فصلنامه مطالعات راهبردی فضای سایبر
سال اول، شماره ۲، بهار ۱۴۰۱
مقاله دوم، از صفحات ۶۲-۴۵

تشخیص و مقابله با حملات سایبری منع خدمت توزیع‌شده با استفاده از ضریب همبستگی پیرسون در شبکه‌های نرم‌افزار محور

میلاد غلامی[۱]، محسن نیک رأی[۲]

تاریخ دریافت: ۱۴۰۰/۰۲/۱۰ **تاریخ پذیرش: ۱۴۰۰/۰۷/۱۶**

چکیده

امروزه با ظهور شبکه‌های نرم‌افزار محور، سازمان‌ها و ارگان‌ها، معماری شبکه‌های نرم‌افزار محور را جایگزین شبکه‌های سنتی کرده و از مزایای این معماری در مدیریت سازمان بهره می‌برند. شبکه‌های نرم‌افزار محور با جداسازی بخش کنترل از بخش انتقال داده، مزیت‌های فراوانی را همچون کنترل‌پذیری بهتر، مدیریت پویا و استفاده بهینه از پهنای باند و منابع شبکه را به ارمغان آورده اما هنوز هم در برابر حملات منع خدمت سرویس آسیب‌پذیر بوده و مهاجمان می‌توانند با ارسال بسته‌های بی‌شمار منابع سخت‌افزاری و نرم‌افزاری شبکه را اشغال کرده و کاربران را در دسترسی به خدمات و سرویس‌ها دچار اخلال کنند.

به این منظور در این مقاله به بررسی و شبیه‌سازی حملات منع خدمت سرویس و نحوه مقابله با آن در شبکه‌های نرم‌افزار محور پرداخته‌شده و الگوریتمی برای شناسایی و کاهش حملات منع خدمت سرویس پیشنهاد داده‌شده است. در الگوریتم پیشنهادی از ضریب همبستگی پیرسون برای تشخیص حملات استفاده‌شده است؛ سپس الگوریتم پیشنهادی توسط شبیه‌ساز مینی نت و با استفاده از کنترل‌کننده OpenDayLight ارزیابی و در انتها کارایی و برتری الگوریتم پیشنهادی نسبت به روش‌های پیشین نشان داده شده است.

کلیدواژه‌ها: شبکه‌های نرم‌افزار محور، حملات منع خدمت توزیع‌شده، ضریب همبستگی پیرسون، کاهش اثر حمله

۱. دانشجوی مقطع دکترای مهندسی فناوری اطلاعات، دانشگاه تهران (نویسنده مسئول)
gholami.miladi@gmail.com
۲. دانشیار و عضو هیئت علمی دانشگاه قم.

مقدمه

موضوع بازدارندگی یکی از نیازهای اساسی تأمین امنیت کشور است. طبیعتاً امنیت در فضای سایبری یکی از مؤلفه‌های اصلی امنیت ملی کشور است. امروزه با رشد هر چه بیشتر فضای سایبری و وابستگی‌های زندگی بشری به این حوزه فناوری، تهدیدات سایبری به زیرساخت‌های ملی نیز موردتوجه دشمنان هر جامعه‌ای قرارگرفته است (مولایی و دیگران، ۱۳۹۷: ۱٤۲)؛ همچنین امروزه با رشد فناوری‌های جدید کاربران شبکه‌های رایانه‌ای افزایش‌یافته و از طرفی مدیریت این حجم از تقاضا در شبکه‌های سنتی IP به دلیل ادغام لایه کنترل و لایه انتقال در سوئیچ‌ها، پیچیده و بسیار مشکل است (Kreutz, ۲۰۱٥: ۱٤)؛ ازاین‌رو معماری شبکه‌های نرم‌افزار محور[۱] (SDN)برای حل مشکلات شبکه‌های سنتی و مدیریت پویا و بهینه منابع به وجود آمد (Ge, ۲۰۱۸: ٥٦۸).

شبکه‌های نرم‌افزار محور با جداسازی بخش کنترل از بخش انتقال داده و متمرکز ساختن بخش کنترل باعث ارائه هرچه بهتر سرویس‌ها و خدمات متنوعی برای کاربران می‌شود . (Raghavan, ۲۰۱۲)شبکه‌های نرم‌افزار محور توسط سیستم‌عامل‌های شبکه، بخش کنترل را هدایت می‌کند که سیستم‌عامل‌های OpenDayLight،Floodlight و Beacon نمونه‌هایی از آن هستند؛ این سیستم‌عامل‌ها توسط پروتکل OpenFlow استانداردسازی شده‌اند. علی‌رغم تمام مزایایی که شبکه‌های نرم‌افزار محور دارا هستند، به دلیل متمرکز بودن بخش کنترل در معرض آسیب‌پذیری بیشتری در مقابل حملات امنیتی هستند ((Porras, ۲۰۱۲: ۱۲۱.

یکی از حملات امنیتی در شبکه‌ها حمله منع خدمت توزیع‌شده[۲] (DDOS)است که با فرستادن تعداد بی‌شماری درخواست، منابع شبکه را اشغال کرده و آن‌ها را غیرقابل‌دسترس می‌کند (Mirkovic,۲۰۰٤:۳۹) .هر بار که یک بسته جدید وارد شبکه‌های نرم‌افزار محور می‌شود و سوئیچ نمی‌تواند ورودی جریان متقابل را پیدا کند، بسته را به کنترل‌کننده ارسال می‌کند تا در مورد چگونگی هدایت به آن کسب تکلیف کند؛ این‌یک فرصت خوب برای مهاجم برای از بین بردن

۱. Software defined Networks

۲. Distributed denial of service attack

منابع و تهیه دسترسی به شبکه است. به‌تازگی، راه‌حل‌های پیشنهادی برای کاهش اثرات مشکلات امنیتی ارائه‌شده است.

بعضی از ایده‌ها سعی می‌کنند تنظیمات خاصی را انجام دهند و یا داده‌های موردنیاز امنیت و داده‌های نرمال را برای جدا کردن فرایند طبقه‌بندی کنند. بعضی از دیگران چارچوب را برای ایجاد هماهنگی و ایمن تمام اجزای سازنده، سفارشی می‌کنند؛ بااین‌حال، هیچ‌کدام از آن‌ها به‌اندازه کافی برای دفاع از حمله منع خدمت توزیع‌شده متمرکز نیستند (Callaghan, ۲۰۱۳).

