جریان‌کاوی بلادرنگ هشدارهای نفوذی برای تشخیص حملات چندمرحله‌ای

علی احمدیان رمکی[۱]، رضا ابراهیمی آتانی[۲]، الناز حاجی علیلو[۳]

[۱]دانش آموخته کارشناسی ارشد مهندسی کامپیوتر – نرم افزار، دانشگاه گیلان، رشت
ahmadianrali@msc.guilan.ac.ir
[۲]استادیار و عضو هیأت علمی گروه مهندسی کامپیوتر، دانشگاه گیلان، رشت
rebrahimi@guilan.ac.ir
[۳]دانشجوی کارشناسی ارشد مهندسی فناوری اطلاعات – شبکه های کامپیوتری، پردیس دانشگاه گیلان، رشت
elnaz.hajialilo@gmail.com

چکیده

سیستم‌های تشخیص نفوذ به‌عنوان یکی از تجهیزات امنیتی سیستم‌ها و شبکه‌های کامپیوتری، وظیفه تشخیص رخدادهای امنیتی و تولید هشدارهای لازم به هنگام تخطی از سیاست‌های امنیتی تعریف شده را بر عهده دارند. این هشدارها جهت تحلیل‌های بعدی برای مدیر امنیتی ارسال می‌شوند. مشکل اصلی سیستم‌های تشخیص نفوذ تولید زیاد هشدارها، نادرست بودن خیل عظیمی از هشدارهای تولیدی و نیز ضعف در برابر تشخیص سناریوی حملات چندمرحله‌ای است که در آن مهاجم با بهره‌گیری از مجموعه‌ای از حملات منفرد، عمل خصمانه خود را انجام می‌دهد. یکی از چالش‌های مهم در این زمینه تجمیع بلادرنگ هشدارها در جهت کشف سناریوی حملات چندمرحله‌ای است. همبسته‌سازی هشدارها فرآیندی است که در آن هشدارهای تولید شده توسط حس‌گرهای تشخیص نفوذ موجود در یک محیط تحت نظارت، تحلیل شده تا در نهایت دید کاملی از تلاش‌های نفوذی احتمالی مهاجم به‌دست آید. در این مقاله، روشی کارا و مبتنی بر ترکیبی از تکنیک‌های آماری و جریان‌کاوی برای همبسته‌سازی هشدارها جهت تشخیص بلادرنگ سناریوی حملات چندمرحله‌ای پیشنهاد می‌گردد. نتایج ارزیابی روش پیشنهادی بر روی مجموعه داده‌های آزمایشگاهی معتبر موجود در این زمینه، حاکی از تشخیص بلادرنگ و دقیق سناریوی حملات است که در مقایسه با پژوهش‌های معتبر پیشین از سرعت بالاتری در کشف حملات برخوردار می‌باشد.

کلمات کلیدی

امنیت شبکه، سیستم تشخیص نفوذ، همبسته‌سازی هشدار، حمله چندمرحله‌ای، درخت حمله

۱- مقدمه

امروزه سیستم‌های تشخیص نفوذ[۱] (IDS)، به‌طور قابل ملاحظه‌ای برای افزایش امنیت در سیستم‌های کامپیوتری مورد استفاده قرار می‌گیرند. تشخیص نفوذ، فرآیند نظارت بر وقایع رخداده در یک شبکه و یا سیستم کامپیوتری در جهت کشف موارد انحراف از سیاست‌های امنیتی است. این سیستم‌ها در مواجهه با نقض سیاست‌های امنیتی، هشداری را جهت اعلان وضعیت جاری امنیتی، برای مدیران سطح بالا تولید می‌نمایند. به منظور مقابله با نفوذگران به سیستم‌ها و شبکه‌های کامپیوتری، روش‌های تشخیص نفوذ زیادی معرفی شده‌اند[۱]. اهمیت سیستم‌های تشخیص نفوذ در برقراری امنیت سیستم‌ها و شبکه‌های کامپیوتری، انکارناپذیر است [۲]. با این حال، استفاده از سیستم‌های تشخیص نفوذ مشکلات مربوط به خود را داراست [۳]. از جمله مهم‌ترین این مشکلات می‌توان به تولید حجم زیادی از هشدارهای خام نفوذی، نرخ بالای هشدارهای مثبت–غلط[۲] و ناکارآمدی در تشخیص سناریوی حملات چندمرحله‌ای اشاره کرد.

هشدارهایی که حس‌گرهای تشخیص نفوذ تولید می‌کنند، هشدارهای سطح پایینی هستند که چنانچه به‌صورت منفرد در نظر گرفته شوند، تهدیدات واقعی سیستم را به‌درستی نشان نمی‌دهند. معمولاً مهاجمین برای نیل به اهداف خود از حملات چندمرحله‌ای استفاده می‌کنند [۳]. به‌این صورت که، در ابتدا از یک یا چند آسیب‌پذیری در سیستم استفاده کرده و حمله خود را یک گام جلو می‌برند و سپس با بهره‌گیری از پی‌آمدهای[۳] این گام که پیش‌نیاز گام بعدی حمله است، گام بعدی را اجرا می‌کنند و به این ترتیب حمله خود را گام‌به‌گام جلو می‌برند. بنابراین، با توجه به عدم امکان کشف این ارتباطات توسط سیستم‌های تشخیص نفوذ، نیازمند ایجاد یک دید سطح بالاتر از وضعیت امنیتی سیستم هستیم. همبسته‌سازی هشدارها چنین دیدی را برای مدیر سیستم یا شبکه کامپیوتری فراهم می‌آورد.

