کاهش تاثیرگذاری حملات سیل آسای SYN با ارتقای دقت الگوریتم PSO توسط فیلتر موثر انطباقی

محمد مومنی[۱] – ثریا غراوی[۲] – فاطمه حورعلی[۲]
(۱) دانشجوی دکتری - گروه مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران
(۲) مربی - دانشکده برق و کامپیوتر، مجتمع آموزش عالی اسفراین، خراسان شمالی، اسفراین، ایران

تاریخ دریافت: ۱۳۹۷/۷/۱۷ تاریخ پذیرش: ۱۳۹۷/۱۲/۱۳

خلاصه: مدیریت ارتباط پروتکل TCP مستعد یک حمله‌ی کلاسیک می‌باشد که SYN-flooding نام دارد. در این حمله، مبدأ تعداد زیادی از سگمنت‌های SYN را به طعمه می‌فرستد بدون اینکه گام سوم از الگوریتم دست تکانی سه مرحله‌ای را کامل نماید. این امر سبب می‌شود منابع اختصاص یافته برای برقراری ارتباط در سیستم تحت حمله و پهنای باند شبکه به سرعت مصرف شود و در نتیجه از ادامه‌ی فعالیت باز بماند و درگیر رسیدگی به تقاضاهای بی‌مورد شود. این مقاله سیستم تحت حمله را با استفاده از تئوری صف‌بندی مدل‌سازی کرده و مسأله‌ی دفاع در برابر حملات SYN-flooding را به یک مسأله‌ی بهینه‌سازی نگاشت می‌دهد. سپس با استفاده از ترکیب فیلتر موثر انطباقی و الگوریتم PSO روش پیشنهادی خود را ارائه کرده و به حل این مسأله می‌پردازد. نتایج شبیه‌سازی نشان می‌دهد که مکانیزم دفاعی پیشنهادی از نظر میزان درخواست‌های بلوکه شده، احتمال موفقیت در برقراری ارتباط، کاهش احتمال موفقیت حمله‌کننده و همچنین استفاده‌ی بهینه از بافر اختصاص داده شده دارای کارآیی قابل ملاحظه‌ای می‌باشد.

کلمات کلیدی: حملات SYN-flooding، فیلتر موثر انطباقی، الگوریتم PSO، DoS، TCP.

Reducing the Impact of SYN Flood Attacks by Improving the Accuracy of the PSO Algorithm by Adaptive Effective Filters

Mohammad Momeni[1] - Sorayya Gharravi[2] - Fatemeh Hourali[2]
(1) PhD Student – Dept. of Computer Engineering, Yazd University, Yazd, Iran
mohamad.momeny@stu.yazd.ac.ir
(2) Indicator – Electrical and Computer College, Esfarayne Higher Education Complex, Northern Khorasan, Esfarayene, Iran
gharavi@esfarayen.ac.ir
hourali@esfarayen.ac.ir

Abstract: TCP connection management is susceptible to a classic attack called SYN-flooding. In this attack, the source sends a large number of SYN segments to the victim system, without completing the third step of the three-step handshaking algorithm. This lead to consuming the resources allocated to communicate with under attack system and bandwidth of the network quickly and, as a result, system cannot continue to work and engage in unnecessary requests. This paper models the attacked system using quadratic theory and maps the problem of defense against SYN-flooding attacks into an optimization problem. Then, using an effective adaptive filter combination with the PSO algorithm, it presents its proposed method and solves this problem. The simulation results show that the proposed defense mechanism has a significant performance in terms of the amount of blocked requests, the likelihood of success in communication, the likelihood of success of the attacker, and the optimal use of the dedicated buffer.

Index Terms: SYN Flooding Attacks, Adaptive Effective Filter, PSO Algorithm, DoS, TCP.

نویسنده مسئول: ثریا غراوی، مربی - دانشکده برق و کامپیوتر، مجتمع آموزش عالی اسفراین، خراسان شمالی، اسفراین، ایران، gharavi@esfarayen.ac.ir