هدف اصلی ما در این مقاله تشخیص و کاهش اثر حملات منع خدمت توزیع‌شده در شبکه‌های نرم‌افزار محور است که برای رسیدن به این هدف روشی مبتنی بر ساختار شبکه‌ها نرم‌افزار محور با استفاده از ضریب همبستگی پیرسون تعداد بسته‌های ورودی به کنترلر ارائه‌شده است.

بیان مسئله

امنیت شبکه نیاز به هماهنگی دقیق بسیاری از اجزای شبکه برای دفاع در برابر حمله دارد. در معماری سنتی اینترنت، روترها عملیات مسیریابی و کنترل را به‌صورت توزیع‌شده انجام می‌دهند اما در شبکه‌های نرم‌افزار محور، وجود یک کنترل‌کننده مرکزی کار را برای مهاجمان بسیار ساده‌تر می‌کند (Lim, ۲۰۱٤: ٦۳).

مهاجمان می‌توانند با ارسال جریان‌های دروغین با الگوی سرآیندهای غیرتکراری منجر به وقفه یا تخریب زیرساخت‌های شبکه‌های نرم‌افزار محور شوند. برخلاف شبکه‌های سنتی که در آن فقط یک دستگاه، میزبان یا سرویس موردحمله قرار می‌گرفت در شبکه‌های نرم‌افزار محور امروزی به دلیل متمرکز بودن بخش کنترل، کل زیرساخت دچار مشکل می‌شود و دامنه گسترده‌ای از تخریب را به همراه دارند (Chen, ۲۰۱۷: ۲۸).

مهاجمان می‌توانند با ارسال انبوهی از بسته‌های جدید و غیرتکراری ضمن مشغول کردن کنترلر به بررسی وضعیت بسته‌ها جداول جریان سوئیچ‌ها پرکرده و باعث سرریز شدن قوانین آن شوند که در این صورت ارسال بسته عادی و روتین شبکه هم دچار اختلال شده و عملاً شبکه به بن‌بست می‌رسد (Yao, ۲۰۱٦: ۳۱۹).

باوجود پژوهش‌های صورت گرفته در این موضوع هنوز هم مشکلات زیادی در این حوزه باقی است و هدف ما در این مقاله ارائه یک راه‌حل ساده، با محاسبات کم و بدون اشغال کردن منابع زیادی از کنترلر برای تشخیص و مقابله با حمله منع خدمت سرویس در شبکه‌های نرم‌افزار محور است.

مبانی نظری

پیشینه پژوهش

در مقاله (Burduk, ۲۰۱٦: ۷۹۷) برای تشخیص حملات منع خدمت توزیع‌شده، روش مبتنی بر مدل‌های آماری تغییرات میانگین و واریانس در سری‌های زمانی وضعیت ترافیکی شبکه پیشنهادشده است.

در مقاله (Bawany, ۲۰۱۷: ٤۲٥) چارچوبی گسترده مانند شهر هوشمند بر مبنای شبکه‌های نرم‌افزار محور ارائه‌شده است.

در مقاله (Yonghong, ۱۳: ۱۰٥۲) با استفاده از تئوری بی‌نظمی و یک الگوریتم تشخیص ناهنجاری ترافیک غیرطبیعی را تشخیص می‌دهد و بر این اساس رفتار شبکه را پیش‌بینی می‌کند.

مقاله (Kamesh, ۲۰۱٤: ٥۹٦۸) راه‌حل‌های تشخیص و مقابله با حملات منع خدمت توزیع‌شده را به سه دسته روش‌های آماری، روش‌های مبتنی بر الگوریتم و روش‌های مبتنی بر یادگیری ماشین تقسیم می‌کند.

در مقاله (Sengar, ۲۰۰۸: ۷۹٤) با استفاده از فاصله هلینگر حملات منع خدمت توزیع‌شده را در شبکه‌های انتقال صدا بر مبنای آدرس مبدأ تشخیص می‌دهد؛ در ابتدا یک نمونه‌گیری بر روی بسته‌ها صورت می‌گیرد تا برخی پارامترهای الگوریتم به دست آید، سپس بر این اساس به بررسی دیگر بسته ورودی پرداخته و با مقایسه آن‌ها با مقادیر نمونه به مقابله با حمله احتمالی می‌پردازد.

در مقاله ((Mousavi, ۲۰۱٥: ۷۷با نمونه‌گیری بسته‌های ورودی و بررسی آدرس مقصد بسته توسط کنترلر و محاسبه آنتروپی آن‌ها حملات منع خدمت را تشخیص می‌دهد.

مقاله ((Kandoi, ۲۰۱۵: ۱۳۲۲بر روی سوئیچ‌ها تمرکز کرده و جدول‌های جریان سوئیچ‌ها را مورد رصد قرار می‌دهد تا در صورت سرریز شدن این جدول‌ها به مقابله با حمله احتمالی بپردازد.

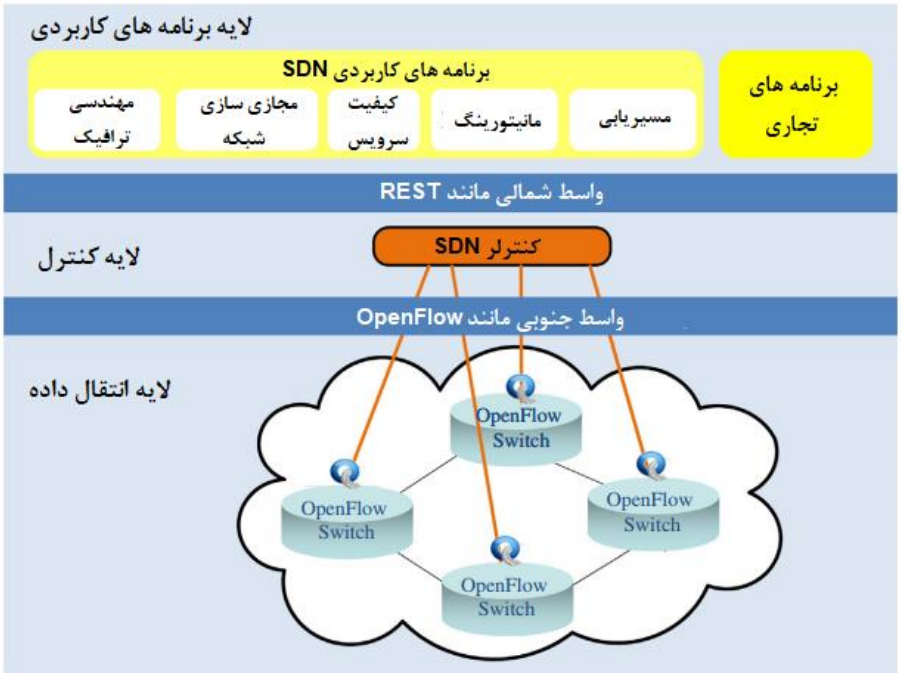
در مقاله (Jian-rui, ۲۰۱۰) سامانه تشخیص نفوذ به‌صورت لایه‌ای بر روی سطح شبکه‌های نرم‌افزار محور و OpenFlow پیشنهادشده که به‌صورت یک کنترلر مرکزی بر روی زیر کنترلرها نظارت دارد و با دریافت رویدادها از زیر کنترلرها از آن‌ها در مقابل حملات احتمالی پشتیبانی می‌کند.