همبسته‌سازی هشدارها فرآیندی است که طی آن هشدارهای تولید شده توسط یک یا چند حس‌گر موجود در شبکه، تحلیل شده تا یک دیدگاه مختصر و سطح بالایی از تلاش‌های نفوذ احتمالی فراهم گردد [۴]. همبسته‌ساز با بررسی هشدارهای تولید شده و کشف ارتباطات منطقی آن‌ها به‌جای تولید صدها هشدار سطح پایین و گسسته، یک هشدار سطح بالا را به مدیر سیستم ارائه می‌کند [۵]. بنابراین تعیین رخدادهای بسیار مهم از میان حجم زیادی از وقایع ثبت شده و تشخیص سناریوی حملات چندمرحله‌ای، هدف نهایی فرآیند همبسته‌سازی است تا بر کیفیت هشدارها افزوده و از کمیت آن‌ها کاسته شود. تاکنون روش‌های زیادی برای همبسته‌سازی هشدارها معرفی شده‌اند [۴]. مشکل اصلی بیشتر روش‌های همبسته‌سازی، عدم کارایی آن‌ها برای کاربردهای بلادرنگ است که علاوه بر کارایی روش، سرعت پردازش هشدارها نیز بسیار مهم است [۶].

در پژوهش جاری، یک چارچوب همبسته‌سازی هشدار پیشنهادی برای کاربردهای بلادرنگ ارائه شده است. این چارچوب بر اساس تصفیه توالی رویدادهای بحرانی[۴] که می‌توانند بخشی از سناریوی حملات چندمرحله‌ای باشند، عمل می‌نماید. چارچوب پیشنهادی با استفاده از ترکیبی از روش‌های آماری و جریان‌کاوی در دو مود عملیاتی برخط[۵] و برون‌خط[۶] کار می‌کند. در مود برون‌خط با استفاده از ساخت یک درخت توالی رویداد به‌کمک ماتریس همبسته‌سازی هشدار، رفتارهای مهاجم را یاد می‌گیرد. در فاز برخط نیز با استفاده از هشدارهای دریافتی، سناریوی حملات را کشف می‌نماید. در ادامه با ارزیابی الگوریتم پیشنهادی بر روی مجموعه داده معتبر DARPA 2000 [۷] با هدف تشخیص سناریوی حملات چندمرحله‌ای, قادر به دستیابی به این اهداف با دقت قابل‌قبولی بوده‌ایم.

ادامه این مقاله به‌صورت زیر سازمان‌دهی شده است. ابتدا در بخش دوم برخی از مهم‌ترین کارهای صورت گرفته پیشین در حوزه تشخیص سناریوی حملات چندمرحله‌ای تشریح شده است. در بخش سوم، عملکرد چارچوب پیشنهادی به‌همراه وظایف آن در هر یک از مودهای کاری برون‌خط و برخط شرح داده می‌شود. در بخش چهارم، نتایج مربوط به ارزیابی روش پیشنهادی آورده شده است. در انتها نیز در بخش پنجم، نتیجه‌گیری از مقاله آورده شده است.

۲- کارهای مربوطه

تاکنون پژوهش‌های زیادی در زمینه همبسته‌سازی هشدارها در جهت کشف سناریوی حملات چندمرحله‌ای صورت گرفته است. شین و همکاران [۸] چارچوبی مبتنی بر داده‌کاوی برای همبسته‌سازی هشدارها ارائه کرده‌اند. این چارچوب شامل یک مؤلّفه‌ی استخراج‌کننده قوانین وابسته‌سازی جهت کشف قوانین وابسته‌سازی که مشابه الگوریتم Apriori عمل می‌کند. دین و همکاران [۹] روشی مبتنی بر جریان‌کاوی برای تولید سناریو ارائه نموده‌اند. در این روش، با دریافت هر هشدار جدید، احتمال تعلق آن هشدار به سناریوهایی که تاکنون به‌وجود آمده‌اند، بررسی شده و هشدار جدید به سناریویی که بیشترین احتمال تعلق به آن را دارد، می‌پیوندد. یانگ و زینفا [۱۰] روشی بر پایه تصفیه آماری ارائه کرده‌اند. در این روش ابتدا با استفاده از یک فاز تصفیه هشدار، هشدارهای غیرمعتبر و پراکنده جدا شده و سپس با استفاده از به‌دست آوردن دنباله‌های تکراری، الگوهای تکراری حملات کشف می‌شوند.

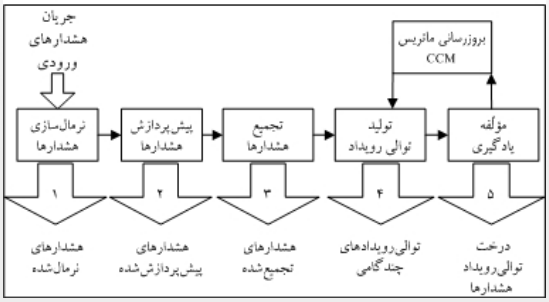
برخی از پژوهش‌های موجود در این زمینه از روش‌های یادگیری ماشین برای این منظور استفاده نموده‌اند. به‌عنوان نمونه، ژو و قربانی [۱۱] روشی بر مبنای شبکه‌های عصبی برای به‌دست آوردن همبستگی بین هشدارها معرفی کرده‌اند. در این روش از یک شبکه عصبی پرسپترون چندلایه[۷] برای تعیین میزان همبستگی هشدارها استفاده شده است. یکی دیگر از پژوهش‌های جدید وارد در این حوزه، پژوهش سلیمانی و قربانی [۱۲] است. آن‌ها روشی را برای تشخیص حملات چندمرحله‌ای ارائه کرده‌اند که بر مبنای استخراج توالی رویدادهای مشکوک به حمله در جریان هشدارهای نفوذی عمل می‌کند. روش آن‌ها بر اساس ترکیبی از تکنیک‌های جریان‌کاوی و استفاده از مفهوم درخت تصمیم است.