۱- مقدمه

به طور کلی در شبکه‌های کامپیوتری حملاتی ناشی از عواملی از قبیل سرویس‌های فعال، پروتکل‌های استفاده شده و پورت‌های باز رخ می‌هد. تعدادی از سرویس‌ها توانایی لازم برای حملات را داشته و ضروری است مسائل امنیتی در زمان پیکربندی آنها مورد توجه قرار گیرد. حمله‌ی DoS در بین حمله‌هایی که برای از بین بردن امنیت شبکه‌های کامپیوتری انجام می‌شود، نوعی حمله است که با هدر دادن منابع سیستم هدف، قصد از کار انداختن آن را دارد. حمله‌های SYN flooding، متداول‌ترین نوع حمله‌ی DoS است که با مصرف سریع منابع اختصاص یافته برای برقراری ارتباط در سیستم تحت حمله باعث می‌شود سیستم تحت حمله از ادامه‌ی فعالیت باز بماند. جهت برنامه‌ریزی این حمله، از ضعف پروتکل TCP در برقراری ارتباط بین دوکامپیوتر در الگوریتم دست‌تکانی سه‌مرحله‌ای استفاده می‌شود. در الگوریتم دست تکانی، مبدأ یک بسته‌ی SYN به مقصد می‌فرستد و مقصد با دریافت این بسته آنرا در صف پشتیبان قرار می‌دهد. سپس یک ارتباط نیمه‌باز ایجاد کرده و یک بسته‌ی SYN-ACK به مبدأ می‌فرستد. در این صورت اگر مبدأ، به بسته‌ی ارسالی از مقصد با ارسال یک بسته‌ی ACK جواب دهد، ارتباط بین مبدا و مقصد برقرار می‌شود. به این صورت که ارتباط نیمه‌باز ایجاد شده خاتمه یافته و منابع اختصاص داده شده به آن آزاد می‌شود. اما اگر آدرس IP مبدأ جعلی باشد بسته‌ی فرستاده شده بی‌جواب مانده و ارتباط نیمه‌باز ایجاد شده تا زمانی که مقصد به آن خاتمه دهد، باقی می‌ماند. این درخواست‌ها تا جایی ادامه پیدا می‌کنند که دستگاه سرویس‌دهنده را از کار بیندازد. دراین صورت سیستم سرویس‌دهنده دیگر توانایی پاسخ‌گویی به کاربران قانونی خود را نخواهد داشت.

پژوهش‌های زیادی در این زمینه انجام شده است. Nski و همکارانش جهت دفاع در برابر حملات SYN، طرح نمونه‌برداری دقیقی را پیشنهاد کردند. این طرح جهت تایید اعتبار اتصالات قانونی به بررسی بخش‌های TCP برای پیدا کردن حداقل یکی از بخش‌های چندگانه ACK که به سرور وارد می‌شوند، می‌پردازد [۱]. Haris و همکارانش در [۲] جهت تشخیص حملات SYN-flooding از کشف ناهنجاری استفاده کرده‌اند. در [۳] برای تشخیص حملات SYN-flooding، یک سیستم ایمنی مصنوعی بر اساس الگوریتم DCA طراحی و پیاده‌سازی شده است. در [۴] یک مکانیسم دفاعی ارائه شده است که از طریق روتر لبه با شناسایی آدرس IP جعلی شبکه، SYN-ACK ورودی را تعیین اعتبار می‌کند. در [۵] با استفاده از اطلاعات تأخیر توزیع ترافیک شبکه، به طور مستقل حملات در طرف قربانی تشخیص داده می‌شوند. در [۶] مکانیزم مبتنی بر MMDBMS وزن سبک برای تشخیص و پیشگیری از حملات SYN-flooding پیشنهاد شده است. در پژوهشی دیگر با بررسی آنتروپی بسته SYN بین زمان ورود، یک معیار اندازه گیری تصادفی پیشنهاد شده است [۷]. در [۸] روش لیست سفید جهت مقابله با حملات ارائه شده است که در آن سعی بر این است که احتمال موفقیت ارتباطات قانونی افزایش یابد. در [۹] جهت محافظت از سرور تحت حملات DoS، مکانیزم کنترل پارامترهای مورد استفاده (UPC) در مد انتقال ناهمگام (ATM) شبکه‌ها مورد استفاده قرار گرفته است. Xiaofeng و همکارانش با تمرکز بر روی ویژگی‌های سرویس حفاظت به چگونگی دفع حمله و ردیابی منبع حملات پرداخته‌اند [۱۰]. در [۱۱] از یک مدل صف‌بندی برای ارزیابی حمله‌های DoS در شبکه‌های کامپیوتری استفاده شده است. در [۱۲] حمله‌های DoS از نوع flooding با تمرکز روی منبع cpu بررسی شده است و برای کشف حمله‌های DoS از سه پارامتر نرخ ورود درخواست‌ها، نرخ رشد صف و زمان پاسخگویی استفاده شده است.

در ادامه‌ی این مقاله در بخش دوم الگوریتم بهینه‌سازی اجتماع ذرات تشریح شده است. فیلتر موثر انطباقی در بخش سوم مورد بررسی قرار می‌گیرد. در بخش چهارم الگوریتم پیشنهادی، نحوه‌ی نگاشت الگوریتم PSO، فیلتر موثر انطباقی و حمله‌ی SYN-flooding ارائه می‌گردد. در بخش پنجم روش پیشنهادی در محیط MATLAB شبیه‌سازی شده و نتایج بدست‌آمده، مورد ارزیابی قرار می‌گیرد. در نهایت نتیجه‌گیری در بخش ششم این مقاله آمده است.

