SPUNNID: روشی جدید در تشخیص حملات ازکاراندازی سرویس توزیع شده مبتنی بر ترکیب روش‌های آماری و شبکه‌های عصبی مصنوعی بدون ناظر

رسول جلیلی، فاطمه ایمانی‌مهر، مرتضی امینی
دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
{ jalili@, imani@mehr., amini@mehr.} sharif.edu

چکیده

از آنجایی‌که پیش گیری قطعی از رخداد حملات *DDoS* ممکن نیست، تشخیص این حملات می‌تواند گام مهمی در جلوگیری از پیشرفت حمله باشد. در حمله *DDoS* مهاجم با ارسال بسته‌هایی شبیه به بسته‌های نرمال سعی در سیلابی نمودن ترافیک ماشین هدف دارند. در نتیجه سیستم‌های تشخیص حمله همه‌منظوره موفقیت چندانی در تشخیص حملات *DDoS* ندارند. از سوی دیگر ماهیت توزیع شدگی این حملات، تشخیص آن ها را مشکل تر مینماید. در این مقاله یک روش برای تشخیص این‌گونه حملات بر مبنای ترکیب پیش‌پردازنده آماری و شبکه‌های عصبی بدون ناظر ارائه شده است. در این روش ابتدا با در نظر گرفتن مجموعه بسته‌های موجود در یک بازه زمانی، ویژگی‌های آماری نشان دهنده رفتار این‌گونه حملات، از آنان استخراج شده است. سپس با استفاده از شبکه‌های عصبی بدون ناظر، این ویژگی‌ها تحلیل و دسته‌بندی شده‌اند.

ایده اصلی این مقاله در تشخیص این‌گونه حملات این است که مهاجم در حمله *DDoS* هر چقدر هم که بتواند بسته‌های نرمال به سمت ماشین هدف ارسال نماید، باز نمی‌تواند ترافیکی شبیه به ترافیک نرمال ماشین هدف روی آن ایجاد کند. در پایان، این روش با استفاده از ترافیکی که در یک محیط واقعی جمع‌آوری شده است، ارزیابی شده است. در این ارزیابی پارامترهای درصد تشخیص درست، درصد خطای مثبت غلط و درصد خطای منفی غلط به عنوان معیار در نظر گرفته شده اند. نتایج ارزیابی نشاندهنده کارایی بالای روش ارائه شده نسبت به روشهای قبلی میباشد.

کلید واژه: حملات از کار اندازی سرویس توزیع شده (*DDoS*)، پیش‌پردازنده‌ی آماری، شبکه‌ی عصبی مصنوعی بدون‌ناظر، شبکهٔ *ART1*.

۱- مقدمه

تولید حملات از کاراندازی سرویس سیلابی[۱]، روی ماشین‌های قدرتمند امروزی، توسط یک ماشین امکان‌پذیر نیست. هم‌چنین پی‌گیری حمله‌ای که از چند مبدأ متفاوت صورت گرفته باشد، به مراتب مشکل‌تر از حمله‌ایست که فقط از یک مبدأ صورت گرفته باشد. در نتیجه مهاجمان می توانند با به خدمت گرفتن صدها و شاید هزاران ماشین مختلف، و تولید حملات از کار اندازی سرویس توسط آنان یک حمله از کار اندازی سرویس توزیع شده (*DDoS*)[۲] را تشکیل دهند و توسط آن هر ماشین قدرتمندی را از کار بیندازند.

یک حمله *DDoS* شامل سه جزء اصلی است: ۱) فرمانده یا همان مهاجم اصلی، ۲) ماشین‌های میزبان ۳) ماشین قربانی. در مرحله اول حمله، مهاجم اصلی با پیدا نمودن میزبان‌هایی برای شرکت در حمله، نرم‌افزار کوچکی برای ارتباط با آنان در آینده، روی آنان نصب می‌کند. مرحله اصلی حمله، با صدور دستور از طرف فرمانده به سوی میزبان‌ها، توسط آنان صورت می‌پذیرد. هر یک از میزبان‌ها با دریافت این دستور، بسته به نوع دستور، حملات از کار اندازی سرویس سیلابی معرفی شده توسط فرمانده را به سمت ماشین قربانی آغاز می‌کنند. و به این ترتیب قربانی یک حمله *DDoS*، با سیلی از بسته‌های ارسالی از نقاط مختلف روبرو می‌شود، بطوری‌که نمی‌تواند بسته‌های معمول و نرمال را از بسته‌های مربوط به حمله *DDoS* تمییز دهد. و بدین ترتیب درخواست‌های کاربران شبکه ماشین قربانی در طول حمله *DDoS* منع می‌شود.

مدیران شبکه یا سیستم‌ها، با تشخیص بموقع حملات از کار اندازی سرویس، می‌توانند برای جلوگیری از منع بیشتر سرویس‌های کاربران، و یا کشف مهاجمان اقداماتی را انجام دهند.

از آنجایی‌که در تولید حملات *DDoS* از بسته‌هایی با ماهیت نرمال استفاده می‌شود، روش‌های تشخیص حمله عمومی که بر اساس الگوی بسته‌ها و یا اتصالات کار می‌کنند، نمی‌توانند کارایی چندانی در تشخیص حمله *DDoS* داشته باشند.

در این مقاله یک روش جدید برای تشخیص حملات *DDoS* بر مبنای شبکه‌های عصبی بدون ناظر و پیش‌پردازنده‌ی آماری ارائه می‌شود. در این روش با در نظر گرفتن بسته‌های موجود در بازه‌های زمانی کوچک، ویژگی‌های آماری از آنان استخراج می‌شود. این ویژگی‌ها به صورتی تعریف شده‌اند که بتوانند رفتار یک حمله *DDoS* را نشان دهند. سپس با استفاده از شبکه عصبی بدون ناظر *ART1* این ویژگی‌ها تحلیل و دسته‌بندی می‌شوند.

در ادامه این مقاله در بخش ۲ کارهای انجام شده در حیطه این مقاله بررسی می‌شوند. در بخش ۳ ایده روش تشخیص پیشنهاد شده توضیح داده می‌شود. در بخش ۴ معماری سیستم طراحی شده برای این منظور (سیستم *SPUNNID*) بررسی می‌شود. نتایج ارزیابی سیستم معرفی شده در بخش ۵ آورده می‌شود. با توجه به ماهیت حملات *DDoS*، در بخش ۶ یک روش

---

۱ Flooding Denial of Service Attack
۲ Distributed Denial of Service

برای بالا بردن کارایی سیستم معرفی می‌شود. و در نهایت در بخش ۷ به نتیجه‌گیری و ارائه راه‌کارهای آینده پرداخته می‌شود.

۲- کارهای انجام شده

در این بخش ابتدا کارهای انجام شده در تشخیص حملات *DDoS* بررسی می‌شوند و در ادامه نیز استفاده از شبکه‌های عصبی بدون ناظر برای تشخیص تهاجم در تحقیقات گذشته توضیح داده می‌شود.

در [۳] یک ابزار دیده‌بانی در شبکه، با نام *AGURI* معرفی شده است. این ابزار با استفاده از جمع‌آوری الگوی ترافیک، می‌تواند در طولانی مدت بر ترافیک شبکه نظارت داشته باشد و وقوع حمله *DDoS* را تشخیص دهد.