دسته دیگری از روش‌ها مبتنی بر تکنیک‌های آماری می‌باشند. فرهادی و همکارانش [۱۳]، روشی بر پایه جریان‌کاوی مبتنی بر تکنیک‌های آماری معرفی نموده‌اند. در این روش، سناریوی حملات با استفاده از میزان بسامد هشدارها در پنجره‌های زمانی مختلف و تحلیل همبستگی آن‌ها ساخته می‌شود. به‌دلیل عدم استفاده از مفهوم پنجره زمانی لغزان در روش معرفی شده، خطای منفی-غلط روش معرفی شده توسط آن‌ها نیز زیاد است. علاوه بر این، قادر به شناسایی سناریوی حملات شناخته شده نیست. در این مقاله قصد داریم تا با ترکیبی از تکنیک‌های آماری و جریان‌کاوی و با بهره‌گیری از مزایای پژوهش‌های معرفی‌شده پیشین، چارچوبی بلادرنگ برای همبسته‌سازی هشدارها در جهت استخراج سناریوی حملات چندمرحله‌ای پیشنهاد دهیم که در بخش بعدی به جزئیات این چارچوب خواهیم پرداخت.

۳- چارچوب همبسته‌سازی هشدار پیشنهادی

راه‌کارهایی که تاکنون با استفاده از روش‌های مختلف به استخراج سناریوی حملات پرداخته‌اند، ضعف‌هایی داشته و همچنان نیازمند راهکاری کارا برای استخراج سناریوها خواهیم بود [۵]. همان‌طور که پیش از این گفته شد، سیستم‌های تشخیص تشخیص نفوذ، قادر به تشخیص سناریوی حملات چندمرحله‌ای مهاجم نیستند و تنها برای هر یک از این گام‌های حمله، هشدارهایی را تولید و ثبت می‌کنند. بنابراین می‌توان یک حمله چندمرحله‌ای را از روی دنباله هشدارهایی که برای آن ثبت شده است، کشف نمود.

در پژوهش جاری، سعی کرده‌ایم تا از تکراری‌بودن هشدارها برای تشخیص گام‌های حمله بهره ببریم. به‌این ترتیب که با یافتن هشدارهای تکراری و تولید توالی رویداد پرتکرار به ساخت توالی رویداد همبسته محلی می‌پردازیم و سپس این توالی رویدادها را با توالی رویدادهای همبسته بازه‌های زمانی قبل همبسته می‌سازیم. چارچوب همبسته‌سازی هشدار پیشنهادی در



شکل (۱): چارچوب همبسته‌سازی هشدار پیشنهادی

شکل (۱) نشان داده شده است. این چارچوب، در دو فاز برون‌خط و برخط کار می‌کند. در فاز برون‌خط با یادگیری از رفتار مهاجم و تولید درخت توالی رویدادهای بحرانی، سناریوهای حمله را یاد گرفته و در فاز برخط با استفاده از دانش کسب شده در فاز برون‌خط، هشدارهای زودهنگام نفوذی را به مدیر امنیتی اعلان می‌نماید.

چارچوب همبسته‌سازی هشدار پیشنهادی بر این فرضیه عمل می‌کند که در یک شبکه بزرگ، تعداد زیادی حس‌گر تشخیص نفوذ وجود دارد. این مؤلّفه‌ها هشدارها را تحلیل می‌کنند و هشدارهای تولید شده توسط حس‌گرهای تشخیص نفوذ در یک گره مرکزی جمع‌آوری می‌گردند. هشدارهای تولید شده توسط حس‌گرهای تشخیص نفوذ به‌صورت دنباله‌ای از رویدادها به چارچوب همبسته‌سازی می‌رسند که نمونه‌ای از آن، در شکل (۲) نشان داده شده است. با رسیدن هشدارها به مؤلفه همبسته‌سازی، ابتدا نیاز است که هشدارها بر طبق یک قالب استاندارد که در این‌جا همان قالب IDMEF[۸] [۱۴] است، نرمال‌سازی شوند. در ادامه پس از انجام پیش‌پردازش‌های لازم بر روی هشدارهای دریافتی، هشدارها بر اساس برچسب زمانی‌شان که نمایانگر زمان تولید هشدار توسط یک حس‌گر تشخیص نفوذ است، مرتب می‌شوند. سپس برای تحلیل‌های بعدی وارد مؤلفه تجمیع می‌گردند. در ادامه وظایف هر مؤلفه از چارچوب همبسته‌سازی پیشنهادی تشریح شده است.