در مقاله (Geometry, ۲۰۰۳) با استفاده از فن یادگیری ماشین ابتدا آمار جریان‌ها از سوئیچ‌ها جمع‌آوری‌شده و از پارامترهایی همچون تعداد بسته‌های متوسط در هر جریان، تعداد بیت‌های متوسط در هر جریان و رشد یک جریان واحد برای آموزش SOM مورداستفاده قرار می‌گیرند.

در مقاله (Dao, ۲۰۱۵: ۳۰۹) یک الگوریتم تشخیص حمله منع خدمت توزیع‌شده بر اساس فیلتر کردن آدرس‌های IP پیشنهادشده که بسته ورودی به کنترلر را بر اساس آدرس IP آن‌ها تقسیم‌بندی می‌کند و با بررسی این گروه‌ها آدرس IP می‌تواند آدرس‌های جعلی را شناسایی کند.

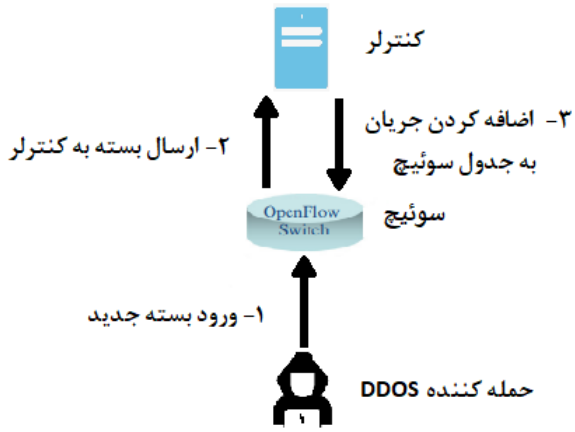
شبکه‌های نرم‌افزار محور

شبکه‌های نرم‌افزار محور یک معماری در حال ظهور است که با جداسازی بخش کنترل از بخش انتقال داده مزایایی همچون استفاده بهینه از منابع شبکه، مدیریت آسان و پویاتر، کاهش هزینه‌های عملیاتی، ترویج نوآوری را به ارمغان آورده است؛ همچنین این جداسازی یک دید متمرکز به مدیر شبکه می‌دهد تا بتواند برنامه‌های موردنیاز در لایه برنامه‌های کاربردی را به فراخور نیاز به کار گیرد. همان‌طور که در شکل ۱ نشان داده‌شده، معماری شبکه‌های نرم‌افزار محور متشکل از سه لایه است که لایه کنترل که توسط کنترلر هدایت می‌شود می‌تواند با واسط‌هایی مانند واسط REST با لایه برنامه‌های کاربردی ارتباط برقرار کرده و درخواست‌ها و سرویس‌های موردنظر را بر روی سوئیچ‌ها در لایه انتقال داده اعمال کند. ارتباط بین لایه کنترل و لایه انتقال داده، می‌تواند توسط واسط‌هایی همچون واسط OpenFlow صورت گیرد (Akyildiz, ۲۰۱۶: ۵۲).



شکل ۱: معماری شبکه‌های نرم‌افزار محور

سوئیچ‌ها بر اساس جدول جریان موجود در حافظه خود بسته‌ها را به مقصد هدایت می‌کنند اما هنگامی‌که بسته جدید از راه می‌رسد که اطلاعات بسته با هیچ‌یک از جریان‌ها یکسان نیست و قانونی برای ارسال بسته وجود ندارد؛ بسته به کنترلر فرستاده می‌شود تا کنترلر برای بسته تصمیم‌گیری کرده و قوانین مربوط به بسته جدید را در سوئیچ‌ها نصب کند. همان‌طور که در شکل ۲ نشان داده‌شده است مهاجمان نیز می‌توانند بر همین اساس بسته‌های جدید و متفاوت را به سوئیچ ارسال کنند تا سوئیچ بسته‌ها را برای کسب تکلیف به کنترلر بفرستد؛ اگر حمله‌کننده، انبوهی از بسته‌های جدید به سوئیچ بفرستد، سوئیچ تمام بسته‌ها را به کنترلر فرستاده و این حجم بالای درخواست می‌تواند در کار کنترلر اختلال ایجاد کند و حتی آن را از کار بیندازد (Yin, ۲۰۱۸).



شکل ۲: مراحل حمله DDOS به کنترلر

روش تحقیق

پژوهش حاضر به‌منظور تشخیص و کاهش تأثیر حملات سایبری منع خدمت توزیع‌شده در شبکه‌های نرم‌افزار محور است؛ لذا در این پژوهش با استفاده از همبستگی تعداد بسته‌های ورودی از آدرس‌های مختلف در شبکه‌های نرم‌افزار محور به تشخیص حملات پرداخته‌شده است و در انتها با شبیه‌سازی حمله منع خدمت توزیع‌شده توسط برنامه اسکاپی[۱] در سیستم‌عامل اوبونتو[۲] و شبیه‌سازی معماری شبکه‌های نرم‌افزار محور در شبیه‌ساز مینی نت[۳] و اجرای برنامه وایرشارک[۴] و تحلیل بسته‌ها می‌توان اثربخشی الگوریتم پیشنهادی در یک محیط آزمایشی را بررسی کرد.

۱ Scapy

۲ Ubuntu

۳ MiniNet

۴ WireShark

راه‌حل پیشنهادی

با توجه به متمرکز بودن بخش کنترل در شبکه‌های نرم‌افزار محور و همچنین برنامه‌های کاربردی موجود در لایه برنامه‌های کاربرد این معماری، می‌توان ابتدا شمای کلی شبکه را به دست آورد و سپس آمار وضعیت سوئیچ‌ها را توسط برنامه‌های کاربردی به دست آورد؛ با استفاده از الگوریتم پیشنهادی که بر مبنای ضریب همبستگی پیرسون است، حملات منع خدمت سرویس را تشخیص داد و با آن مقابله کرد.