در [۴] روش‌های ترتیبی تطبیقی و ترتیبی دسته‌ای برای تشخیص سریع حملات *DDoS* معرفی شده است. در این روش، با استفاده از تحلیل‌های آماری اطلاعات مربوط به لایه‌های مختلف شبکه، سعی در پیدا نمودن تغییر ناگهانی در ترافیک شده است. ایده این روش این است که حملات *DDoS* یک تغییر ناگهانی در مدل آماری ترافیک در مقایسه با ترافیک نرمال ایجاد می‌کنند.

در [۵] ساختار داده *MULTOPS* جهت تشخیص حملات *DDoS* ارائه شده است. با استفاده از این ساختار داده، جریان داده‌های ورودی به یک شبکه با جریان داده‌های خروجی از آن شبکه با یکدیگر مقایسه می‌شوند و در صورت تشخیص نامتقارنی موجود بین این دو، حمله تشخیص داده می‌شود.

در [۶] ترافیک شبکه؛ که به صورت نرخ نشانه‌های *tcp* و نرخ قراردادهای مختلف بیان شده است، تحلیل شده است. و نشان داده شده است که هنگام رخداد حمله، این نرخ‌ها به وضوح تغییر می‌کنند. بعلاوه برای تشخیص حملات *DDoS*، مجموعه‌ای از قوانین حالت[۱] توسط الگوریتم‌های یادگیری ماشین[۲] تولید شده و در یک محیط شبیه‌سازی ارزیابی شده‌اند.

در [۷] یک روش برای تشخیص حملات *DDoS* بر مبنای پایه های تئوری اطلاعات[۳] مخصوصا پیچیدگی *Kolmogorov* بیان شده است. در این روش با فرض این‌که مهاجمان حمله *DDoS*، با تولید بسته‌های مشابه از مقصدهای مختلف، سعی در سیلابی نمودن ماشین هدف می‌نمایند، ادعا شده است که با استفاده از روشی بر پایه پیچیدگی *Kolmogorov* می‌توان بسته‌هایی با الگوهای مشابه را تشخیص داد.

در [۸] *Feinstein* و *Schnackenberg* با استفاده از روش‌های آماری و محاسبه و استخراج توزیع و بی‌نظمی موجود در صفات مختلف بسته‌ها، حمله *DDoS* را تشخیص داده‌اند. آن‌ها نشان داده‌اند که ناهنجاری رخ داده در حملات *DDoS*، با استفاده از صفات انتخاب شده، قابل تشخیص می‌باشد.

بسیاری از تحقیق‌های انجام شده روی سیستم‌های تشخیص تهاجم شبکه‌ای، سعی در استفاده از شبکه‌های عصبی مصنوعی برای این منظور داشته‌اند. این روش‌ها پس از آموزش رفتارهای نرمال یا غیر نرمال ناشی از رخداد تهاجم و یا ترکیب این دو، می‌توانند تهاجم را تشخیص دهند. برای این منظور تا کنون هم از شبکه‌های عصبی با ناظر همچون *MLFF*، *Recurrent*، *Adaptive* استفاده شده است و هم از شبکه‌های عصبی بدون ناظر از قبیل *SOM* و شبکه‌های *ART* برای این منظور بهره گیری شده است. شبکه‌های عصبی با ناظر با تغییر الگوهای ورودی، جهت تحلیل ورودی‌های جدید، نیاز به آموزش مجدد دارند، این در حالی است که شبکه‌های بدون ناظر با قابلیت تطبیق‌پذیری[۴] بالای خود، قادرند به صورت پویا توان تحلیلی خود را بر اساس ورودی های جدید بالا ببرند.[۹]

اکثر تحقیقات انجام شده در زمینهٔ به‌کارگیری شبکه های عصبی بدون‌ناظر در تشخطص تهاجم، از شبکه‌های مصنوعی *SOM* استفاده نموده‌اند و تنها تعداد کمی از آن‌ها از شبکه‌های *ART* برای این منظور استفاده نموده‌اند و این در حالی است که شبکه‌های *ART* از قابلیت‌های بیشتری نسبت به شبکه‌های *SOM* برخوردار می‌باشند.

برای اولین بار، *Cannady* در [۱۰] از شبکه‌های عصبی بدون‌ناظر *SOM* چندگانه برای تشخیص تهاجم در شبکه‌های کامپیوتری استفاده نمود. در این سیستم پردازش و تحلیل ترافیک شبکه بر عهدهٔ مجموعه‌ای از نقشه‌های کوهونن می‌باشد که هر نقشه موظف است بسته‌های مربوط به یک پروتکل خاص شبکه را تحلیل نماید. *Gardian* در [۱۱] از *SOM* برای بصری‌سازی فعالیت شبکه استفاده نمود و بدین ترتیب راه جدیدی را برای مدیران شبکه جهت کاوش، پی‌گیری و تحلیل مهاجمان، با تکیه بر فاکتورهای انسانی، فراهم نموده است.

اغلب تحقیقات جدید انجام شده در این زمینه، سعی در استفاده از بیش از یک شبکهٔ عصبی در ساختاری سلسله مراتبی داشته‌اند. افزایش دقت و صحت در دسته‌بندی الگوها مهم‌ترین مزیت استفاده از آن است. از این جمله در [۱۲] از شبکه *SOM* به صورت سلسله مراتبی در ساختاری دولایه برای تشخیص تهاجم بر اساس توالی از اتصالات شبکه، استفاده شده است. تأکید این سیستم، بیشتر برروی در نظر گرفتن زمان و توسعه تدریجی سلسله مراتب در آن می‌باشد. سیستم معرفی شده در [۱۳]، با نام *NSOM*، نیز از *SOM* ساختیافته برای دسته‌بندی داده‌های برخط شبکه، استفاده می‌کند. این سیستم می‌تواند حملات *DoS* را در برابر ترافیک نرمال، به طور گرافیکی دسته‌بندی نماید.

در روش ارائه شده در این مقاله، بر اساس تجربیات حاصله از کارهای انجام شده در این زمینه از ترکیب روش‌های آماری و شبکه‌های عصبی بدون‌ناظر در تشخیص حملات از کاراندازی سرویس توزیع‌شده استفاده شده است. در سیستم طراحی شده، با استفاده از پیش‌پردازش آماری ترافیک شبکه، ویژگی‌هایی که بتوانند رفتار حملات *DDoS* را نشان دهند، استخراج شده‌اند. سپس با استفاده از شبکه عصبی *ART1*، این ویژگی‌ها تحلیل و در دو گروه نرمال و حمله دسته‌بندی شده‌اند.

---

۱ State Action

۲ machine learning algorithm

۳ fundamental on information theory

۴ Adaptability

۳- از *IDS* چندمنظوره تا *DDoS-IDS* خاص منظوره

سیستم‌های تشخیص حمله‌ی مبتنی بر شبکه، بر مبنای الگوی اتصال و یا بسته عمل می‌کنند. این سیستم‌ها با بررسی الگوی یک اتصال و یا بسته سعی در تشخیص حمله دارند.

در حمله *DDoS* مهاجم با ارسال بسته‌هایی شبیه به بسته‌های نرمال سعی در سیلابی نمودن ترافیک ماشین هدف دارند. در نتیجه سیستم‌های تشخیص حمله همه‌منظوره موفقیت چندانی در تشخیص حملات *DDoS* ندارند. در حمله *DDoS* مهاجم با ارسال بسته‌های مختلف با الگوی نرمال بطور همزمان از صدها و شاید هزاران مبدأ متفاوت، ترافیک شبکه ماشین هدف را آن‌قدر سنگین می‌کنند، تا این ماشین دیگر نتواند به درخواست‌های کاربران شبکه خود پاسخ دهد.