شکل (۲): دنباله هشدارهای دریافتی توسط چارچوب همبسته‌سازی

۳-۱- مؤلّفه تجمیع

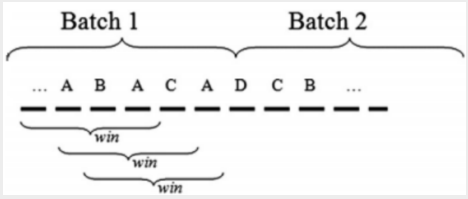
مؤلّفه تجمیع[۹] هشدارها تعداد هشدارهای تولید شده توسط حس‌گرهای تشخیص نفوذ شبکه تحت نظارت را کاهش می‌دهد تا در ادامه مدیریت و تحلیل هشدارها توسط سایر مؤلّفه‌های چارچوب همبسته‌سازی تسهیل یابد. یک هشدار از مجموعه‌ای از صفات خاصه بر اساس ویژگی‌های خاصی از ترافیک شبکه تشکیل شده است. در این مقاله برای پردازش هشدارها از مهم‌ترین ویژگی‌های یک هشدار (نظیر آدرس IP مبدأ، آدرس IP مقصد، شماره درگاه‌های ماشین‌های مبدأ و مقصد، نوع نفوذ یا نوع هشدار، شدّت[۱۰] حمله و نیز برچسب زمانی[۱۱] هشدار جهت همبسته‌سازی استفاده شده است.

در مؤلّفه تجمیع، همه هشدارهای دریافتی از حس‌گرهای تشخیص نفوذ در مدت طول یک پنجره توالی رویداد که دارای نوع هشدارهای یکسان هستند و در یک بازه زمانی نزدیک به‌هم رخ داده باشند، با یکدیگر ادغام می‌شوند تا یک ابرهشدار به‌دست آید. تعداد ابرهشدارهای تحلیل شده توسط مؤلّفه تجمیع کمتر از تعداد هشدارهای اصلی است که در آن پنجره توالی رویداد تولید شده و به مؤلّفه مرکزی تجمیع رسیده‌اند. پس از تجمیع هشدارهای دریافتی، مؤلّفه‌های یک ابرهشدار در فرمول (۱) نشان داده شده است که در آن، $A_h[counter]$ برابر تعداد هشدارهای مشابه می‌باشد.

$$
\begin{aligned}
&A_h[srcIPs] = \{A_{t1}[srcIP], A_{t2}[srcIP], ..., A_{tn}[srcIP]\},\\
&A_h[dstIPs] = \{A_{t1}[dstIP], A_{t2}[dstIP], ..., A_{tn}[dstIP]\},\\
&A_h[srcPorts] = \{A_{t1}[srcPort], At2[srcPort], ..., A_{tn}[srcPort]\},\\
&A_h[dstPorts] = \{A_{t1}[dstPort], At2[dstPort], ..., A_{tn}[dstPort]\},\\
&A_h[t] = \{t_1, t_2, ..., t_n\},\\
&A_h[alertType] = A_{t1}[alertType],\\
&A_h[attackSeverity] = A_{t1}[attackSeverity],\\
&A_h[counter] = |A|
\end{aligned}
\qquad (۱)
$$

۳-۲- استخراج توالی رویداد

در ادامه هشدارهای تجمیع‌شده توسط مؤلّفه تجمیع چارچوب همبسته‌سازی هشدار پیشنهادی، به بخش‌های بزرگی به‌نام دسته[۱۲] گروه‌بندی می‌شوند. هر دسته از هشدارها به قسمت‌های کوچکتری به‌نام پنجره توالی رویداد تقسیم‌بندی می‌شوند. نحوه این عملیات در شکل (۳) نشان داده شده است.



شکل (۳): تقسیم دنباله هشدارها به دسته و پنجره هشدارها

مؤلفه استخراج توالی رویداد، با پردازش هشدارهای موجود در هر دسته، اقدام به تولید توالی رویدادها می‌نماید. یک توالی رویداد یک مجموعه مرتب جزئی[۱۳] از رویدادهای همزمان است که می‌تواند به‌صورت یک DAG[۱۴] تعریف شود.

در چارچوب همبسته‌سازی پیشنهادی از مفهوم ماتریس همبستگی علّی[۱۵] (CCM) برای تشخیص قدرت همبستگی[۱۶] دو هشدار که با یکدیگر رابطه علیت دارند و نیز به‌روزرسانی دانش همبسته‌سازی در طول دو مود کاری استفاده شده است. به‌عبارت دیگر CCM دانشی است که سامانه همبسته‌ساز از آن برای ساخت سناریوی حملات چندمرحله‌ای استفاده می‌نماید. ماتریس CCM معمولا یک ماتریس نامتقارن است. هشدارهای a_i و a_j همبسته نامیده می‌شوند اگر داشته باشیم؛ $\rho(@(a_i), @(a_j)) \geq \tau$، که در آن @ تابع تعیین‌کننده نوع یک هشدار و τ میزان آستانه همبسته‌سازی را مشخص می‌نماید.

هدف از مؤلفه استخراج توالی رویداد، پردازش هشدارهای موجود در دسته هشدارها از دنباله هشدار برای یافتن توالی رویدادهای سریال مکرر است که هر دو ابرهشدار متوالی از آن توالی رویداد، بر اساس ماتریس CCM به‌هم وابسته بوده و به‌عبارتی قدرت همبستگی آن‌ها از یک میزان آستانه (که در مسئله ما ۰٫۵ تعریف شده است) باشد. در این صورت این توالی رویداد سریال مکرر را یک توالی رویداد بحرانی می‌نامیم که می‌تواند به‌عنوان بخشی

از سناریوی حمله چندمرحله‌ای به درخت حمله اضافه شود. توالی رویدادهای بحرانی استخراج‌شده در هر پنجره توالی رویداد به سمت مؤلّفه یادگیری و تشخیص[۱۷] ارسال می‌شوند تا درخت حمله ساخته شود.

