتری به ترتیب در سال‌های ۱۳۹۲ و ۱۳۹۶ از دانشگاه صنعتی شیراز دریافت کرده است. پایان‌نامه دکتری ایشان در رابطه با مهندسی ترافیک در شبکه‌های نرم‌افزار محور است. ایشان چندین مقاله در زمینه شبکه‌های حسگر بی‌سیم زیر آب و شبکه‌های نرم‌افزار محور، در کنفرانس‌ها و مجلات بین‌المللی مختلف داشته‌اند. ایشان از سال ۱۳۹۷ به عنوان استادیار در دپارتمان مهندسی کامپیوتر دانشگاه بوعلی سینا همدان فعالیت دارند. زمینه‌های پژوهشی مورد علاقه ایشان عبارتند از : شبکه‌های نرم‌افزار محور، الگوریتم‌های هیوریستیک، امنیت شبکه‌های نرم‌افزار محور، شبکه‌های حسگر بی-سیم زیر آب، شبکه‌های Ad-hoc، شبکه‌های حسگر بی‌سیم زیر زمین، اینترنت اشیاء و اینترنت اشیاء زیر آب.

نشانی رایانامه ایشان عبارت است از :
r.mohammadi@basu.ac.ir

محمد نصیری در حال حاضر با رتبه دانشیار در دانشگاه بوعلی سینا همدان فعالیت دارند. ایشان مدرک کارشناسی ارشد خود را از دانشگاه صنعتی شریف و مدرک دکتری خود را در سال ۱۳۸۷ از دانشگاه پلی‌تکنیک گرنوبل فرانسه دریافت کرده‌اند. ایشان در سال ۱۳۹۸ به عنوان محقق در دانشگاه گرنوبل آلپ (فرانسه) حضور داشته‌اند. زمینه‌های پژوهشی مورد علاقه ایشان عبارتند از : بهبود کارایی فناوری‌های مختلف بی‌سیم شامل شبکه‌های بی‌سیم محلی، شبکه‌های حسگر بی‌سیم، شبکه‌های زیر آب و فناوری‌های به هم پیوسته اینترنت اشیاء.

نشانی رایانامه ایشان عبارت است از :
m.nassiri@basu.ac.ir

روش ارجاع به مقاله :
ر.فلاحی کپورچالی، ر.محمدی، م.نصیری، تشخیص و پیشگیری از حملات نرخ پایین منع سرویس توزیع شده پروتکل HTTP در شبکه‌های نرم‌افزار محور مبتنی بر سوییچ‌های P4 با بکارگیری الگوریتم‌های یادگیری ماشین، دوفصلنامه محاسبات و سامانه های توزیع شده، سال پنجم، شماره ۲، شماره پیاپی ۱۰، صفحه ۲۳ تا ۴۲، سال ۱۴۰۱

How to cite: R.Fallahi Kapourchaali, R.Mohammadi, M.Nassiri, Detection and prevention of slow-rate DDoS attacks on HTTP protocol in P4-based software defined networks using machine learning techniques, Journal of Distributed Computing and Systems(JDCS), Vol 5, Issue 2, Page 23-42, 2023.