۲- الگوریتم بهینه‌سازی اجتماع ذرات

الگوریتم بهینه‌سازی اجتماع ذرات (PSO) جهت هدایت به منطقه‌ی موردنظر در فضای جستجو از رفتار اجتماعی دسته پرندگان یا گروه ماهی‌ها در حین جستجوی غذا استفاده می‌کند [۱۳،۱۴]. در این الگوریتم هر جواب مساله نشان‌دهنده‌ی موقعیت یک پرنده در فضای جستجو است که آن را ذره می‌نامند. تابع برازندگی که هدف بهینه‌سازی می‌باشد، مقدار شایستگی تمام ذره‌ها را تعیین می‌کند. مسیر حرکت هر ذره که دارای مولفه‌ای بنام سرعت می‌باشد، در فضای جستجو توسط این الگوریتم تعیین می‌کند. جمعیت PSO شامل تمامی ذره‌ها است که Swarm نامیده می‌شود. الگوریتم PSO شامل دو مدل معادله سرعت و مکان می‌باشد و مختصات هر ذره نشان‌دهنده‌ی یک جواب ممکن مرتبط با دو بردار است. بردارهای موقعیت (Xi) و سرعت (Vi) دو بردار وابسته و مرتبط با هر ذره‌ی i در فضای جستجوی N بعدی هستند که به ترتیب با استفاده از معادلات زیر بیان می‌شوند:

$$x_i^{k+1} = x_i^k + cv_i^{k+1} \quad (۱)$$

$$V_i = [v_{i1}, v_{i2}, \ldots, v_{iN}] \quad (۲)$$

در واقع پاسخ‌های ممکن، ذرات مربوط به مجموعه‌ای از پرندگان است که در یک فضای پاسخ ممکن برای جستجوی جواب‌های بهینه پیش می‌روند. موقعیت هر ذره باتوجه به فاکتورهایی از جمله بهترین جستجوی ذره، بهترین تجربه کلی پرواز گروهی و بردار سرعت پیشین خود ذره، بر اساس روابط زیر به‌روز می‌شود.

$$x_i^{k+1} = x_i^k + Cv_i^{k+1} \quad (۳)$$

$$v_i^{k+1} = wv_i^k + c_1r_1(P\,best_i^k - x_i^k) + c_2r_2(G\,best^k - x_i^k) \quad (۴)$$

که در آن c_1 و c_2 دو ثابت عددی مثبت، r_1 و r_2 دو عدد تصادفی با توزیع یکنواخت در محدوده‌ی [۰،۱] و w وزن لختی می‌باشد که به

صورت زیر تعیین می‌شود.

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} \times iter \quad (۵)$$

که $iter_{max}$، تعداد ماکزیمم تکرار و iter، تعداد تکرار جاری می‌باشد. $Pbest_i^k$ بهترین موقعیت ذره i است که بر اساس تجربه ذره به دست آمده و به صورت زیر قابل بیان است:

$$P\,best_i^k = [x_{i1}^{p\,best}, x_{i2}^{p\,best}, ..., x_{iN}^{p\,best}] \quad (۶)$$

$Gbest^k$ بهترین موقعیت ذره بر اساس تجربه کلی گروهی می‌باشد که در آن k شاخص تکرار است و به صورت زیر تعیین می‌گردد.

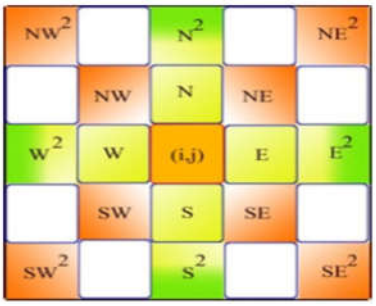
$$G\,best = [x_1^{g\,best}, x_2^{g\,best}, ..., x_N^{g\,best}] \quad (۷)$$

۳- فیلتر موثر انطباقی

بهبود کیفیت تصاویر آسیب دیده با استفاده از نویز ضربه‌ای یکی از مسائل مهم در پردازش تصویر می‌باشد که با روش‌های مختلفی از جمله فیلترهای مبتنی بر میانه صورت می‌گیرد [۱۵]. فیلتر موثر انطباقی برای اصلاح تصاویر نویزی با سطوح خاکستری با شدت [۰،۲۵۵] طراحی شده است [۱۶]. برای حذف نویز ضربه‌ای در این فیلتر، ابتدا یک پنجره دو بعدی ۵×۵ در مرکز هر پیکسل ایجاد می‌شود. بر اساس شکل (۱) برای همسایه‌های هر پیکسل با استفاده از رابطه (۶) و (۷) وزن مخصوصی اختصاص داده می‌شود.

$$w_{m.n}^{1} img\,(i, j) = \begin{cases} 4 & if\ \ img(i, j)\ \ is\ N, S, W, E \\ 3 & if\ \ img\,(i, j)\ \ is\ NW, SW, NE, SE \end{cases} \quad (۸)$$

$$w_{m.n}\,2\,img\,(i, j) = \begin{cases} 2 & if\ \ img\,(i, j)\ \ is\ N, S, W, E \\ 1 & if\ \ img\,(i, j)\ \ is\ NW^2, SW^2, NE^2\ SE^2 \end{cases} \quad (۹)$$



شکل (۱): همسایه‌های هر پیکسل [۴]

Fig. (1): Neighbors per pixel

متغییرهای $Count_1$ و $Count_2$ برای نگهداری تعداد پیکسل‌های نویزی در همسایه‌های پیکسل جاری در نظر گرفته می‌شوند به طوری که اگر هر یک از همسایگان {NW, N, NE, W, E, SW S, SE} مقدار نویزی داشته باشند، آنگاه یک واحد به Count1 اضافه می‌شود.