ایده اصلی این مقاله در تشخیص این‌گونه حملات این است که مهاجم در حمله *DDoS* هر چقدر هم که بتواند بسته‌های نرمال به سمت ماشین هدف ارسال نماید، باز نمی‌تواند ترافیکی شبیه به ترافیک نرمال ماشین هدف روی آن ایجاد کند. به عنوان مثال یک بسته *SYN* که برای باز نمودن اتصال *TCP* به کارگزاری می‌رسد، به تنهایی نرمال است و کارگزار دریافت کننده موظف است اقدامات لازم برای برقراری یک اتصال را انجام دهد. ولی هنگام حمله *SYNFlood* ترافیک شبکه به صورتی است که تعداد زیادی بسته *SYN* به طور ناگهانی به قربانی می‌رسد، که این خود باعث بروز ناهنجاری در ترافیک حمله *SYNFlood* نسبت به ترافیک نرمال می‌شود. هر کارگزار بسته به سرویس‌هایی که در اختیار کاربران خود قرار می‌دهد، ترافیکی مخصوص به خود دارد. هم‌چنین ترافیک مربوط به یک کارگزار در ساعات مختلف شبانه‌روز بسته به سرویس‌هایی که در این ساعات می‌دهد، ممکن است متفاوت باشد. در نتیجه ترافیکی که مهاجم حمله *DDoS* روی یک کارگزار ایجاد می‌کند، معمولا تفاوت فاحشی با ترافیک نرمال آن خواهد داشت.

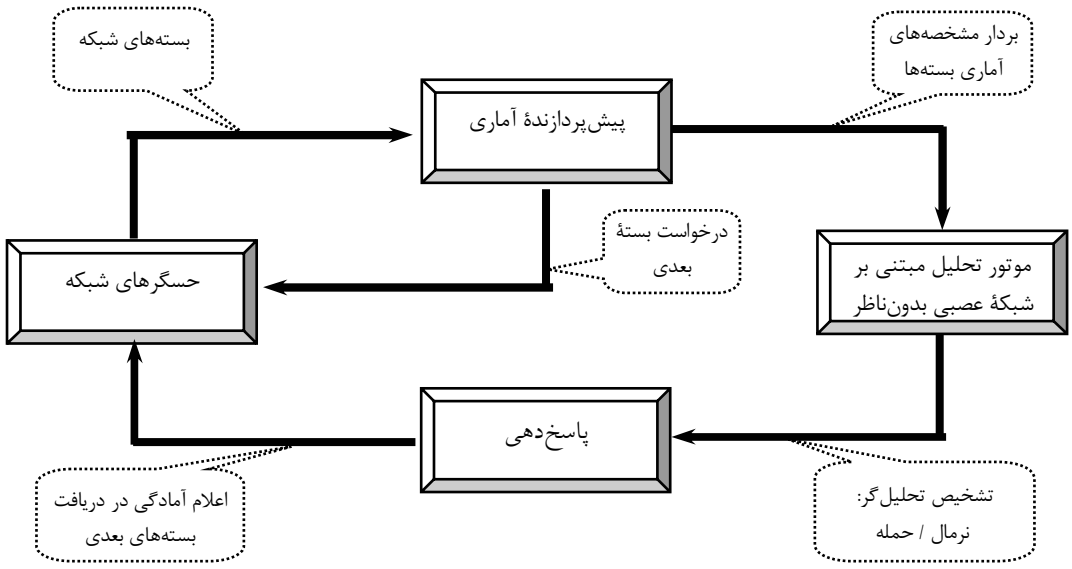
در این مقاله برای تشخیص حمله *DDoS* ویژگی‌های آماری مختلف در یک بازه زمانی مشخص، در ترافیک حمله و ترافیک نرمال بررسی شده‌اند و مشاهده شده است که مقادیر این ویژگی‌ها در ترافیک حمله و ترافیک نرمال با هم تفاوت قابل توجهی دارند. بهره‌برداری از این تفاوت می‌تواند در تشخیص حملات *DDoS* کارگشا باشد.

با توجه به وسیع بودن دامنه ترافیک نرمال یک شبکه و سنگین و سبک بودن آن در زمان‌های مختلف، مقادیر ویژگی‌های آماری استخراج شده، چه در ترافیک حمله و چه در ترافیک نرمال یک شبکه می‌تواند دامنه بسیار وسیعی داشته باشد. شبکه عصبی با امکان تحلیل و دسته‌بندی قوی، یکی از بهترین ابزارهایی است که می‌توان برای این منظور استفاده نمود.

اکثر روش‌های پیشنهادی برای تشخیص حملات *DDoS*، با بررسی ترافیک شبکه سعی در کشف ناهنجاری بوجود آمده در اثر حمله *DDoS* می‌کنند. این روش‌ها ابتدا ترافیک شبکه را به صورت‌های مختلف (که معمولا پایه ریاضی و آماری دارند) مدل می‌کنند. سپس با در نظر گرفته مقادیر آستانه برای پارامترهای مختلف مدل خود، در صورتی که در یک مدل مقادیر پارامترهای تعیین شده از مقادیر آستانه تعریف شده فراتر رود، حمله را تشخیص می‌دهند. تعیین دقیق این مقادیر آستانه با توجه به پیشرفت مهاجمان در طراحی ابزارهای تولید حمله نمی‌تواند بصورت دقیق صورت گیرد. این در حالی است که با استفاده از شبکه عصبی برای تشخیص این‌گونه حملات، دغدغه تعیین مقدار آستانه مناسب وجود ندارد و این مهم به خود شبکه واگذار می‌گردد.

۴- *SPUNNID*

برای به‌کارگیری قابلیت‌های شبکه‌های عصبی بدون‌ناظر در تشخیص حملات *DDoS*، از سیستم *SPUNNID* که تعمیم‌یافتهٔ سیستم *UNNID* [۲] می‌باشد، استفاده نموده‌ایم.



شکل ۱- نمودار جریان داده ها در سیستم SPUNNID

همانطور که در شکل ۱ می‌بینید، در سیستم *SPUNNID*، مؤلفهٔ تأمین‌کنندهٔ داده‌ها، داده‌های مورد نیاز سیستم را توسط مؤلفه‌ی *حسگرهای شبکه*، به صورت برخط یا برون از خط از شبکه و یا فایل ممیزی شبکه دریافت و به مؤلفهٔ پیش‌پردازندهٔ آماری می‌دهد.
پیش‌پردازندهٔ آماری پس از دریافت و جمع‌آوری بسته‌های مربوط به یک بازهٔ زمانی (که از قبل توسط مدیر سیستم بازهٔ آن تعریف شده است) مشخصه‌های آماری مورد نیاز را استخراج می‌نماید. بردار مشخصه‌های آماری استخراج شده بر حسب نوع شبکهٔ عصبی مورد استفاده در موتور تحلیل‌گر سیستم، به فرمت دودویی و یا نرمال‌شده (در بازهٔ صفر و یک) تبدیل شده و به موتور تحلیل‌گر مبتی بر شبکهٔ عصبی بدون‌ناظر ارسال می‌گردد. موتور تحلیل‌گر بر اساس درخواست مدیر سیستم، یا از داده‌ها برای آموزش و تست سیستم استفاده می‌نماید و یا با دریافت اطلاعات آماری مربوط به ترافیک شبکه در یک بازهٔ زمانی، براساس تجربیات حاصله در فاز آموزش، نرمال و یا تهاجمی بودن آن را مشخص می‌نماید و خروجی حاصله را به مؤلفهٔ پاسخ‌دهی ارسال می‌نماید. در مؤلفهٔ پاسخ‌دهی، در صورتی‌که خروجی دریافتی از موتور تحلیل‌گر، نشانگر یک حمله یا تهاجم *DDoS* باشد، اطلاعات مربوط به ترافیک مورد نظر همراه با خروجی حاصله از تحلیل‌گر ثبت و اخطار مناسب به طرق مختلف تولید می‌گردد.