۳-۳- مؤلّفه یادگیری و تشخیص

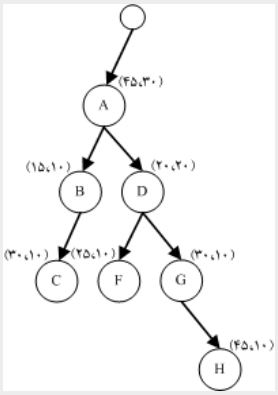
هدف اصلی مؤلّفه یادگیری و تشخیص، ساختن درخت حمله در مودهای کاری برون‌خط و برخط است که این کار با استفاده از تشخیص توالی رویدادهای بحرانی و به‌روزرسانی ماتریس CCM در طول هر دو مود کاری برون‌خط و برخط انجام می پذیرد. در چارچوب پیشنهادی، برای یادگیری و تشخیص حملات از یک روش یادگیری تطبیقی[۱۸] استفاده شده است.

در چارچوب همبسته‌سازی هشدار پیشنهادی، وقتی به‌اندازه یک پنجره توالی رویداد، هشدار دریافت می‌نماییم، در ابتدا، توالی رویدادهای پرتکراری را می‌یابیم که می‌تواند بر اساس CCM همبسته باشد. در هر دو مود کاری برون‌خط و برخط، با استفاده از توالی رویدادهای بحرانی استخراج‌شده یک درخت حمله ساخته می‌شود. در ادامه، توالی رویدادهای دوتایی[۱۹] را تولید می‌نماییم که از هر توالی رویداد بحرانی استخراج شده‌اند و آن‌ها را با نماد $\langle e_i,e_j\rangle$ نشان می‌دهیم که بیانگر آن است که e_j بلافاصله بعد از e_i در توالی رویداد بحرانی رخ داده است. برای مثال، مطابق شکل (۲)، اگر $\langle E,A,B\rangle$ یک توالی رویداد بحرانی باشد، آنگاه توالی رویدادهای دوتایی آن $\langle A,B\rangle$ و $\langle E,A\rangle$ به‌حساب می‌آیند. سپس، ماتریس CCM در هر دو مود کاری برون‌خط و برخط به‌روزرسانی می‌شود. این کار بر اساس توالی رویدادهای دوتایی تولید شده از توالی رویدادهای بحرانی استخراج شده صورت می‌گیرد. نحوه این بروزرسانی در بخش ۳-۵ آورده شده است.

۳-۴- ساخت درخت حمله

در هر دو مود کاری برون‌خط و برخط نیاز داریم تا رفتار مهاجم را مدل نماییم. این کار با استفاده از ساخت درخت حمله بر اساس توالی رویدادهای بحرانی استخراج شده در طول پنجره‌های توالی رویداد صورت می‌گیرد. برای این منظور، ما درخت حمله[۲۰] را تعریف می‌کنیم. تعریف صوری درخت حمله مشابه تعریف توالی رویدادها است. یک درخت حمله به‌صورت یک سه‌تایی $AT=(T,\leq,f)$ تعریف می‌شود که در آن، T یک مجموعه از ابرهشدارهای برچسب‌گذاری شده با استفاده از تابع برچسب‌گذاری f، ≤ یک ترتیب جزئی بر روی مجموعه T (مشخص‌کننده ترتیب زمانی ابرهشدارها) به‌گونه‌ای که برای هر $a_i\in T$، مجموعه $\{a_j\in T|\ a_j\leq a_i\}$ یک مجموعه خوش‌ترتیب[۲۱] بر اساس رابطه ≤ است. تابع برچسب‌گذاری f نیز، به هر گره (که یک ابرهشدار است) یک سه‌تایی شامل نوع هشدار، تعداد تکرار توالی رویداد و تعداد پنجره‌های توالی رویدادی که تا به‌اکنون دیده شده است، می‌باشد.

در ابتدا، یک درخت حمله، تنها شامل یک گره ریشه است. درخت حمله با درج توالی رویدادهای بحرانی استخراج شده به‌روزرسانی می‌شود. برای این منظور، پس از استخراج توالی رویدادهای بحرانی در یک پنجره توالی رویداد، آن‌هایی که یک زیرتوالی از یک توالی رویداد بحرانی دیگر هستند، از لیست توالی رویدادهای بحرانی استخراج شده حذف می‌شوند (برای اجتناب از درج توالی رویداد افزونه در درخت حمله). برای افزودن یک درخت توالی رویداد استخراج شده، اگر پیشوندی از توالی رویداد با پیشوندی از یک مسیر (بدون احتساب گره ریشه) در درخت حمله تطابق یابد، توالی رویداد موردنظر به درخت حمله الحاق می‌شود و بر اساس آن برچسب‌های گره‌های موجود در درخت حمله، به‌روزرسانی می‌شود (به‌جز نوع هشدارهایشان که بدون تغییر باقی می‌ماند). توجه داشته باشید که اگر هیچ مسیری در درخت با این شرایط یافت نشود، توالی رویداد موردنظر به ریشه درخت الحاق می‌شود. نمونه‌ای از ایجاد درخت حمله فرضی با استفاده از توالی رویدادهای بحرانی استخراج شده توسط مؤلّفه استخراج توالی رویداد، در شکل (۴) نشان داده شده است.