همچنین اگر مقدار پیکسل هر یک از همسایه‌های $\{N^2, S^2, E^2, W^2, NE^2, SW^2, NW^2 and\ SE^2\}$ نویزی باشند، آنگاه $Count_2$ یک واحد افزایش می‌یابد. T_1 و T_2 دو پارامتر ثابت هستند که توسط کاربر تعیین می‌شود.

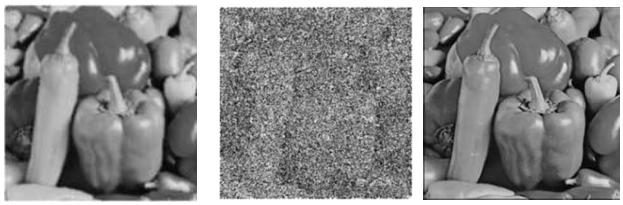
مقدار یک پیکسل نویزی است اگر:

- $count_1 < T_1$ و $count_2 < T_2$
- $count_1 > T_1$ و $count_2 < T_2$
- $count_1 < T_1$ و $count_2 > T_2$

در صورتی که پیکسل جاری نویزی تشخیص داده شد:

- اگر در پنجره ۵×۵ تعداد پیکسل‌های بدون نویز بیشتر از ۳ بود آنگاه میانه مقادیر بدون نویز پنجره برای پیکسل جاری در نظر گرفته می‌شود.
- در غیر این صورت میانگین مقادیر پنجره جایگزین پیکسل جاری می‌گردد.

در شکل (۲) نتایج شبیه‌سازی فیلتر موثر انطباقی نشان داده شده است.



الف. تصویر اصلی ب. تصویر نویزی ج. تصویر بهبود یافته

شکل (۲): نتایج شبیه‌سازی فیلتر موثر انطباقی برای تصویر نویزی با چگالی نویز ضربه‌ای ۸۰٪

Fig. (2): Results of Adaptive Effective Filter for Noise Effect with Noise Efficacy 80%

۴- روش پیشنهادی در این مقاله

جهت برقرای ارتباط بین مبدأ و مقصد، در پروتکل TCP از دو پارامتر اصلی h و m استفاده می‌شود و می‌بایست مدت زمان نگه‌داری ارتباطات نیمه‌باز (m) و تعداد این ارتباطات (h) کنترل گردد. بنابراین رفتار پروتکل TCP تحت‌تأثیر دو پارامتر h و m هستند. به همین دلیل h و m به عنوان ذرات اجتماع در نظر گرفته می‌شوند. مقادیر این پارامترها با استفاده از روش بهینه‌سازی اجتماع ذرات به صورت پویا تغییر کرده و با حرکت به سمت نقطه‌ی بهینه، باعث حذف زودهنگام ارتباطات نیمه‌بازی که برای درخواست‌های حمله اختصاص داده شده است، می‌شوند. به طور کلی باعث بهبود عملکرد TCP در برابر حملات SYN-flooding می‌شوند و باعث افزایش تعداد کل ارتباطات سیستم می‌شود.

در الگوریتم پیشنهادی مقادیر h و m با استفاده از الگوریتم PSO در هر مرحله محاسبه و به‌روز رسانی می‌شود. سپس فیلتر موثر انطباقی جهت انتخاب یک مقدار از بین مجموعه مقادیری که برای h و m از طریق الگوریتم PSO حاصل شده، به کار گرفته می‌شود.

۴-۱- بکارگیری الگوریتم PSO

در الگوریتم PSO دو مدل معادله‌ی سرعت و مکان مورد استفاده قرار می‌گیرد. مختصات هر ذره نشان‌دهنده‌ی یک جواب ممکن مرتبط با دو بردار موقعیت (xi) و سرعت (vi) است که مربوط به ذره‌ی i در فضای جستجوی N بعدی می‌باشند و به صورت زیر تعریف می‌شوند:

$x_i = [x_{i1}, x_{i2}, \ldots, x_{iN}]$ و $v_i = [v_{i1}, v_{i2}, \ldots, v_{iN}]$

$$v_h^{k+1} = wv_h^k + c_1 r_1 (Pbest_h^k - h^k) + c_2 r_2 (Gbest^k - h^k) \quad (۱۰)$$

$$v_m^{k+1} = wv_m^k + c_1r_1(Pbest_m^k - m^k) + c_2r_2(Gbest^k - m^k) \quad (۱۱)$$

که در آن c_1 و c_2 دو ثابت عددی مثبت، r_1 و r_2 دو عدد تصادفی می‌باشند. $Pbest_i^k$ بهترین موقعیت ذره‌ی i است که براساس تجربه ذره به دست‌آمده و طبق معادله‌ی زیر بیان می‌شود.