۴-۱- موتور شبکه عصبی

یکی از مؤلفه‌های اصلی سیستم *SPUNNID* موتور تحلیل‌گر مبتنی بر شبکهٔ عصبی بدون‌ناظر آن می‌باشد. در موتور تحلیل‌گر می‌توان از شبکه‌های عصبی بدون‌ناظر مختلفی استفاده نمود که در این مرحله با توجه به کارایی بالاتر شبکهٔ عصبی *ART-1 [۱]* در تشخیص تهاجم در مقایسه با دیگر انواع شبکه‌های عصبی بدون‌ناظر *[۱۴]*،در موتور تحلیل‌گر سیستم *SPUNNID* از آن استفاده نموده‌ایم.
در شبکه‌های *ART* بدون‌ناظر، الگوهای ورودی ممکن است چندین بار و با ترتیب مختلف به شبکه داده شوند. در هربار که یک الگو به عنوان ورودی به شبکه داده می‌شود، یک واحد خوشهٔ[۱] مناسب انتخاب و وزن‌های مربوط به آن تغییر می‌یابند. در این شبکه‌ها انتخاب واحد برنده بر اساس تفاضل مطلق بردارها (که در شبکه‌های *SOM* مطرح است) صورت نمی‌پذیرد، بلکه شباهت نسبی بردار ورودی با بردار وزن یک واحد خوشه، معیار انتخاب واحد برنده می‌باشد [۱].
شبکه‌های*ART* به گونه‌ای طراحی شده‌اند که به کاربر این امکان را می‌دهند که درجهٔ شباهت الگوهایی که در یک خوشه قرار می‌گیرند را با تنظیم پارامتر هوشیاری[۲] کنترل نماید. هم‌چنین در این شبکه‌ها نیازی نیست که تعداد خوشه‌ها از قبل تعیین شده باشد و می‌توان در آن‌ها از پارامتر هوشیاری برای تعیین تعداد مناسب خوشه‌ها استفاده نمود، چرا که هر چه‌قدر این پارامتر افزایش یابد، تعداد خوشه‌ها نیز افزایش و هر چه قدر کاهش یابد، تعداد خوشه‌ها نیز کاهش می‌یابد. در این کاربرد، تعداد خوشه‌ها باید به اندازه‌ای تعیین گردد که انواع مختلفی از حملات *DDoS* که تا حدودی شبیه به هم هستند در یک خوشهٔ یکسان قرار نگیرند.
علاوه بر خصوصیت اساسی فوق، شبکه‌های *ART* دو خصوصیت مهم دیگر را نیز دارا می‌باشند که عبارتند از:

- پایداری[۳]: یعنی عدم نوسان یک الگو در مراحل مختلف آموزش بین خوشه‌های مختلف.
- انعطاف‌پذیری[۴]: یعنی توانایی شبکه در یادگیری الگوهای جدید در هر یک از مراحل یادگیری.

پایداری و انعطاف‌پذیری شبکه‌های *ART* و قابلیت آن‌ها در دسته‌بندی الگوهای ورودی با میزان شباهت تحت کنترل کاربر، آن‌ها را مناسب‌تر از سایر شبکه‌های عصبی بدون‌ناظر برای استفاده در سیستم‌های تشخیص تهاجم نموده است. به همین دلیل در سیستم*SPUNNID* از شبکهٔ *ART-1* که اولین شبکهٔ مطرح در مجموعه شبکه‌های *ART* می‌باشد، استفاده نموده‌ایم. شبکهٔ *ART-1* تنها قادر است داده‌های دودویی را پردازش نماید.
در سیستم *SPUNNID*، بردار ورودی دودویی توسط مولفهٔ پیش پردازندهٔ آماری از مشخصه‌های آماری استخراج شده از ترافیک شبکه، فراهم شده و به شبکهٔ عصبی در موتور تحلیل‌گر سیستم داده می‌شود. در فاز آموزش، ابتدا داده‌های ورودی بدون توجه به ماهیت آن‌ها (نرمال و یا حمله) توسط شبکهٔ *ART-1* دسته‌بندی می‌گردند. بعد از اتمام آموزش، سیستم باید واحدهای مربوط به هر یک از خوشه‌ها را مشخص و با استفاده از ماهیت از قبل تعیین شدهٔ داده‌های آموزشی، نام مناسبی را به آن‌ها اختصاص دهد. نام هر خوشه معادل نام واحدهایش بوده و نام هر واحد معادل است با نوع اکثریت داده‌هایی که با اِعمال آن‌ها به ورودی سیستم، واحد مورد نظر به عنوان واحد برنده انتخاب گردیده است. نتیجه این کار ایجاد یک نقشه خوشه‌بندی[۵] می‌باشد. دراین نقشه تعدادی از واحدها با یکدیگر تشکیل خوشه‌ای را می‌دهند که بیانگر ترافیک نرمال است. تعدادی دیگر با یکدیگر تشکیل خوشه‌ای را می‌دهند که بیانگر یکی از انواع حملات شناخته شدهٔ *DDoS* می‌باشد و بقیه نیز تشکیل خوشه‌ای را می‌دهند که نه بیانگر ترافیک نرمال می‌باشد و نه نشانگر ترافیک حملات شناخته شده و لذا بیان‌گر ترافیک حملات جدید *DDoS* می‌باشد.
باتوجه به توضیح فوق برای آموزش موتور تحلیل‌گر مبتنی بر شبکه عصبی نیاز داریم که از هر دو ترافیک نرمال و حمله، در این سیستم استفاده نماییم تا بوسیلهٔ آن توانایی تشخیص انواع حملات تولید شدهٔ *DDoS* و هم‌چنین حملات جدید روی‌داده در شبکه را داشته باشیم. بنابراین سیستم *SPUNNID* دو روش تشخیص سوء استفاده و تشخیص ناهنجاری را با به‌کارگیری شبکه‌های عصبی بدون‌ناظر با یکدیگر ترکیب نموده است و لذا می‌تواند مزایا و فواید هر دو روش را در تشخیص

---
۱ Cluster Unit
۲ Vigilance
۳ Stability
۴ Plasticity
۵ Clustering Map

حملات شناخته شده و حملات جدید درخود دارا باشد.