شکل (۴): یک درخت حمله ایجاد شده به‌کمک توالی رویدادهای بحرانی

۳-۵- بروزرسانی ماتریس CCM

همان‌گونه که پیش از این گفته شد، برای تخمین همبستگی علّی دو نوع هشدار، پس از استخراج توالی رویدادهای بحرانی در هر پنجره توالی رویداد، توالی رویدادهای دوتایی (نظیر $\langle e_i,e_j\rangle$، به‌گونه‌ای که e_j بلافاصله بعد از e_i در توالی رویداد باشد) تولید می‌شوند. این توالی رویدادهای دوتایی برای تخمین همبستگی علّی بین نوع هشدارها در یک فرآیند یادگیری تطبیقی مورد استفاده قرار می‌گیرند. برای این منظور، ابتدا چند ضریب را برای نوع هشدارهای متفاوت موجود در توالی رویدادهای دوتایی محاسبه نموده و در ادامه سلول مربوط به آن دو نوع هشدار در CCM را بر اساس ضرایب به‌دست آمده و مقادیر قبلی آن سلول بروزرسانی می‌نماییم.

برای تعیین مشابهت دو ابرهشدار، آدرس IP مبدأ، آدرس IP مقصد، شماره درگاه مبدأ و شماره درگاه مقصد مورد استفاده قرار می‌گیرد. برای این هدف، ما دو معیار مشابهت با نام‌های مشابهت IP و مشابهت درگاه برای دو ابرهشدار تعریف می‌نماییم. دو تابع IPSim و PortMatching (به‌ترتیب فرمول (۲) و فرمول (۳)) برای محاسبه میزان مشابهت دو مجموعه از آدرس‌های IP مبدأ و مقصد و محاسبه میزان مشابهت یک مجموعه از شماره درگاه‌های مبدأ و مقصد تعریف شده است. در فرمول (۲)، تابع $ipsim=k/32$ است که k برابر با بیشینه تعداد بیت‌های با ارزش دو آدرس IP است که با هم مطابقت داشته باشند. فرمول (۲) میزان مشابهت دو مجموعه از IPها نظیر ips_i و ips_j را محاسبه می‌نماید. بر اساس این دو معیار، مشابهت دو هشدار از فرمول (۴) به‌دست می‌آید که در آن w_1 و w_2 وزن تأثیر هر یک از آن‌ها بر میزان مشابهت کل است.

$$IPSim(ips_i, ips_j) = \frac{\sum_{IP_i\in ips_i}\sum_{IP_j\in ips_j} ipsim(IP_i, IP_j)}{|ips_i|.|ips_j|} \quad (۲)$$

$$PortMatching(ports_i, ports_j) = \frac{\sum_{Port_i\in ports_i}\sum_{Port_j\in ports_j} portmatching(Port_i, Port_j)}{|ports_i|.|ports_j|} \quad (۳)$$

(۴)

$$sim(a_1, a_2) = (w_1.\frac{IPSim(a_1[srcIPs], a_2[srcIPs]) + IPSim(a_1[dstIPs], a_2[dstIPs])}{2} + (w_2.\frac{PortMatching(a_1[srcPorts], a_2[srcPorts]) + PortMatching(a_1[dstPorts], a_2[dstPorts])}{2})$$

یکی از روش‌های مفید جریان‌کاوی، کاوش قوانین وابستگی[۲۲] است. قوانین وابستگی نمایانگر روابط جذاب بین ابرهشدارهای تجمیع‌شده در یک دنباله هشدار هستند. در حقیقت، در چارچوب پیشنهادی ما، هر توالی رویداد دوتایی از یک توالی رویداد بحرانی، می‌تواند به‌عنوان یک قانون همبستگی درنظر گرفته شود. با استفاده از روش ارائه‌شده توسط هارمز و دوگان [۱۵]، توالی رویدادهای دوتایی (قوانین وابستگی) برای تعیین میزان همبستگی ابرهشدارها مورد استفاده قرار می‌گیرند. به بیان دیگر، اطمینان یک قانون وابستگی بیانگر میزان همبستگی علّی ابرهشدار تالی به ابرهشدار مقدم در توالی رویداد است. این میزان اطمینان بر اساس فرمول (۵) به‌دست می‌آید که در آن تابع fr بیانگر بسامد توالی رویداد بحرانی مشاهده شده است. توجه داشته باشید که A_1 و A_2 برچسب‌هایی (نوع هشدارهایی) از ابرهشدار a_1 و a_2 هستند که در توالی رویداد وجود دارند. در ادامه فرمول (۶) که برای محاسبه اطمینان وزنی[۲۳] (wc) دو ابرهشدار به‌کار می‌رود، آورده شده است.

$$confidence(a_1, a_2) = \frac{fr(\langle A_1, A_2 \rangle)}{fr(\langle A_1 \rangle)} \quad (۵)$$

$$wc(a_1, a_2) = sim(a_1, a_2).confidence(a_1, a_2) \quad (۶)$$

در انتها نیز، در فرمول (۷) ، فرمول به‌روزرسانی سلول‌های CCM ارائه شده است که در آن α و β ضرایب وزنی (به‌طوری که $\alpha+\beta=1$ و $\alpha,\beta\leq 1$)، A_1 و A_2 برچسب‌هایی (نوع هشدارهایی) از ابرهشدار a_1 و a_2، و همچنین $\varphi(A_1,A_2)$ و $\varphi'(A_1,A_2)$ مقادیر قدیمی و جدید میزان همبستگی دو نوع هشدار هستند.