$$Pbest_i^k = [x_{i1}^{pbest}, x_{i2}^{pbest}, ..., x_{iN}^{pbest}] \quad (۱۲)$$

$Gbest^k$ بهترین موقعیت ذره براساس تجربه‌ی کلی گروهی می‌باشد و به صورت زیر تعریف می‌شود.

$$Gbest = [x_1^{gbest}, x_2^{gbest}, ..., x_N^{gbest}] \quad (۱۳)$$

فاکتور سرعت باید مقادیر h و m را که در الگوریتم پیشنهادی ذره‌ها هستند به مقادیر Gbest نزدیکتر کند. در این الگوریتم Pbest زمانی حاصل می‌شود که مقادیر h و m به گونه‌ای باشند که معیارهای مورد نظر به مقدار بهینه‌ی خود برسند. در این صورت مقادیر به‌روز شده‌ی h و m برای مقدار Pbest در نظر گرفته می‌شود. بهترین پاسخ، زمانی به دست خواهد آمد که مقادیر h و m با مقادیر Gbest برابر باشند. در نهایت با استفاده از مقادیر به دست آمده برای Pbest و Gbest و جایگذاری در معادلات ۱۰ و ۱۱ به ترتیب سرعت و موقعیت ذرات تعیین خواهد شد. موقعیت بعدی هر ذره، با استفاده از روابط زیر تعیین می‌شود:

$$h^{k+1} = h^k + v_h^{k=1} \quad (۱۴)$$

$$m^{k+1} = m^k + v_m^{k=1} \quad (۱۵)$$

h^{i+1} و m^{i+1} بدست‌آمده از روابط (۱۴) و (۱۵)، ذره‌های جدید هستند و به این ترتیب مقادیر h و m به‌روز می‌شوند. به‌این طریق مقادیر h و m استاتیک در پروتکل TCP در الگوریتم PSO به متغیرهای پویایی تبدیل می‌شوند که با توجه به شرایط شبکه تغییر می‌کنند.

۴-۲- بکارگیری فیلتر موثر انطباقی

مقادیر m و h بهینه در الگوریتم PSO بر اساس تجربه و با گذر زمان قابل تعیین می‌باشند. بنابراین در صورتی که تعداد درخواست‌های حمله زیاد شوند، سرعت الگوریتم PSO در تشخیص m و h بهینه پایین آمده و یا با خطا مواجه می‌شود. با توجه به اینکه زمان نقش بسیار مهمی در کشف و دفاع در برابر حملات سیل‌آسای SYN ایفا می‌کند، برای تسریع در رسیدن به مقادیر بهینه m و h، پس از تعیین مقادیر m و h بهینه توسط الگوریتم PSO، فیلتر موثر انطباقی مقادیر m و h را فیلتر کرده تا نتایج بهبود یابد. شکل (۳) نمایانگر برازندگی و تنظیم مقادیر h, m براساس الگوریتم PSO و فیلتر موثر انطباقی است.

شکل (۳): استفاده از PSO و فیلتر موثر انطباقی جهت برازندگی و تنظیم مقادیر h و m

Fig. (3): Using PSO and effective adaptive filter for fitness and adjusting the values of h and m

ضمن اینکه فرض‌های زیر در نظر گرفته می‌شوند:

- پنجره: مقادیر همسایه‌های یک گره
- اندازه پنجره: به عنوان پارامتر ورودی تعیین می‌شود.
- مقدار معمول: مقداری که طبق فیلتر موثر انطباقی سالم تشخیص داده شود.
- مقدار نویزی: مقداری که با استفاده از فیلتر موثر انطباقی نویزی شناسایی گردد.

نحوه‌ی عملکرد فیلتر موثر انطباقی به این صورت است که ابتدا پنجره‌ای با مقادیر همسایه‌های یک گره در اندازه مشخص شده با پارامتر ورودی شکل می‌گیرد. اگر گره دارای مقدار بالاترین شدت یا پایین‌ترین شدت بود از روش زیر برای فیلتر استفاده می‌شود.

گام اول: بر اساس موقعیت ذرات در الگوریتم PSO، یک پنجره در اندازه ۵×۵ در محدوده گره جاری همراه با مقادیر همسایه‌های گره ایجاد کن.

گام دوم: طبق روش فیلتر موثر انطباقی مقادیر نویزی در پنجره ایجاد شده را شناسایی کن.

گام سوم: برای مقادیر نویزی:

- اگر در پنجره ۵×۵ تعداد پیکسل‌های بدون نویز بیشتر از ۳ بود آنگاه میانه مقادیر بدون نویز پنجره برای گره جاری در نظر گرفته می‌شود.
- در غیر اینصورت میانگین مقادیر پنجره برای گره جاری در نظر گرفته می‌شود.