۴-۲- پیش‌پردازندهٔ آماری

یک حمله *DDoS* ممکن است شامل حملات از کار اندازی سرویس سیلابی از جمله *UDP Flood*، *SYN Flood*، *ICMP Flood* و *SMURF* و ... یا ترکیبی از انواع حملات فوق باشد. در نتیجه هنگام وقوع این حمله بسته به نوع آن، در تعداد بسته‌های مختلف یک شبکه تغییر ناگهانی ایجاد می‌شود. این تغییر در یک ترافیک کاملا نرمال ولی سنگین نیز می‌تواند رخ دهد. ولی نرخ بسته‌های مختلف در ترافیک نرمال، خواه ترافیک، سنگین باشد و خواه سبک، تقریبا یکسان است. این در حالی است که این مقادیر در هنگام وقوع حمله تفاوت فاحشی با ترافیک نرمال خواهد داشت. به عنوان مثال اگر یک حمله *DDoS* فقط از حمله *UDP Flood* استفاده کند، نرخ بسته‌های *UDP* بسیار بیشتر از حالت نرمال خواهد شد. حتی اگر در یک حمله *DDoS*، از ترکیب انواع مختلف حملات از کار اندازی سرویس سیلابی استفاده شود باز هم نرخ بسته‌ها با مقادیر مشابه در ترافیک نرمال تفاوت خواهد داشت. در نتیجه در این مقاله ویژگی‌های استخراج شده از مجموعه بسته‌های موجود در یک بازه زمانی عبارتند از:

- N_{TCP}: درصد بسته‌های *tcp*
- N_{ICMP}: درصد بسته‌های *icmp*
- N_{UDP}: درصد بسته‌های *udp*
- N_{TCPSYN}: درصد بسته‌های *syn* در بسته‌های *tcp*
- $N_{TCPSYNACK}$: درصد بسته‌های *syn+ack* در بسته‌های *tcp*
- N_{TCPACK}: درصد بسته‌های *ack* در بسته‌های *tcp*
- $A_{PacketHeaderSizes}$: متوسط اندازهٔ سرآیند بسته‌ها
- $A_{packetDataSizes}$: متوسط اندازهٔ بخش داده‌ای بسته‌ها

بردار ورودی به موتور تحلیل‌گر مبتنی بر شبکه عصبی با تبدیل مقادیر این ویژگی‌ها به فرم دودویی تشکیل می‌شود.

شکل ۲- عملکرد پیش‌پردازنده آماری، مشابه سیستم اتوبوس‌رانی می‌باشد.

عمل‌کرد پیش‌پردازنده آماری، مشابه سیستم اتوبوس‌رانی می‌باشد. در فاصله‌های زمانی معین، اتوبوس در ایستگاه توقف می‌کند و مسافران (بسته‌های) جمع شده در ایستگاه را به مقصد (مرکز استخراج ویژگی‌ها) می‌رساند. در مرکز استخراج ویژگی‌ها، بسته‌های رسیده پردازش می‌شوند و ویژگی‌های تعیین شده، استخراج می‌گردند. این ویژگی‌ها، جهت تحلیل به مؤلفه تحلیل‌گر *SPUNNID* ارسال می‌شوند.

۵- ارزیابی

مهم‌ترین عامل در بالا بردن کارایی شبکهٔ عصبی، کامل بودن داده‌های فاز آموزش است. به این معنی که این داده‌ها باید بتوانند حتی‌الامکان انواع مختلف رفتارهای نرمال را در خود دارا باشند. در نتیجه برای تهیه داده‌های فاز آموزش در روش ارائه شده در این مقاله، باید ترافیک نرمال یک کارگزار برای آموزش شبکه‌ی عصبی در ساعات مختلف شبانه‌روز جمع‌آوری شود تا این ترافیک بتواند زمان شلوغی و زمان خلوتی شبکه و سرویس‌های مختلفی که ممکن است بسته به زمان، این کارگزار به مشتریان بدهد را پوشش دهد. علاوه بر این داده‌های ترافیک فاز آموزش باید به مقدار کافی شامل ترافیک حمله‌های *DDoS* مختلفی که ممکن است روی کارگزار رخ دهد نیز باشد. برای این منظور در بخش جمع‌آوری داده‌ها، یک روش مناسب برای جمع‌آوری و تولید داده‌های آموزش و آزمون سیستم *SPUNNID* ارائه شده است.

پس از جمع‌آوری داده‌های آموزش و آزمون، کارایی تشخیص سیستم *SPUNNID* بر اساس چهار معیار مورد ارزیابی قرار گرفت که در بخش نتایج ارزیابی، حاصل این امر آمده است. این چهار معیار ارزیابی عبارتند از: درصد تشخیص صحیح نوع (تفکیک صحیح ترافیک نرمال از حمله و تشخیص صحیح نوع حملهٔ شناخته شده در موارد رویداد آن) که به اختصار آن را *ETTR* نامیم.

- درصد تشخیص درست (تنها تفکیک صحیح ترافیک نرمال از حمله بدون در نظر گرفتن نوع حملهٔ تشخیص داده شده) که به اختصار آن را *TR* گوییم.
- درصد خطای مثبت غلط (تشخیص حمله درحالیکه حمله‌ای روی نداده) که به اختصار آن را *FPR* گوییم.
- درصد خطای منفی غلط (عدم تشخیص حمله در زمان بروز حمله) که به اختصار آن را *FNR* گوییم.

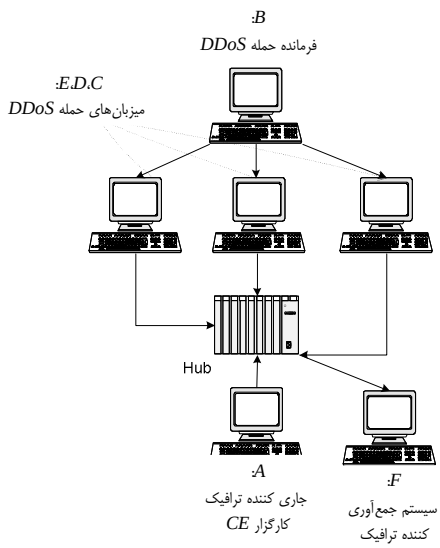
۵-۱- جمع‌آوری داده‌های آموزش و آزمون

تولید حمله *DDoS* روی یک کارگزار نوعی و سپس جمع‌آوری داده‌ها، بدلیل ماهیت این نوع حمله، کار بسیار پرهزینه‌ای است. از طرفی، گوناگونی داده‌های فاز آموزش شبکه عصبی در بالا بردن کارایی آن نقش عمده‌ای دارند. با توجه به این موارد، در این مقاله سعی شده است که حتی‌الامکان داده‌های جمع‌آوری شده، مشابه داده‌های محیط واقعی باشند. برای جمع‌آوری داده‌های آزمون مناسب، یک شبکه محلی مطابق با توپولوژی نشان داده شده در شکل ۳، ساخته شد. برای استفاده از این توپولوژی، ابتدا لازم است نمونه‌هایی از ترافیک نرمال کارگزار مورد نظر به اندازه‌ای که بتواند ترافیک کارگزار را در ساعات مختلف شبانه‌روز تحت پوشش قرار دهد، جمع‌آوری شود. برای این منظور در این مقاله ترافیک کارگزار *CE*[1] جمع‌آوری شده است.

توپولوژی ارائه شده در شکل ۳ از ۶ کامپیوتر که پنج تای آن‌ها توسط یک هاب به هم متصل شده‌اند، تشکیل شده است. ماشین *A* ترافیک جمع‌آوری شدهٔ ماشین *CE* را در شبکه، به جریان می‌اندازد. ماشین *B* به

[1] کارگزار دانشکده کامپیوتر دانشگاه صنعتی شریف

عنوان ماشین فرمانده در حمله *DDoS*، دستور آغاز حمله را به ماشین‌های *C*، *D* و *E* صادر می‌کند. این ماشین‌ها نیز به محض دریافت این دستور، حملات خود را به سمت سیستم نمایندهٔ کارگزار *CE* آغاز می‌کنند. از ابتدای شروع به کار این شبکه، ماشین *F* ترافیک‌های در جریان توسط ماشین‌های میزبان حمله *DDoS* و ماشین *CE* را به صورت ترکیبی جمع‌آوری می‌کند. اگر در این شبکه حمله‌ای در جریان نباشد، ماشین *F* فقط ترافیک ماشین *CE* را جمع می‌کند. به این ترتیب در نهایت ترافیک جمع‌آوری شده توسط ماشین *F* هم شامل ترافیک نرمال ماشین *CE* و هم شامل ترافیک حمله *DDoS* ایجاد شده روی این ماشین خواهد بود.