فن ضریب همبستگی پیرسون برای تشخیص حملات

در مباحث آماری، ضریب همبستگی پیرسون یا ضریب همبستگی حاصل‌ضرب گشتاور پیرسون میزان همبستگی خطی بین دو متغیر تصادفی را می‌سنجد؛ مقدار این ضریب بین ۱- تا ۱ تغییر می‌کند که ۱ به معنای همبستگی مثبت کامل، به معنی نبود همبستگی و ۱- به معنی همبستگی منفی کامل است ((Kreyszig, ۲۰۱۰.

مقدار قدر مطلق ضریب همبستگی، شدت یا درجه رابطه بین دو متغیر و علامت آن (مثبت یا منفی) جهت رابطه (مستقیم یا معکوس) را نشان می‌دهد. ضریب همبستگی پیرسون یک شاخص متقارن است؛ یعنی همبستگی بین متغیرهای X و Y با همبستگی بین متغیرهای Y و X برابر است؛ شدت یا درجه همبستگی؛ درصورتی‌که ضریب همبستگی محاسبه‌شده معنی‌دار باشد، هر چه مقدار قدر مطلق آن پایین و نزدیک صفر باشد، رابطه ضعیف و هرچه بالاتر و نزدیک ۱ باشد، رابطه، قوی‌تر است.

ضریب تعیین میزان اشتراک تغییرات، درصد پراکندگی یا واریانس مشترک بین دو متغیر را تعیین می‌کند. با محاسبه این ضریب می‌توان تعیین کرد که چند درصد از کل واریانس X ناشی از واریانس Y است؛ هر چه این درصد بالاتر باشد، به‌طورقطع رابطه قوی‌تر خواهد بود.
ضریب همبستگی پیرسون بین دو متغیر تصادفی برابر با کوواریانس آن‌ها تقسیم‌بر انحراف معیار آن‌ها تعریف می‌شود.

$$\rho_{X,Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \quad (1)$$

ضریب همبستگی پیرسون برای یک نمونه آماری با n زوج داده، به‌صورت زیر تعریف می‌شود:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \quad (2)$$

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i - \bar{X}}{s_X}\right)\left(\frac{Y_i - \bar{Y}}{s_Y}\right) \quad (3)$$

که در آن کمیت‌ها به‌صورت زیر تعریف‌شده‌اند:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \quad (4)$$

$$s_X = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2} \quad (5)$$

تشخیص حملات منع خدمت توزیع‌شده

در الگوریتم پیشنهادی ما، برای تشخیص حملات منع خدمت توزیع‌شده تعداد بسته‌های ورودی از لینک سوئیچ در بازه زمانی مشخصی، در کنترلر موردبررسی قرار می‌گیرد که ما در این مقاله برای دستیابی به‌دقت تشخیص بالا و هزینه محاسباتی پایین، طول بازه زمانی را ۰/۱ ثانیه در نظر می‌گیریم و اطلاعات به‌دست‌آمده را در بردارهایی ذخیره می‌کنیم؛ سپس این بردارها مورد تجزیه‌وتحلیل قرار می‌گیرند و با توجه به فرمول ضریب همبستگی پیرسون میزان درجه همبستگی بردارها مشخص می‌شوند. مقدار به‌دست‌آمده با حد آستانه مقایسه می‌شوند. اگر مقدار همبستگی بردارها از حد آستانه مربوطه بیشتر باشد، حمله منع خدمت توزیع‌شده، رخ‌داده است.

جریان کاری روش پیشنهادی

در روش پیشنهادی همان‌طور که در شکل ۳ مشخص‌شده است، تعداد بسته‌های ورودی در مدت‌زمان مشخص Δt که در این الگوریتم مقدار آن برابر ۰.۱ ثانیه در نظر گرفته‌شده را به دست آورده و به‌صورت یک‌درمیان در دو بردار X و Y قرار می‌گیرد و بنا به فرمول شماره ۱ میزان ضریب همبستگی دو بردار به دست می‌آورید؛ ده مرتبه و در هر بار تکرار پنج ضریب همبستگی را نمونه‌برداری می‌کنیم؛ نمونه‌برداری برای تمام پورت‌های سوئیچ‌های موجود در شبکه انجام می‌شود؛ درنهایت مقدار ضریب پیوستگی‌های نمونه‌برداری‌شده با مقدار حد آستانه مقایسه می‌شوند و درصورتی‌که هر پنج نمونه، مقداری بیش‌ازحد آستانه داشته باشند، پورت موردنظر موردحمله قرارگرفته است.

Inputs:
γ_j: The sample set of arrival rate
t: The period of time for packet arrivals
MAX_K: The maximum number of sample data
MAX_I: The maximum number of sample correlation
Output:

The port ddos Port carrying out the DDoS attack and being detected.

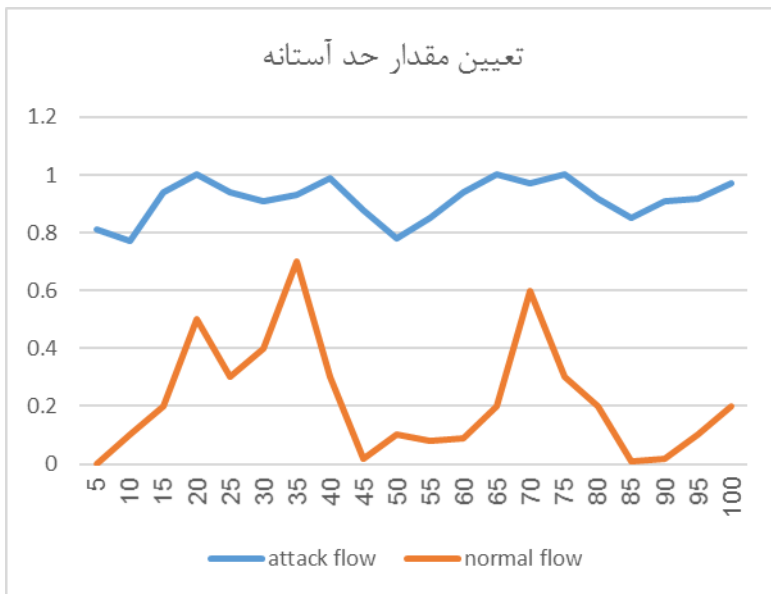
Procedure:
۰۱: MAX_K = ۱۰
۰۲: MAX_I = ٥
۰۳: i = ۰
۰٤: for each i Until MAX_I
۰٥: for each k Until MAX_K
۰٦: γ_j = number of arrival packets during time t
۰۷: if j is odd
۰۸: X[k] = γ_j
۰۹: else
۱۰: Y[k] = γ_j
۱۱: k=k+۱
۱۲: Calculate $\rho_{X,Y}$ by Equ. (۱)