برای فیلترکردن m نیز از روشی مشابه روش تعریف شده‌ی فوق استفاده می‌شود. پروتکل TCPدرخواست‌هایی که با بسته‌ی SYN وارد سرور می‌شوند را در بافر پشتیبان قرار داده و منابع لازم برای برقراری یک ارتباط کامل را از بافر پشتیبان به آنها تخصیص می‌دهد و ارتباط نیمه باز را برقرار می‌کند. اما تعداد ارتباطات نیمه‌بازی که سرور می‌تواند ایجاد نماید، محدود بوده و مدت زمان نگه‌داری این ارتباطات نیمه‌باز نیز زمان ثابتی می‌باشد. هدف از ارائه‌ی الگوریتم پیشنهادی این است که این دو پارامتر به صورت پویا و با توجه به وضعیت شبکه تغییر کنند. برای این منظور پارامترهای زیر تعریف می‌شوند.

درصد تملک بافر توسط درخواست‌های عادی (P_r): وقتی یک ارتباط نیمه‌باز ایجاد شد، در بافر قرار می‌گیرد. مجموع مدت زمان ارتباطات نیمه‌باز ایجادشده توسط درخواست‌های عادی نسبت به مجموع مدت زمان همه‌ی ارتباطات نیمه‌باز را درصد تملک بافر توسط درخواست‌های عادی تعریف می‌کنیم.

درصد تملک بافر توسط درخواست‌های حمله (P_a): به طور مشابه، مجموع مدت زمان ارتباطات نیمه‌باز ایجادشده توسط درخواست‌های حمله نسبت به مجموع مدت زمان همه‌ی ارتباطات نیمه‌باز به عنوان درصد تملک بافر توسط درخواست‌های حمله تعریف می‌شوند.

توانایی یک سرور در ارائه‌ی خدمات وابسته به تعداد سرویس‌های مطلوبی است که سرور به درخواست‌های عادی ارائه می‌کند. بنابراین مقدار تملک بافر توسط درخواست‌های عادی، باید به اندازه‌ی کافی بزرگ باشد و به دلیل مشابه بایستی مدت تملک بافر توسط درخواست‌های حمله، به

اندازه‌ی کافی کوچک باشد. از این اطلاعات استفاده کرده و تابع هدف برای الگوریتم PSO فیلتر موثر انطباقی به صورت زیر تعریف می‌شود.

Objective Function: Maximize P_r / P_a (۱۶)

به طور کلی هرچه مقدار این تابع بیشتر شود، توانایی سرور در ارائه‌ی خدمات مطلوب بیشتر خواهد بود. بنابراین الگوریتم پیشنهادی، سعی در ماکزیمم کردن این تابع دارد.

۵- نتایج شبیه‌سازی

در این مقاله از توزیع‌های آماری برای ورود درخواست‌های عادی و حمله جهت شبیه‌سازی حمله‌های SYN flooding، استفاده شده است. توزیع پواسون با آهنگ میانگین λ جهت نرخ ورود درخواست‌های عادی به سرور و توزیع پواسون با آهنگ میانگین $k\lambda$ جهت نرخ ورود درخواست‌های حمله در نظر گرفته می‌شود که k شدت ورود درخواست‌های حمله نسبت به درخواست‌های عادی می‌باشد. همچنین توزیع نمایی با میانگین μ جهت مدت زمان نگه‌داری ارتباطات نیمه‌باز برای درخواست‌های عادی در نظر گرفته می‌شود. ارتباطات نیمه‌باز برای درخواست‌های حمله، به مدت زمان h نگه داشته می‌شوند. جهت پیاده سازی الگوریتم پیشنهادی مقادیر m و h به صورت پیش فرض و ثابت در نظر گرفته می‌شوند و فرض می‌شود هیچ مکانیزم دفاعی وجود ندارد. مبنای مقادیر پیش فرض m و h مقادیر Linux2.4 در نظر گرفته می‌شوند که به ترتیب عبارتند از ۷۵ و ۱۲۸. در هر دو روش، λ=10/s و μ =100/s در نظر گرفته شده‌اند و همچنین جهت آزمایش سرور موردنظر هنگامی که تحت شدت حملات مختلف قرار می‌گیرد مقدار k با توزیع یکنواخت بین بازه ی ۰ تا ۲ در نظر گرفته شده است.

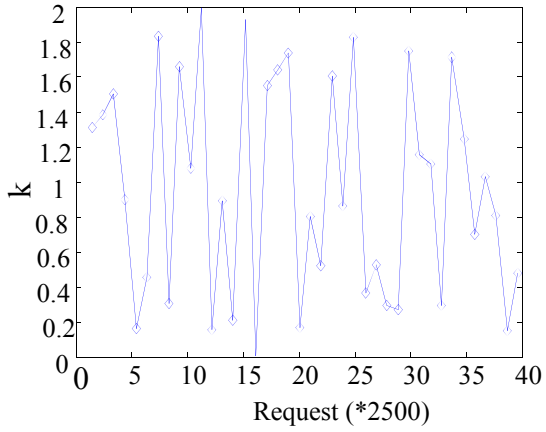
این مسئله در محیط MATLAB پیاده‌سازی شده و با توجه به سربار پایین الگوریتم بهینه‌سازی PSO، زمان صرف شده برای اجرای الگوریتم ناچیز است. ضمن اینکه اجرای الگوریتم نیاز به سخت‌افزار خاصی نداشته و الگوریتم روی سیستم‌های عادی قابل اجرا است.