شکل ۳- توپولوژی شبکه جمع‌آوری داده‌های شبکه عصبی.

برای تولید داده‌های آموزشی، ترافیک ماشین *CE* برای مدت زمان نسبتا طولانی جمع‌آوری شد. سپس با استفاده از روش مذکور، چند حمله *DDoS* مختلف روی ترافیک جمع‌آوری شده ایجاد شد. برای آزمون شبکه آموزش داده شده نیز، ترافیک آزمون با استفاده از همین روش؛ با تولید حملات مختلف روی ترافیک ماشین *CE* که در زمان دیگری و با حجم کمتری جمع‌آوری شده بود، تولید شد.

۵-۲- نتایج ارزیابی

برای بالا بردن کارایی روش پیشنهادی چند پارامتر مهم دخالت دارند:

- تعداد دسته‌های شبکه *ART1*
- تعداد تکرار الگوهای آموزشی
- مقدار پارامتر هوشیاری در شبکه *ART1*
- طول بازه‌های زمانی
- مقدار پارامتر هوشیاری در فاز استفاده

▪ تعداد دسته‌های شبکه *ART1*

تعداد دسته‌ها به گوناگونی داده‌های فاز آموزش بستگی دارد. در روش ارائه شده در این مقاله، تعداد و گوناگونی داده‌های آموزش شبکه *ART1* بستگی مستقیم به طول بازه زمانی تعریف شده دارد. در نتیجه تغییر طول بازه زمانی باعث تغییر تعداد دسته‌های مورد نیاز خواهد شد. به طور کلی تعداد دسته‌های تعریف شده باید گوناگونی داده‌های فاز آموزش را بپوشاند.

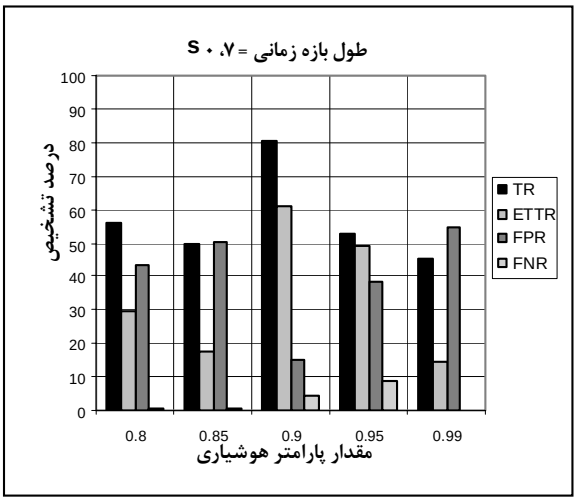
▪ تعداد دفعات آموزش در فاز آموزش *ART1*

دومین عاملی که می‌تواند روی کارایی شبکه تأثیر داشته باشد، تعیین تعداد دفعات آموزش است. آزمون‌های انجام شده روی این پارامتر در روش پیشنهادی نشان می‌دهد، عدد ۱۰۰، عدد مناسبی برای این منظور است.

▪ مقدار پارامتر هوشیاری در شبکه *ART1*

پارامتر هوشیاری مهم‌ترین پارامتری است که روی کیفیت دسته‌بندی داده‌ها، اثر می‌گذارد. این پارامتر، درجه مشابهت الگوهایی که در یک دسته قرار می‌گیرد را مشخص می‌کند.

آزمون‌های انجام شده نشان می‌دهند که نه مقدار کم این پارامتر مناسب است و نه مقدار بسیار بزرگ آن. برای بدست آوردن مقدار مناسب این پارامتر، سیستم پیشنهادی، با مقادیر مختلف برای این پارامتر ارزیابی شد. نتایج این ارزیابی در نمودار شکل ۴ نشان داده شده است. همان‌طور که این نمودار نشان می‌دهد، مقدار ۰/۹ برای این پارامتر، بالاترین کارایی را نصیب سیستم می‌نماید.



شکل ۴- نمودار تأثیر تغییر پارامتر هوشیاری در کارایی تشخیص سیستم.

▪ طول بازه زمانی

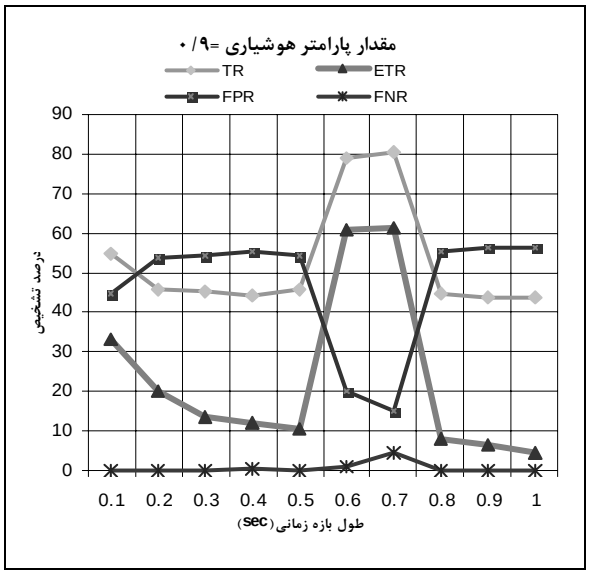
انتخاب طول بازه زمانی مناسب بستگی به حجم داده‌های آموزشی جمع‌آوری شده، وجود قدرت پردازشی کافی و کارایی مورد انتظار دارد. آزمون‌ها نشان می‌دهند که نه مقدار خیلی کم برای این منظور مناسب است و نه مقدار زیاد.

با در نظر گرفتن حجم داده مناسب، انتخاب بازهٔ زمانی کوتاه، گوناگونی داده‌های آموزش را کاهش می‌دهد، این در حالی است که در این حالت، تعداد بردارهای ورودی شبکه عصبی افزایش می‌یابد. اگر بازه زمانی بزرگ انتخاب شود، تعداد بردارهای ورودی به شبکه کاهش می‌یابد، ولی

گوناگونی آن‌ها بیشتر می‌شود. در نتیجه انتخاب بازه زمانی مناسب ارتباط مستقیم با حجم داده‌های آموزش جمع‌آوری شده دارد، ولی طول زمان آن نباید از حداکثر زمان پاسخ مورد انتظار سیستم فراتر رود؛ چرا که زمان پاسخ سیستم *SPUNNID* برابر با طول بازهٔ زمانی انتخاب شده می‌باشد. در نتیجه اگر طول بازه زمانی افزایش یابد، زمان پاسخ نیز افزایش می‌یابد. این در حالی است که افزایش زمان پاسخ، در شرایطی که باعث به تأخیر افتادن اقدامات لازم در صورت بروز حمله شود، چندان خوشایند نیست. در حالت کلی به نظر می‌رسد طول بازه زمانی نباید بیشتر از دو دقیقه در نظر گرفته شود.