۱۳: $\rho_i = \rho_{X,Y}$
۱٤: i=i+۱
۱٥: for each i Until MAX_I
۱٦: if $\eta U \leq \rho_i \leq ۱$
۱۷: s = s + ۱
۱۸: if s = sum
۱۹: Drop packets from this IP address

شکل ۳: الگوریتم پیشنهادی

تعیین مقدار حد آستانه در الگوریتم دارای اهمیت بسیار زیادی است و اگر مقدار حد آستانه زیاد باشد، بسیاری از بسته حمله به‌عنوان بسته‌های عادی شبکه تلقی می‌شوند و سیستم دچار اخلال می‌شود؛ همچنین درصورتی‌که مقدار حد آستانه کم انتخاب شود، بسیاری از بسته‌های عادی شبکه به‌عنوان بسته‌های حمله تشخیص داده‌شده و جریان عادی شبکه دچار اشکال می‌شود؛ به همین منظور ما مقدار شباهت کسینوسی را برای صد مورد در حالت عادی شبکه و همچنین در وضعیت حمله نمونه‌برداری کردیم.

همان‌طور که در شکل ٤ مشخص است اگر مقدار حد آستانه را برابر ۰.۷ در نظر بگیریم، می‌توان بسته‌های حمله را تشخیص داد بدون آنکه خللی به جریان عادی شبکه وارد شود.

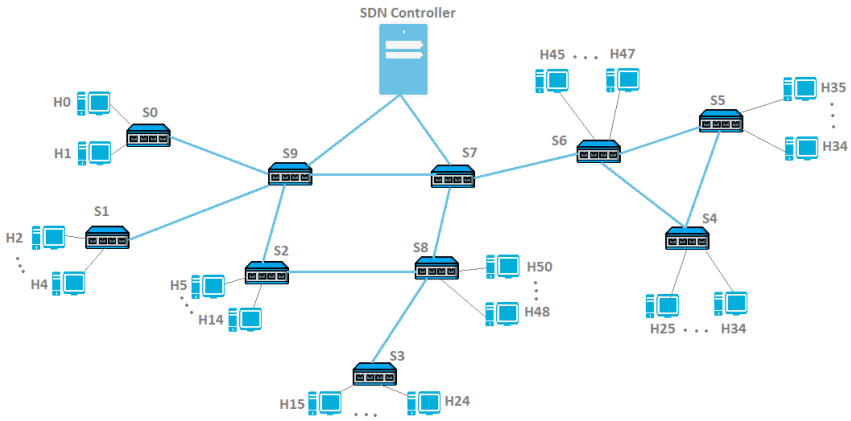


شکل ٤: تعیین مقدار حد آستانه

شبیه‌سازی حمله منع خدمت توزیع‌شده

برای راه‌اندازی شبکه نرم‌افزار محور و تولید ترافیک‌های حمله منع خدمت توزیع‌شده در محیط مینی نت، مطابق شکل ۵، توپولوژی درختی متشکل از پنجاه میزبان و نُه سوئیچ را به همراه لینک‌های مجازی ایجاد می‌کنیم؛ سوئیچ‌های این شبکه به نحوی تنظیم می‌شوند که به یک کنترلر در حال اجرای خارجی با آدرس مشخص متصل باشند؛ ما در این توپولوژی برای ایجاد سوئیچ‌ها، از سوئیچ مجازیک OVS که باقابلیت اجرا بر روی سخت‌افزار و نرم‌افزار است، استفاده می‌کنیم. بعد از ایجاد توپولوژی، کنترلر OpenDayLight را برای شناسایی آدرس‌های مک در لایه دو و انجام عمل سوئیچینگ، تنظیم و اجرا می‌کنیم. با اجرای کنترلر، سوئیچ‌ها به کنترلر متصل می‌شوند. توپولوژی ایجادشده را در شکل بالا می‌بینید؛ همان‌طور که در این شکل نشان داده‌شده است، رایانه‌های میزبان به سوئیچ‌های مجازی بر اساس پروتکل OpenFlow متصل شده‌اند و خود این سوئیچ‌ها نیز به یک سوئیچ سطح بالاتر متصل هستند؛ این مجموعه، تشکیل‌دهنده بخش داده‌ای شبکه است. بخش داده‌ای شبکه از طریق سوئیچ‌ها به کنترلر OpenDayLight که خارج از شبکه مجازی قرار دارد، متصل می‌شود و دستورهای مربوطه را در صورت لزوم از آن دریافت می‌کند؛ پس از ایجاد توپولوژی، با استفاده از برنامه اسکاپی دو نوع ترافیک حمله و ترافیک عادی را در شبکه تولید می‌کنیم. برای تعیین اینکه یک بسته اطلاعاتی در اینترنت یا سایر شبکه‌ها به چه برنامه‌ای در میزبان مقصد تعلق بگیرد، از شماره درگاه استفاده می‌شود. با توجه به اینکه در حالت پیش‌فرض پروتکل OpenFlow تنها سرآیند بسته‌ها به کنترلر ارسال می‌گردد، بسته‌های تولیدی در این شبیه‌سازی فاقد محتوی خواهند بود. پس از ایجاد توپولوژی و اجرای کنترلر در مینی نت، برنامه وایرشارک را اجرا می‌کنیم. در مینی نت IP آدرس‌های همه میزبان‌ها از ۱.۰.۰.۱۰ شروع‌شده و به ترتیب افزایش می‌یابد. برای ایجاد حمله به یک میزبان، ما میزبان‌های h۱,h۲ را به‌عنوان میزبان تسخیرشده به‌منظور حمله به میزبان h۳ در نظر می‌گیریم. در یک ترمینال جدا در محیط مینی نت، برنامه اسکاپی را اجرا کرده و بسته‌های ترافیک نرمال را از سمت میزبان‌های h۱,h۲ به مقصد میزبان h٦ در شبکه تولید می‌کنیم. در اکسترمینال مربوط به میزبان h۱,h۲ برنامه اسکاپی

دیگری را به‌منظور ایجاد بسته‌های ترافیک حمله به مقصد میزبان h۳ اجرا می‌کنیم؛ در این شبیه‌سازی، تولید بسته‌های جعلی به‌منظور انجام حمله ممانعت از سرویس توزیع‌شده در شبکه را به مدت ۱۸۰ ثانیه ادامه می‌دهیم؛ در این حمله شبیه‌سازی‌شده، بسته‌های ترافیک حمله با سرعت بیشتری نسبت به بسته‌های ترافیک نرمال تولید خواهند شد؛ پس از انجام حمله، بسته‌های ردوبدل شده در شبکه را توسط برنامه وایرشارک اجراشده در مینی نت، رصد می‌کنیم. در حملات ممانعت از سرویس توزیع‌شده تولید بسته‌های حمله، سریع‌تر از تولید بسته‌های ترافیک نرمال انجام می‌گیرد.