تغییرات مقدار k در طول شبیه‌سازی، در شکل (۴) نشان داده شده است. مدت زمان تملک بافر توسط درخواست‌های عادی بر اساس تغییرات k، در جدول (۱) و شکل (۵) ارائه شده است. همانطوری که مشاهده می شود، روش پیشنهادی نسبت به الگوریتم LA-SFDD [۱۷] و حالتی که از مقادیر پیش فرض استفاده شده است، دارای مقادیر مطلوبتری از نظر سرویس‌دهی به کاربران عادی را دارا می‌باشد. همچنین جدول (۲) و شکل (۶) که مدت زمان تملک بافر توسط درخواست‌های حمله را نشان می‌دهد، بیانگر کاهش سرویس‌دهی به درخواست‌های حمله می‌باشد. این بهبودها، حاصل تغییرات پویای مقادیر h و m می‌باشد.

Table (1): The duration of the buffer ownership by ordinary requests

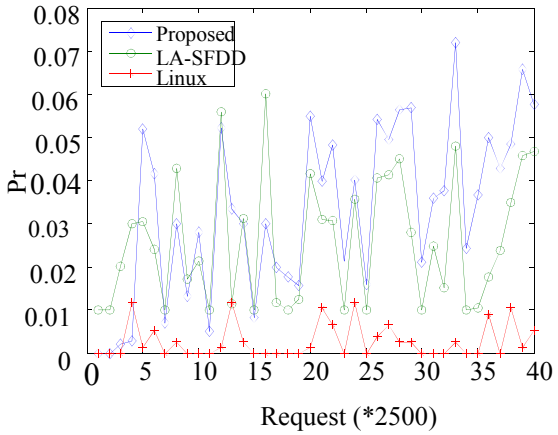
جدول (۱): مدت زمان تملک بافر توسط درخواست‌های عادی

Methods	Time (Second * 50)							
	۵	۱۰	۱۵	۲۰	۲۵	۳۰	۳۵	۴۰
Linux	۰/۰۰۲	۰/۰۰۱	۰/۰۰۱	۰/۰۰۲	۰/۰۰۱	۰/۰۰۱	۰/۰۰۱	۰/۰۰۶
LA-SFDD	۰/۰۳۱	۰/۰۲۱	۰/۰۱۰	۰/۰۴۲	۰/۰۱۰	۰/۰۱۰	۰/۰۱۱	۰/۰۴۷
Proposed	۰/۰۵۲	۰/۰۲۸	۰/۰۰۸	۰/۰۵۵	۰/۰۱۶	۰/۰۲۱	۰/۰۳۷	۰/۰۵۸

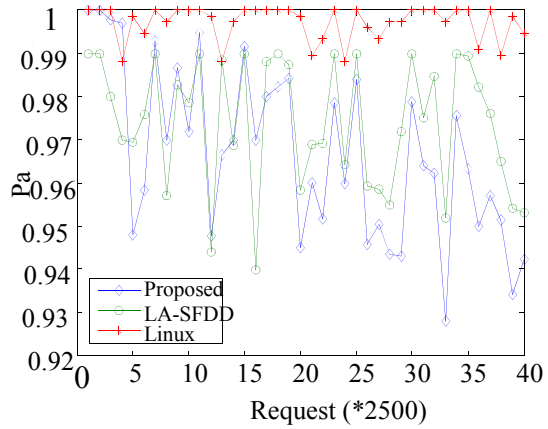


شکل (۴): مقدار k با شدت حمله مختلف

Fig. (4): The value of k with different attack severity



شکل (۵): مدت زمان تملک بافر توسط درخواست‌های عادی
Fig. (5): The duration of the buffer ownership by ordinary requests



شکل (۶): مدت زمان تملک بافر توسط درخواست‌های حمله
Fig. (6): The duration of the buffer ownership by attack requests

Table (2): Duration of buffer ownership by attack requests
جدول (۲): مدت زمان تملک بافر توسط درخواست‌های حمله

Methods	Time (Second * 50)							
	۵	۱۰	۱۵	۲۰	۲۵	۳۰	۳۵	۴۰
Linux	۰ / ۹۸۸	۰ / ۹۹۹	۰ / ۹۹۷	۰ / ۹۹۹	۰ / ۹۸۸	/ ۹۹۶	۰ / ۹۹۹	۰ / ۹۹۹
LA-SFDD	۰ / ۹۶۹	۰ / ۹۷۹	۰ / ۹۹۰	۰ / ۹۵۸	۰ / ۹۹۰	۰ / ۹۹۰	/ ۹۸۹	۰ / ۹۵۳
Proposed	۰ / ۹۹۸	۰ / ۹۹۹	۰ / ۹۹۹	۰ / ۹۹۸	۰ / ۹۹۹	۰ / ۹۹۹	۰ / ۹۹۹	۰ / ۹۹۴

۶- نتیجه‌گیری

این مقاله، روشی را برای مقابله با حملات SYN flooding ارائه می‌کند. برای این کار، از الگوریتم بهینه‌سازی اجتماع ذرات و فیلتر موثر انطباقی برای بروز کردن دو پارامتر مهم و تاثیرگذار در حملاتSYN flooding استفاده شده است. نتایج شبیه‌سازی نشان می‌دهد که روش پیشنهادی، با توجه به عملکرد آگاهانه و هوشمند خود و تصمیم‌گیری‌های به موقع جهت تنظیم پارامترهای تاثیرگذار در عملکرد سیستم، از نظر افزایش احتمال برقراری ارتباط عادی و همچنین کاهش احتمال موفقیت حمله، بهبود قابل‌توجهی نسبت به الگوریتم‌های مشابه خواهد داشت.