اگر حجم داده‌های آموزشی جمع‌آوری شده، به اندازه کافی بزرگ نباشد، انتخاب بازه زمانی کوتاه‌تر نیز می‌تواند کارایی قابل قبولی داشته باشد. طول بازه زمانی، حداقل باید برابر با مقداری باشد که با استفاده از ویژگی‌های انتخاب شده، از روی مجموعه بسته‌های قرار گرفته در یک بازه، رفتار حمله مشخص باشد. به عنوان مثال اگر طول بازه زمانی به قدری کوچک باشد که فقط دو بسته در آن قرار بگیرند، از روی این دو بسته نمی‌توان به راحتی وقوع حمله را تشخیص داد.

بطور کلی، انتخاب طول بازه زمانی مناسب بستگی مستقیم به حجم داده آموزش دارد. هم‌چنین با بیشتر شدن حجم داده آموزش، مدت زمان آموزش شبکه عصبی نیز افزایش می‌یابد و نیاز به قدرت پردازشی بیشتری دارد.

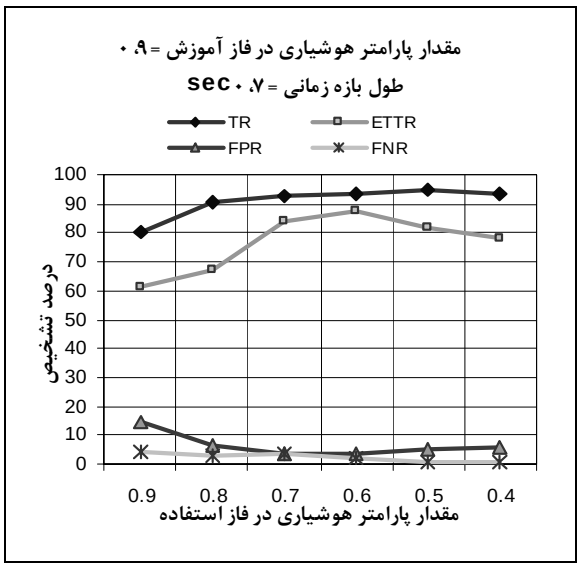


شکل ۵- تأثیر تغییر طول بازه زمانی در کارایی تشخیص سیستم.

در این مقاله تأثیر طول بازه زمانی، برای حدود ۳۰۰ مگا بایت داده آموزشی جمع‌آوری شده، ارزیابی شد. نتیجه این ارزیابی در نمودار شکل ۵ نشان داده شده است. این نمودار نشان می‌دهد که با در نظر گرفتن طول بازه زمانی به مدت۰/۷ ثانیه برای داده‌های آموزشی جمع‌آوری شده، بهترین کارایی حاصله ۸۰/۶۱٪ می‌باشد.

- **مقدار پارامتر هوشیاری در فاز استفاده**

یکی از نقاط ضعف سیستم‌های تشخیص تهاجم تجاری مبتنی بر سیستم‌های خبره، عدم توانایی آن‌ها در تشخیص حملات شناخته شدهٔ تغییر شکل یافته و حملات جدید است. در این مقاله برای بالا بردن کارایی و انعطاف‌پذیری سیستم *SPUNNID*، نشان داده شده است که با کاهش مقدار پارامتر هوشیاری در فاز استفاده، حساسیت دسته‌بندی شبکه عصبی بدون ناظر کاهش می‌یابد و در نتیجه تعداد پیام‌های *"Cannot cluster"* کاهش و انعطاف‌پذیری سیستم افزایش می‌یابد. این تغییر در اغلب موارد باعث می‌شود الگوهای حمله و نرمال تغییر یافته نیز به درستی دسته‌بندی شوند. نمودار شکل ۶ نشان می‌دهد که در شرایطی که طول بازه زمانی برابر ۰/۷ انتخاب شده باشد و مقدار پارامتر هوشیاری در فاز آموزش نیز برابر ۰/۹ باشد، انتخاب پارامتر هوشیاری در فاز استفاده برابر ۰/۵، می‌تواند کارایی را تا حدود ۹۴/۷٪ بهبود بخشد.



شکل ۶- تأثیر کم نمودن مقدار پارامتر هوشیاری در فاز استفاده.

۶- بالا بردن کارایی *SPUNNID* با بهینه‌سازی مؤلفهٔ پاسخ‌دهی

مدیران سیستم با تشخیص وقوع حملهٔ *DDoS*، بلافاصله باید اقدامات لازم برای جلوگیری از منع سرویس بیشتر را انجام دهند. یکی از مهم‌ترین اقدامات در این زمان، مسدود نمودن آدرس‌های شبکه مشکوک می‌باشد. در نتیجه یک تشخیص غلط منجر به مسدود شدن آدرس‌های شبکه زیادی می‌شود، که این خود باعث از کار اندازی سرویس کاربران این آدرس‌ها می‌شود.

زمان پاسخ *SPUNNID* برابر با طول بازه زمانی انتخاب شده است. همان‌طور که نتایج ارائه شده در بخش قبل نشان می‌دهد، معمولا این زمان در حدود چند دهم ثانیه است. این در حالی است که حتی اگر در این زمان اندک، حمله‌ای رخ داده شده باشد و سیستم نیز آن را به درستی تشخیص دهد، نباید آن را به حساب شروع یک حمله طولانی مدت پنداشت و اقدامات مقتضی را انجام داد. بلکه بهتر است تشخیص دقیق را از روی چند خروجی آخر تحلیلگر سیستم انجام داد. به این ترتیب که اگر درصد بالایی از این خروجی های آخر حمله تشخیص داده شده باشد، در

آن صورت به احتمال خیلی قوی حمله رخ داده است و تشخیص نهایی اعلام می‌شود.

این امر می‌تواند در *SPUNNID*، با تغییر جزئی مؤلفهٔ پاسخ‌دهی سیستم محقق گردد. برای این منظور در این مؤلفه، یک پنجره به طول M در نظر گرفته می‌شود، این پنجره، در هر زمان شامل M خروجی آخر تحلیلگر سیستم می‌باشد، با این تغییر در مؤلفهٔ پاسخ‌دهی، به نحو قابل توجهی اثر خطاهای *مثبت غلط* و *منفی غلط* سیستم کاهش می‌یابد. مقایسه خروجی مورد انتظار با خروجی *SPUNNID*، در یک پنجره نشان می‌دهد که:

- خطای *مثبت غلط* به صورت نادر، در شرایطی رخ می‌دهد که ماهیت بازه‌های زمانی قبل و بعد بازهٔ مورد نظر، نرمال تشخیص داده شود.
- خطای *منفی غلط* وقتی رخ می‌دهد که در بین چند بازه زمانی حمله، یک بازه زمانی به صورت نرمال تشخیص داده شده باشد.

در نتیجه با در نظر گرفتن پنجره زمانی در تشخیص نهایی، تشخیص بازه‌های زمانی قبل و بعد از یک بازه نیز در تشخیص نهایی داخل می‌شوند، که این امر خود اثر خطاهای سیستم را کاهش می‌دهد.

۶-۱- ارزیابی کارایی سیستم با مؤلفه پاسخ‌دهی بهبود یافته

مهم‌ترین پارامتر در رسیدن به کارایی بالاتر در استفاده از مؤلفه پاسخ‌دهی بهبود یافته، اندازه پنجره؛ یعنی همان عدد M می‌باشد. اندازه پنجره حداکثر می‌تواند برابر مقداری قرار گیرد، که زمان پاسخ سیستم از زمان مورد انتظار مدیران سیستم بیشتر نشود.