شکل ۵: معماری شبیه‌سازی‌شده

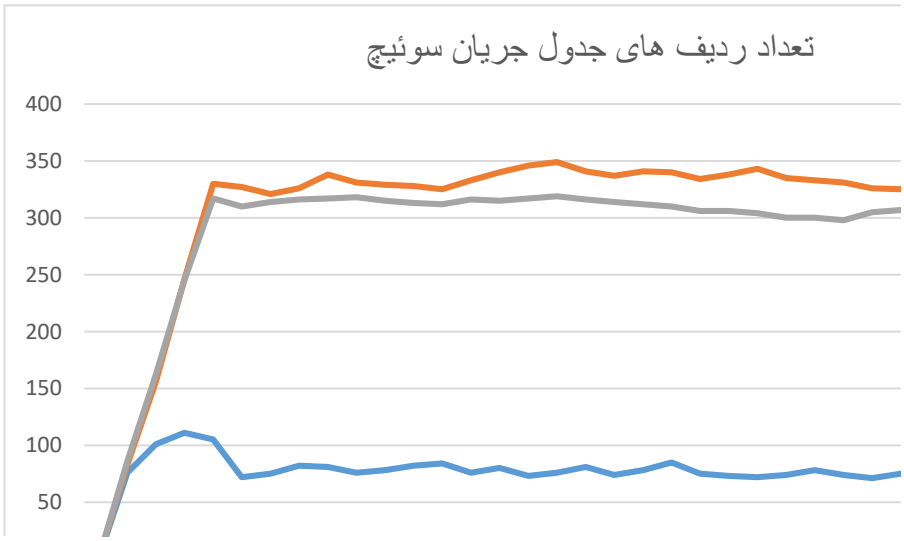
ارزیابی و مقایسه

در این بخش، الگوریتم ارائه‌شده را با دو روش دیگر مقابله با حملات منع خدمت توزیع‌شده مقایسه کرده‌ایم. الگوریتم روش اول با ردگیری آدرس IP ورودی به شبکه و شمارش بسته‌های ورودی به کنترلر حملات منع خدمت توزیع‌شده را تشخیص می‌دهد (Dao, ۲۰۱۵: ۳۰۹)؛

الگوریتم روش دوم با شمارش آدرس مبدأ بسته‌ها و کنترل تعداد بایت‌های موجود در هر درخواست به تشخیص حملات منع خدمت توزیع‌شده می‌پردازد (Gkountis, ۲۰۱۸).

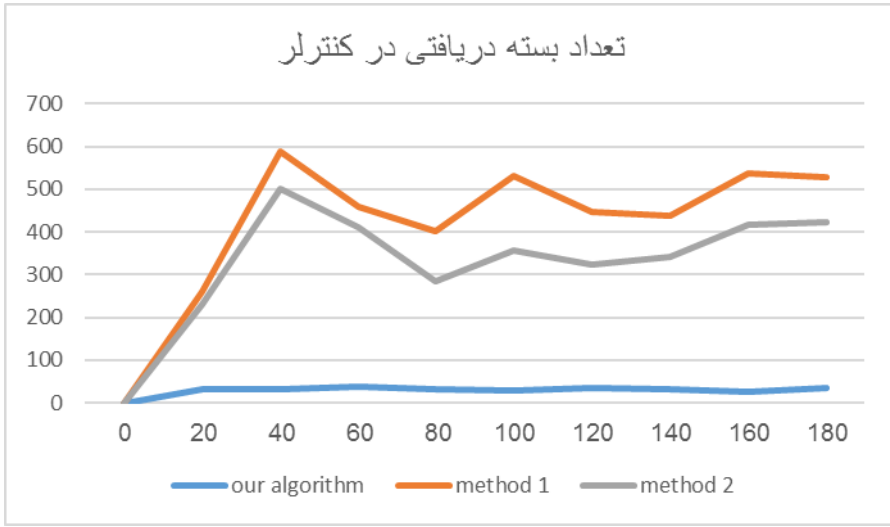
اگر مقدار idle timeout را برابر بیست در نظر بگیریم، درصورتی‌که به مدت ۲۰ ثانیه از قوانین جدول جریان سوئیچ‌ها استفاده نشود، آن قانون حذف می‌شود. همان‌طور که در شکل ٦ نشان داده‌شده است، تعداد قوانین موجود در جدول جریان در الگوریتم پیشنهادی و دو روش

دیگر مقایسه شده است. الگوریتم پیشنهادی برخلاف دو روش دیگر که در ۲۰ ثانیه ابتدایی، سیری کاملاً صعودی دارند، در ثانیه پنجم موفق به تشخیص حمله شده و با حذف بسته‌ها از ورود آن‌ها به کنترلر خودداری می‌کند و با توجه به مقدار idle timout تقریباً بعد از گذشت ۲۰ ثانیه، الگوریتم روند ثابتی را در پیش می‌گیرد و کارهای روتین شبکه انجام می‌شود؛ اما در دو روش دیگر چون بسته‌های ورودی از طرف مهاجمان به‌صورت منحصربه‌فرد هستند برای هریک قانونی در سوئیچ نصب می‌شود و به همین علت تعداد قوانین نصب‌شده در جدول جریان در ابتدا سیر صعودی دارند که به دلیل منحصربه‌فرد بودن قوانین پس از ۲۰ ثانیه قوانین اضافی حذف می‌شود و نمودارها، روند تقریباً ثابتی را در پیش می‌گیرند و عملاً موفق به تشخیص حمله نمی‌شوند.