$$\varphi'(A_1, A_2) = \alpha.\varphi(A_1, A_2) + \beta.wc(a_1, a_2) \quad (۷)$$

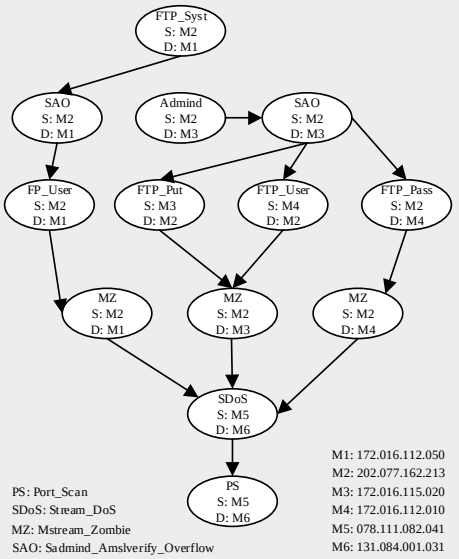
٤- پیاده‌سازی و ارزیابی

برای ارزیابی چارچوب همبسته‌سازی هشدار پیشنهادی از مجموعه داده DARPA 2000 با نام LLDoS2.0.2 (insider part) استفاده شده است. این مجموعه داده با استفاده از سیستم تشخیص نفوذ Snort مورد ارزیابی قرار گرفت و هشدارهای به‌دست آمده از آن، به‌عنوان ورودی چارچوب همبسته‌ساز خواهد بود. نوع هشدارهای به‌دست آمده از تحلیل ترافیک مجموعه داده LLDoS2.0.2 توسط Snort در مرجع [۱۳] آورده شده است. مهم‌ترین پارامترهای مورد استفاده در چارچوب همبسته‌سازی نیز در جدول (۱) نشان داده شده است. شبیه‌سازی‌های موردنظر نیز با زبان برنامه‌سازی جاوا و بر روی یک ماشین دو هسته‌ای با سرعت پردازنده 2.6GHz، مقدار حافظه 4GB از نوع DDR3 و نیز سیستم‌عامل ویندوز ۷ صورت گرفته است. علاوه بر مقداردهی پارامترها، باید CCM هم پیش از اجرای الگوریتم به‌روزرسانی گردد. برای مقداردهی اولیّه این مقادیر از برخی پژوهش‌هایی که روابط پیش‌نیازی بین هشدارها و میزان همبستگی آن‌ها را بیان نموده‌اند، استفاده کرده‌ایم [۱۳]. برای این منظور، میزان همبستگی هشدارهایی را که با یکدیگر همبستگی بالایی دارند را، مقدار ۰٫۵، هشدارهایی که میزان همبستگی کمتری دارند ولی تا حدودی همبسته هستند، مقدار ۰٫۲۵ و سایر مقادیر باقی‌مانده را برابر صفر درنظر گرفته‌ایم. پس از مقداردهی اولیّه پارامترها و ماتریس CCM، چارچوب پیشنهادی را بر روی هشدارهای موجود در مجموعه داده اعمال نموده‌ایم.

جدول (۱): پارامترهای چارچوب همبسته‌سازی هشدار پیشنهادی

نام پارامتر	مقدار پارامتر
طول دسته (Batch Time)	۱ ساعت
طول پنجره توالی رویداد (Episode Time Window)	۵ دقیقه
کمینه تکرار هشدار	۳
τ	۰٫۵
w1	۰٫۸
w2	۰٫۲
α	۰٫۴
β	۰٫۶
حداکثر اندازه طول سناریو	۶
پنجره زمانی برای تجمیع هشدارها	۱ دقیقه

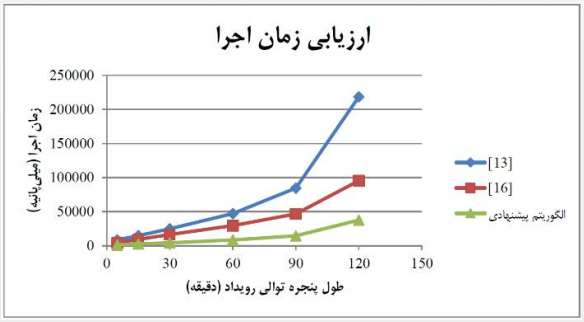
در ادامه، مهم‌ترین سناریوهای کشف شده توسط چارچوب پیشنهادی در شکل (۵) نشان داده شده است. نتایج به‌دست آمده از اجرای این الگوریتم در پردازش هشدارهای به‌دست آمده از مجموعه داده LLDoS2.0.2 قابل مقایسه با نتایج حاصل از کارهای دیگر انجام شده در این زمینه است. برای این منظور نتایج حاصل را با دو روش معتبر و جدید [۱۳] و [۱۶] که هر دو از مجموعه داده مورد استفاده در این پژوهش برای تشخیص سناریوی حملات استفاده نموده‌اند، مقایسه کرده‌ایم. سناریوهای حمله DDoS[۲۴] کشف شده در این پژوهش سناریوهای کاملی هستند که همه سناریوهای گزارش شده در پژوهش‌های ذکر شده را پوشش می‌دهند.



شکل (۵): سناریوی حملات چندمرحله‌ای در مجموعه داده **LLDoS2.0.2** به‌دست آمده توسط الگوریتم پیشنهادی

جنبه دیگری از الگوریتم که آن را مورد آزمون قرار داده‌ایم، زمان اجرای الگوریتم در پردازش هشدارهای دریافتی بلادرنگ با طول پنجره معین است که بر اساس شکل (۶)، نتایج حاکی از آن است که نسبت به پژوهش‌های صورت گرفته در زمینه همبسته‌سازی هشدار بر روی این مجموعه داده، زمان پردازش جهت تشخیص سناریوهای حمله به‌میزان قابل‌توجهی کاهش داده

شده و عملیات تشخیص سناریوها، در زمان کمتری و با دقت قابل‌قبولی صورت گرفته است.