همان‌طور که در بخش ارزیابی آمده است، با در نظر گرفتن طول بازه زمانی برابر ۰/۷ ثانیه، بالاترین کارایی برای سیستم حاصل گردیده است. در نتیجه برای داشتن زمان پاسخ برابر یک دقیقه، می‌توان اندازه پنجره، یعنی عدد M را برابر ۸۵ در نظر گرفت. این عدد از تقسیم ۶۰ ثانیه بر بهترین طول بازهٔ زمانی یعنی ۰/۷ ثانیه حاصل گردیده است. در این صورت، تشخیص نهایی با در نظر گرفتن ۸۵ پاسخ آخر مؤلفه‌ی تحلیل‌گر سیستم صادر می‌گردد.

پارامتر دیگری که در این‌جا اهمیت دارد، نحوه پردازش ۸۵ پاسخ آخر است. ساده‌ترین روش پردازش، این است که در صورتی که تعداد بازه‌های زمانی که حمله تشخیص داده شده‌اند، از عدد خاص D بیشتر شود، حمله تشخیص داده شود. این عدد می‌تواند به صورتی انتخاب شود که بتواند خطای *منفی غلط* موجود را کاهش دهد و در حد امکان آن را محو نماید.

بررسی خروجی مؤلفه‌ی تحلیل‌گر در حالتی که می‌باید همه‌ی بازه‌ها به صورت حمله تشخیص داده می‌شدند، نشان می‌دهد که در حدود ۰/۵۶٪ از تشخیص‌ها به دلیل وجود خطای *منفی غلط*، به اشتباه نرمال تشخیص داده شده‌اند. در نتیجه مقدار عدد D در این حالت باید حداکثر برابر با ۹۹٪ باشد. از آنجایی‌که اندازه متوسط خطای منفی غلط، برابر با ۰/۵۶٪ است، ارزیابی‌ها روی داده‌های آزمون جمع‌آوری شده، نشان می‌دهد که اگر D برابر ۹۵٪ انتخاب شود، تا حدود ۹۹٪ می‌توان وقوع حملات *DDoS* مدت‌دار (به طول حداقل یک دقیقه) و مضر را بدرستی تشخیص داد.

۷- نتیجه‌گیری

در این مقاله سیستم *SPUNNID* برای تشخیص حملات *DDoS* معرفی شد. سیستم *SPUNNID* بر مبنای یک پیش‌پردازندهٔ آماری و یک موتور تحلیل‌گر مبتنی بر شبکه‌های عصبی بدون ناظر بنا شده است. در این سیستم، با در نظر گرفتن بازه‌های زمانی کوچک، از مجموعه بسته‌های موجود در هر بازه، ویژگی‌های آماری نشان دهنده رفتار حملات *DDoS* استخراج می‌شود و سپس این ویژگی‌ها با استفاده از شبکه عصبی بدون ناظر *ART1* تحلیل و دسته‌بندی می‌شوند. ارزیابی‌های انجام شده روی این سیستم نشان می‌دهد که با استفاده از آن می‌توان تا ۹۴/۹ موارد حملات *DDoS* را بدرستی تشخیص داد.

هم‌چنین با توجه به ماهیت حملات *DDoS* نشان داده شد که اگر نتیجه تشخیص نهایی حمله به جای این‌که بر اساس یک خروجی تحلیل‌گر سیستم باشد، بر اساس چند خروجی اخیر حاصل گردد، می‌توان کارایی سیستم را در تشخیص این‌گونه حملات تا حدود ۹۹٪ بهبود بخشید.

برای تمایز بین ترافیک حمله *DDoS*، و ترافیک بسیار سنگین یک کارگزار، می‌توان از این نکته استفاده نمود که در حمله *DDoS*، ترافیک شبکه به طور ناگهانی با استفاده از بسته‌های مشابه افزایش می‌یابد. در نتیجه به نظر می‌رسد اگر به جای این‌که ویژگی‌های استخراج شده از یک بازه زمانی، بردار ورودی به شبکه عصبی را تشکیل دهند، ویژگی‌های استخراج شدهٔ چند بازه آخر، بردار ورودی شبکه عصبی را تشکیل دهند، به نوعی همبستگی و مشابهت بسته‌ها نیز در تشخیص حملات *DDoS* دخیل می‌شوند که به این ترتیب می‌توان به نتایج بهتری در تشخیص این‌گونه حملات دست یافت. در ادامه تحقیقاتمان قصد داریم با اعمال تغییر فوق در سیستم *SPUNNID*، میزان افزایش کارایی سیستم را مورد ارزیابی و تحلیل قرار دهیم.

۸- مراجع

[1] L. Fauset , "Fundamentals of Neural Networks", Prentice-Hall, 1994.

[2] M. Amini, and R. Jalii, "Network-Based Intrusion Detection Using Unsupervised Adaptive Resonance Theory (ART) ", Proceedings of the 4th Conference on Engineering of Intelligent Systems (EIS 2004), Madeira, Portugal, 2004.

[3] Ryo Kaizaki Kenjiro Cho, Osamu Nakamura, "Detection Denial of Service Attacks Using AGURI", International Conference Telecommunications, Beijing China, June 2002.

[4] R. Bazek, H. Kim, B.Rozovskii, and A. Tartakovsky, "A novel approach to detection of denial-of-service attacks via adaptive sequential and batch-sequential change-point methods", IEEE

Systems, Man and Cybernetics Information Assurance Workshop, June 2001.

[5] T. M. Gil and M. Poletter. "Multops: a data-structure for bandwidth attack detection", In Proceedings of USENIX Security Symposium'2001, 2001.

[6] Sanguk Noh , Cheolho Lee ,Gihyun Jung, Kyunghee Choi, "Using Inductive Learning for the Detection of Distributed Denial of Service Attacks", International Conference on Advances in Infrastructure for Electronic Business, Education, Science, Medicine and Mobile Technologies on the Internet, 2003.

[7] A.B. Kulkarni, S.F. Bush, and S.C. Evans, "Detecting Distributed Denial-of-Service Attacks Using Kolmogorov Complexity Metrics", GE Research & Development Center, technical report, 2001CRD176, February 2002.

[8] L. Feinstein, D. Schnackenberg, R. Balupar, D. Kindred, "Statistical Approaches to DDoS Attack Detection and Response", DARPA Information Survivability Conference and Exposition, 2003.

[9] J. Cannady, "Artificial Neural Networks for Misuse Detection", In Proceedings of National Information Systems Security Conference, 1998.

[10] B.C. Rhodes, J.A. Mahaffey, and J. D. Cannady, "Multiple Self-Organizing Maps for Intrusion Detection", In Proceedings of 23rd National Information Systems Security Conference, 2000.

[11] L. Girardin, "An Eye on Network Intruder-Administrator Shootouts", In Proceedings of the First USENIX Workshop on Intrusion Detection and Network Monitoring, Santa Clara, USA, 1999.

[12] P. Lichodzijewski, A. N. Zincir-Heywood, and M. I. Heywood, "Dynamic Intrusion Detection Using Self-Organizing Maps", The 14th Annual Canadian Information Technology Security Symposium, CITSS, 2002.

[13] K. Labib, and R. Vemuri, "NSOM: A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps", Networks and Security, 2002.

[14] R. Jalili, and M. Amini, "An Intrusion Detection System Based on the ART Neural Networks", Proceedings of the Fifth Conference on Intelligent Systems (CIS 2003), Ferdowsi University of Mashhad, Mashhad, Iran, 2003.