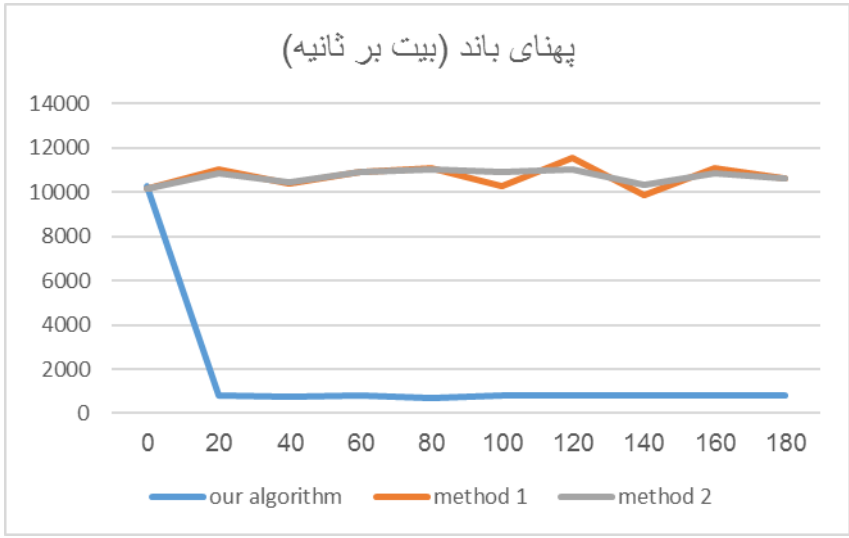
شکل ٦: تعداد ردیف‌های جدول جریان سوئیچ

اگر قانونی در جدول جریان برای هدایت بسته‌ها موجود نباشد، بسته به کنترلر فرستاد می‌شود. در شکل ۷ به بررسی تعداد بسته‌های ورودی به کنترلر در الگوریتم پیشنهادی و دو روش دیگر پرداخته‌ایم؛ همان‌طور که در شکل مشخص است، تعداد بسته‌هایی که در الگوریتم پیشنهادی ظرف مدت ۱ ثانیه به کنترلر فرستاده می‌شوند نسبت به دو روش دیگر کمتر بوده و سوئیچ‌ها با اعمال قوانین از ورود بسته‌های مهاجم به کنترلر جلوگیری می‌کنند.



شکل ۷: تعداد بسته دریافتی در کنترلر

پهنای باند کانال ارتباطی سوئیچ‌ها و کنترلر در شکل ۸ با یکدیگر مقایسه شده‌اند که الگوریتم پیشنهادی در زمان آغاز حمله به‌اندازه دو روش دیگر از پهنای باند استفاده کرده اما پس مدت‌زمان کوتاهی حمله را تشخیص داده و با مقابله با مهاجم از اشغال پهنای باند جلوگیری می‌کند و پهنای باند را به جریان عادی شبکه اختصاص می‌دهد؛ اما در روش‌های دیگر تشخیص حمله به‌درستی صورت نمی‌گیرد و پهنای باند زیادی توسط مهاجم اشغال می‌شود.



شکل ۸: پهنای باند (بیت بر ثانیه)

نتیجه‌گیری

پژوهش حاضر به‌منظور تشخیص و کاهش اثر حملات منع خدمت توزیع‌شده در چارچوب شبکه‌های نرم‌افزار محور انجام گرفت. شبکه‌های نرم‌افزار محور ضمن متمرکز کردن بخش کنترل و مدیریت آسان‌تر امکان حملات منع خدمت توزیع‌شده را نسبت به شبکه‌های سنتی افزایش می‌دهد. در این مقاله با استفاده از مزیت‌های شبکه‌های نرم‌افزار محور الگوریتمی برای تشخیص و کاهش اثر حملات منع خدمت توزیع‌شده ارائه‌شده است.

در الگوریتم پیشنهادشده، از ضریب همبستگی پیرسون برای شناسایی بسته‌های جعلی استفاده‌شده است و روش پیشنهادی منابع زیادی از کنترلر را برای تشخیص حمله مورداستفاده قرار نمی‌دهد و نیاز به پیش پردازهای طولانی ندارد و از یک مقدار آستانه برای تعیین اینکه آیا یک حمله منع خدمت توزیع‌شده رخ‌داده است، استفاده تا مهاجم واقعی پیدا و حمله در منبع متوقف شود.

درنهایت، نتایج شبیه‌سازی نشان می‌دهد که الگوریتم پیشنهادی می‌تواند در مدت‌زمان کوتاهی حملات منع خدمت توزیع‌شده را تشخیص دهد و آسیب‌پذیری شبکه در مقابل حملات را بهبود بخشد.

در کارهای آینده می‌توان کارایی الگوریتم ارائه‌شده را در مجموعه‌ای از کنترلرهای به‌هم‌پیوسته بررسی کرد.

فهرست منابع و مآخذ

الف- منابع فارسی

- ملائی، علی؛ کارگری، مهرداد و خراشادی‌زاده، محمدرضا (۱۳۹۷)، الگوی بازدارندگی در فضای سایبر بر اساس نظریه بازی‌ها، فصلنامه امنیت ملی، ۸ (۲۹)، ۱۷۲-۱۴۱.