شکل (۶): مقایسه زمان اجرای الگوریتم پیشنهادی با مراجع [۱۳] و [۱۶]

۵- نتیجه

در این مقاله، ضمن بررسی مفهوم همبسته‌سازی هشدار و مرور مهم‌ترین کارهای انجام شده در این زمینه، چارچوبی بلادرنگ برای همبسته‌سازی هشدارها ارائه شده است. در این چارچوب که هدف اصلی آن استخراج سناریوی حملات چندمرحله‌ای و به‌دست آوردن طرح‌های حمله مهاجم از جریان هشدارهاست، سعی شده است تا با بهره‌گیری از تکنیک‌های آماری و داده‌کاوی، الگوریتمی کارا برای این هدف ارائه شود. از مزایای چارچوب پیشنهادی می‌توان به بلادرنگ بودن روش، کارایی مناسب در محیط‌های با پردازش آنی و نیز ترکیب دانش‌ها اشاره کرد که در آن برخلاف بسیاری از پژوهش‌های صورت گرفته در این زمینه، که برای تعیین علیّت بین هشدارها یا از دانش قبلی موجود در مورد هشدارها و روابط بین آن‌ها و یا از روابط آماری بین هشدارهای تولید شده بهره می‌برند، از ترکیبی از این دو دانش برای یافتن علیّت بین هشدارها بهره می‌برد تا میزان همبستگی هشدارها را به‌طور کامل‌تر و دقیق‌تری به‌دست آورد.

مراجع

[1] Liao, H. J., Richard Lin, C. H., Lin, Y. C., & Tung, K. Y. (2013). Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications, 36 (1), 16-24.

[2] Lunt, T. F. (1993). A survey of intrusion detection techniques. Computers & Security, 12 (4), 405-418.

[3] Ramaki, A. A., Atani, R. E., Abadi, R. K. I., & Tavaghoe, M. (2013). Enhancement Intrusion Detection using Alert Correlation in Co-operative Intrusion Detection Systems.

[4] Valeur, F., Vigna, G., Kruegel, C., & Kemmerer, R. A. (2004). Comprehensive approach to intrusion detection alert correlation. Dependable and Secure Computing, IEEE Transactions on, 1 (3), 146-169.

[5] Salah, S., Maciá-Fernández, G., & Díaz-Verdejo, J. E. (2013). A model-based survey of alert correlation techniques. Computer Networks, 57 (5), 1289-1317.

[6] Ramaki, A. A., Amini, M., & Atani, R. E. (2014). RTECA: Real Time Episode Correlation Algorithm for Multi-Step Attack Scenarios Detection. Computers & Security. (to Appear)

[7] MIT Lincoln Laboratory, 2000 DARPA Intrusion Detection Scenario Specific Dataset, 2000. Available at: http://http://www.ll.mit.edu/.

[8] Shin, M. S., & Jeong, K. J. (2006). Alert correlation analysis in intrusion detection. In Advanced Data Mining and Applications (pp. 1049-1056). Springer Berlin Heidelberg.

[9] Dain, O. M., & Cunningham, R. K. (2001, June). Building scenarios from a heterogeneous alert stream. In Proceedings of the 2001 IEEE workshop on Information Assurance and Security (Vol. 235). West Point, NY, USA.

[10] Yang, L., & Xinfa, D. (2010, April). Alert Correlation Model Design Based on Self-regulate. In Multimedia and Information Technology (MMIT), 2010 Second International Conference on (Vol. 1, pp. 266-269). IEEE.

[11] Zhu, B., & Ghorbani, A. A. (2005). Alert correlation for extracting attack strategies (Doctoral dissertation, University of New Brunswick, Faculty of Computer Science).

[12] Soleimani, M., & Ghorbani, A. A. (2008, May). Critical episode mining in intrusion detection alerts. In Communication Networks and Services Research Conference, 2008. CNSR 2008. 6th Annual (pp. 157-164). IEEE.

[13] Farhadi, H., AmirHaeri, M., & Khansari, M. (2011). Alert correlation and prediction using data mining and HMM. ISeCure-The ISC International Journal of Information Security, 3 (2), 77-102.

[14] A. Li, RFC 4756, Intrusion Detection Message Exchange Format (IDMEF), 2006. Available at: http://tools.ietf.org/html/rfc4756.

[15] Harms, S. K., & Deogun, J. S. (2004). Sequential association rule mining with time lags. Journal of Intelligent Information Systems, 22 (1), 7-22.

[16] Kavousi, F., & Akbari, B. (2014). A Bayesian network-based approach for learning attack strategies from intrusion alerts. Security and Communication Networks, 7 (5), 833-853.

زیرنویس‌ها

[1] Intrusion Detection System
[2] False Positive
[3] Consequence
[4] Critical Episodes
[5] Online
[6] Offline
[7] Multilayer Perceptron
[8] Intrusion Detection Message Exchange Format
[9] Aggregation Component
[10] Severity
[11] Timestamp
[12] Batch
[13] Partial Order
[14] Directed Acyclic Graph
[15] Causal Correlation Matrix
[16] Correlation Strength
[17] Learning and Detection
[18] Adaptive Learning
[19] Binary Sub-Episodes
[20] Attack Tree
[21] Well-Ordered
[22] Association Rule Mining
[23] Weighted Confidence
[24] Distributed Denial of Service