References

[1] M. Korczynski, L. Janowski, A. Duda, "An accurate sampling scheme for detecting SYN flooding attacks", Proceeding of the IEEE/ICC, pp. 1-5, Kyoto, Japan, June 2011 (doi:10.1109/icc.2011.5962593).

[2] S.H.C. Haris, R.B. Ahmad, M.A.H.A. Ghani, "Detecting TCP SYN flood attack based on anomaly detection", Proceeding of the IEEE/Netapps, pp. 240-244, Kedah, Malaysia, Sep. 2010 (doi:10.1109/NETAPPS.2010.50).

[3] N. B.I. Al-Dabagh, I.A. Ali, "Design and implementation of artificial immune system for detecting flooding attacks", Proceeding of the IEEE/HPCSim, pp. 381-390, Istanbul, Turkey, July 2011 (doi:10.1109/HPCSim.2011.5999850).

[4] H. Safaa, "A collaborative defense mechanism against SYN flooding attacks in IP networks", Journal of Network and Computer Applications, Vol. 31, pp. 509–534, Nov. 2008 (doi:10.1016/j.jnca.2007.12.004).

[5] B. Xiao, W. Chen, Y. He, "An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victim side independently", Journal of Parallel and Distributed Computing archive, Vol. 68 No. 4, pp. 456-470, April 2008 (doi:10.1016/j.jpdc.2007.06.013).

[6] S. G. Bhirud, V. Katkar, "SYN flood attack prevention using main-memory database management system", Proceeding of the IEEE/AH-ICI, pp. 1-6, Kathmandu, Nepal, Nov. 2011 (doi: 10.1109/ AHICI.2011.6113945).

[7] L. Arshadi, A. Jahangir, "Entropy based SYN flooding detection", Proceeding of the IEEE/LCN, pp. 139-142, Bonn, Germany, Oct. 2011 (doi:10.1109/LCN.2011.6115171).

[8] T. Kim, Y. Choi, J. Kim, S. Je Hong, "Annulling SYN flooding attacks with whitelist", Proceeding of the IEEE/WAINA, Okinawa, Japan, March 2008 (doi:10.1109/WAINA.2008.218).

[9] Y.W. Chen, "Study on the prevention of SYN flooding by using traffic policing", Proceeding of the IEEE/NOMS, April 2000 (doi:10.1109/NOMS.2000.830416).

[10] Q. Xiaofeng, H. Jihong, C. Ming, "A mechanism to defend SYN flooding attack based on network measurement system", Proceeding of the IEEE/ITRE, June/July 2004 (doi:10.1109/ITRE.2004.1393677).

[11] Y. Wang , C. Lin , Q. Li, Y. Fang ,"A queueing analysis for the denial of service (DoS) attacks in computer network", Computer Networks, Vol. 51, pp. 3564–3573, Aug. 2007 (doi:10.1016/j.comnet.2007.02.011).

[12] S. Khan, I. Traore, "Queue-based analysis of DoS attacks", Proceeding of the IEEE/IAW, pp. 266–273, West Point, NY, June 2005 (doi: 10.1109/IAW.2005.1495962).

[13] F. Khajeh-khalili, M.A. Honarvar, "Design and simulation of a wilkinson power divider with high isolation for tri-band operation using PSO algorithm", Journal of Intelligent Procedures in Electrical Technology, Vol. 6, No. 23, pp. 13-20, Autumn 2015 (Text in Persian).

[14] H. Li, D. Yang, W. Su, J. Lü, X. Yu, "An overall distribution particle swarm optimization MPPT algorithm for photovoltaic system under partial shading", IEEE Trans.on Industrial Electronics, Vol. 66, No. 1, pp. 265-275, Jan. 2019 (doi:10.1109/TIE.2018.2829668).

[15] R. C. Gonzalez, R. E. Wood, Digital Image Processing, Prentice Hall, 2002.

[16] M. M. Javidi, R. Hoseinpour-Fard, S. Khatami, M. Jampour, "An effective adaptive technique for impulse noise detection and reduction in digital images", Proceeding of the IEEE/HIS, pp. 217-229, Melacca, Malaysia , Dec. 2011 (doi:10.1109/HIS.2011.6122108).

[17] M. Bekravi, S. Jamali, G. Shaker, "Defense against SYN-flood denial of service attacks based on learning automata", International Journal of Computer Science, vol. 9, No. 3, pp. 514-520, 2012.