- Akyildiz, I. F., Lee, A., Wang, P., Luo, M., & Chou, W. (۲۰۱۶). Research challenges for traffic engineering in software defined networks. IEEE Network, ۳۰(۳), ۵۲–۵۸. https://doi.org/۱۰,۱۱۰۹/MNET.۲۰۱۶,۷۴۷۴۳۴۴
- Bawany, N. Z., Shamsi, J. A., & Salah, K. (۲۰۱۷). DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions. Arabian Journal for Science and Engineering, ۴۲(۲), ۴۲۵–۴۴۱. https://doi.org/۱۰,۱۰۰۷/s۱۳۳۶۹-۰۱۷-۲۴۱۴-۵
- Burduk, R., Jackowski, K., Kurzyński, M., Woźniak, M., & Żołnierek, A. (۲۰۱۶). Proceedings of the ۹th international conference on computer recognition systems CORES ۲۰۱۵. Advances in Intelligent Systems and Computing, ۴۰۳, ۷۹۷–۸۰۶. https://doi.org/۱۰,۱۰۰۷/۹۷۸-۳-۳۱۹-۲۶۲۲۷-۷
- Callaghan, O., Security, S. S. D. N., In, A. S., & Sdn, I. (۲۰۱۳). SDN Security : A Survey Queen ' s University Belfast - Research Portal SDN Security : A Survey. ۱–۷. https://doi.org/۱۰,۱۱۰۹/SDN۴FNS.۲۰۱۳,۶۷۰۲۵۵۳
- Chen, K. Y., Junuthula, A. R., Siddhrau, I. K., Xu, Y., & Chao, H. J. (۲۰۱۷). SDNShield: Towards more comprehensive defense against DDoS attacks on SDN control plane. ۲۰۱۶ IEEE Conference on Communications and Network Security, CNS ۲۰۱۶, ۲۸–۳۶. https://doi.org/۱۰,۱۱۰۹/CNS.۲۰۱۶,۷۸۶۰۴۶۷
- Dao, N. N., Park, J., Park, M., & Cho, S. (۲۰۱۵). A feasible method to combat against DDoS attack in SDN network. International Conference on Information Networking, ۲۰۱۵-Janua, ۳۰۹–۳۱۱. https://doi.org/۱۰,۱۱۰۹/ICOIN.۲۰۱۵,۷۰۵۷۹۰۲
- Ge, M., Hong, J. B., Yusuf, S. E., & Kim, D. S. (۲۰۱۸). Proactive defense mechanisms for the software-defined Internet of Things with non-patchable vulnerabilities. Future Generation Computer Systems, ۷۸, ۵۶۸–۵۸۲. https://doi.org/۱۰,۱۰۱۶/j.future.۲۰۱۷,۰۷,۰۰۸
- Gkountis, C., Taha, M., Lloret, J., & Kambourakis, G. (۲۰۱۸). Lightweight algorithm for protecting SDN controller against DDoS attacks. Proceedings - WMNC ۲۰۱۷: ۱۰th Wireless and Mobile Networking Conference, ۲۰۱۸-Janua(April ۲۰۱۸), ۱–۶. https://doi.org/۱۰,۱۱۰۹/WMNC.۲۰۱۷,۸۲۴۸۸۵۸
- Jian-rui, C., Zeng-ying, H. E., & Yong-cheng, L. (۲۰۱۰). Design of Network Intrusion Detection System on IPv۶. Computer, ۹.

- Kamesh, & Sakthi Priya, N. (۲۰۱٤). Gbaam. International Journal of Applied Engineering Research, ۹(۲۲), ٥۹٦۸–٥۹۷٤. https://doi.org/۱۰,۱۰۰۲/sec
- Kandoi, R., & Antikainen, M. (۲۰۱٥). Denial-of-service attacks in OpenFlow SDN networks. Proceedings of the ۲۰۱٥ IFIP/IEEE International Symposium on Integrated Network Management, IM ۲۰۱٥, ۱۳۲۲–۱۳۲٦. https://doi.org/۱۰,۱۱۰۹/INM.۲۰۱٥,۷۱٤۰٤۸۹
- Kreutz, D., Ramos, F. M. V., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (۲۰۱٥). Software-defined networking: A comprehensive survey. Proceedings of the IEEE, ۱۰۳(۱), ۱٤–۷٦. https://doi.org/۱۰,۱۱۰۹/JPROC.۲۰۱٤,۲۳۷۱۹۹۹
- Kreyszig, E. (۲۰۱۰). Advanced Engineering Mathemathics, ۱۰th Ed. Wiley, ۱٤. https://doi.org/۱۰,۲۳۰۷/۳٦۱۲٥۲۳
- Lim, S., Ha, J., Kim, H., Kim, Y., & Yang, S. (۲۰۱٤). A SDN-oriented DDoS blocking scheme for botnet-based attacks. International Conference on Ubiquitous and Future Networks, ICUFN, ٦۳–٦۸. https://doi.org/۱۰,۱۱۰۹/ICUFN.۲۰۱٤,٦۸۷٦۷٥۲
- Mirkovic, J., & Reiher, P. (۲۰۰٤). A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, ۳٤(۲), ۳۹. https://doi.org/۱۰,۱۱٤٥/۹۹۷۱٥۰,۹۹۷۱٥٦
- Mousavi, S. M., & St-Hilaire, M. (۲۰۱٥). Early detection of DDoS attacks against SDN controllers. ۲۰۱٥ International Conference on Computing, Networking and Communications, ICNC ۲۰۱٥, ۷۷–۸۱. https://doi.org/۱۰,۱۱۰۹/ICCNC.۲۰۱٥,۷۰٦۹۳۱۹
- Porras, P., Shin, S., Yegneswaran, V., Fong, M., Tyson, M., & Gu, G. (۲۰۱۲). FortNox-HotSDN۲۰۱۲. ۱۲۱–۱۲٦.
- Raghavan, B., Ghodsi, A., Ratnasamy, S., Shenker, S., & Berkeley, I. U. C. (۲۰۱۲). Software-Defined Internet Architecture.pdf.
- Sengar, H., Wang, H., Wijesekera, D., & Jajodia, S. (۲۰۰۸). Detecting voIP floods using the hellinger distance. IEEE Transactions on Parallel and Distributed Systems, ۱۹(٦), ۷۹٤–۸۰٥. https://doi.org/۱۰,۱۱۰۹/TPDS.۲۰۰۷,۷۰۷۸٦
- Shalimov, A., Zuikov, D., Zimarina, D., Pashkov, V., & Smeliansky, R. (۲۰۱٤). Advanced study of SDN/OpenFlow controllers. (October), ۱–٦. https://doi.org/۱۰,۱۱٤٥/۲٥٥٦٦۱۰,۲٥٥٦٦۲۱
- Yao, Z., & Yan, Z. (۲۰۱٦). Security in software-defined-networking: A survey. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), ۱۰۰٦٦ LNCS(December), ۳۱۹–۳۲۲. https://doi.org/۱۰,۱۰۰۷/۹۷۸-۳-۳۱۹-٤۹۱٤۸-٦_۲۷
- Yin, D., Zhang, L., & Yang, K. (۲۰۱۸). A DDoS Attack Detection and Mitigation with Software-Defined Internet of Things Framework. IEEE Access, ٦(c), ۲٤٦۹٤–۲٤۷۰٥. https://doi.org/۱۰,۱۱۰۹/ACCESS.۲۰۱۸,۲۸۳۱۲۸٤
- Yonghong Chen, Xinlei Ma, & Xinya Wu. (۲۰۱۳). DDoS Detection Algorithm Based on Preprocessing Network Traffic Predicted Method and Chaos Theory. IEEE Communications Letters, ۱۷(٥), ۱۰٥۲–۱۰٥٤. https://doi.org/۱۰,۱۱۰۹/lcomm.۲۰۱۳,۰۳۱۹۱۳,۱۳۰۰٦٦