

تشخیص حملات سایبری پیشرفته با استفاده از مدل‌سازی رفتاری مبتنی بر پردازش زبان طبیعی

کورش داداش‌تبار احمدی^{۱*}، مرجان خیرخواه^۲، علی جبار رشیدی^۳

۱- استادیار، ۲- دانشجوی کارشناسی ارشد، ۳- دانشیار، دانشگاه صنعتی مالک اشتر

(دریافت: ۹۶/۰۹/۱۰، پذیرش: ۹۶/۱۱/۱۴)

چکیده

رشته حملات پیچیده و ماندگار نفوذ به شبکه از مراحل نامحسوس و مخفی متعددی تشکیل شده‌اند. یکی از دلایل ناکارآمدی سامانه‌های تشخیص نفوذ در برابر این حملات، استفاده از سازوکار دفاعی مبتنی بر آنالیز ترافیک شبکه‌ای سطح پایین است که در آن به روابط پنهان بین هشدارها توجه نمی‌شود. فرض ما این است که اطلاعات ساختاری پنهان در داده‌های ترافیکی وجود دارند و ما می‌خواهیم در ترافیک شبکه‌ای قواعدی مانند قواعد زبان تعریف کنیم و آن را برای توصیف الگوهای فعالیت‌های شبکه‌ای بدخواهانه به کار بگیریم. به این وسیله می‌توانیم مسئله کشف الگوهای سوء استفاده و ناهنجاری را همانند مسئله یادگیری ساختارهای نحوی و قطعات مفهومی "زبان شبکه" حل کنیم. در این مقاله برای مدل‌سازی در مرحله تولید دنباله‌ها برای اولین بار در حوزه سایبری از یک خوشه‌بندی جدید به‌عنوان خوشه‌بندی MD_DBSCAN که یکی از انواع بهبودیافته خوشه‌بندی DBSCAN است، استفاده شده است. علاوه بر این، از یک الگوریتم حریمانه با الهام از القاء گرامر در پردازش زبان طبیعی استفاده شده تا با ادغام فعالیت‌های سطح پایین بتوانیم فعالیت‌های سطح بالا را کشف کنیم و روابط بین فعالیت‌های سطوح مختلف را تعریف کنیم. در بخشی از الگوریتم پیشنهادی برای کشف فعالیت‌های سطح بالا، برای اولین بار معیار شباهت ویرایش در خوشه‌بندی سلسله‌مراتبی به معیارهای موجود در الگوریتم پایه اضافه شده است. نتایج نشان می‌دهد دقت تشخیص در فعالیت‌های سطح بالا نسبت به فعالیت‌های سطح پایین با توجه به نمودار ROC حدود ۳۰٪ بیشتر است. همچنین، با تنظیم بهترین حد آستانه در الگوریتم تشخیص حملات، با در نظر گرفتن معیار FI، برای لغات سطوح یک تا سه به ترتیب به نتایج ۷۲/۳ و ۹۶/۲ و ۹۶/۴ در پنجره پیش‌بینی با اندازه سه رسیده‌ایم که به‌طور کلی حدود ۰/۲ نسبت به الگوریتم پایه بهبود نشان می‌دهد.

واژگان کلیدی: رفتار، حملات ماندگار پیشرفته، حملات سایبری، ادغام داده، پردازش زبان طبیعی

۱- مقدمه

برچسب‌های از پیش تعیین شده استفاده می‌کنند که هم واقع‌گرایانه نیست و هم محدود به فعالیت‌های از پیش تعیین شده است. در حالی که معمولاً تشخیص فعالیت‌های سطح بالا به صورت عام‌تر، سودمندتر است.

تاکنون از روش‌های متعددی برای شناسایی فعالیت‌های سطح بالا مانند ادغام اطلاعات سطح بالا و پردازش زبان طبیعی استفاده شده است. روش به‌کاررفته در این مقاله برای استخراج فعالیت‌های سطح بالا استفاده از مفاهیم NLP و القاء گرامر متناظر با زبان شبکه می‌باشد.

پردازش زبان‌های طبیعی یکی از زیرشاخه‌های با اهمیت در حوزه علوم رایانه، هوش مصنوعی و نیز دانش زبان‌شناسی محاسباتی است که به تعامل بین رایانه و زبان‌های (طبیعی) انسانی می‌پردازد؛ پردازش زبان‌های طبیعی عبارت است از استفاده از رایانه برای پردازش زبان گفتاری و زبان نوشتاری. بدین معنی که رایانه‌ها را قادر سازیم که گفتار یا نوشتار

تشخیص فعالیت‌های سطح بالای رفتار از طریق ادغام داده‌های حسگرهای مختلف، موضوعی است که کاربردهای متنوعی دارد. در این نوع سامانه‌های چندحسگری تشخیص دقیق فعالیت‌ها امکان‌پذیر شده است. در جاهایی که داده‌های ورودی معمولاً دنباله‌ای تک‌بعدی نیستند، ادغام حسگرهای چندگانه می‌تواند با پیش‌پردازش صورت‌گرفته بر روی این داده‌های چندبعدی تشخیص فعالیت‌ها را بهبود بخشد. طبق مطالعات انجام‌شده چالشی که در به‌کارگیری این سامانه‌ها با آن مواجه هستیم، حسگرآمیزی ناهمگن^۱ است.

چالش بعدی در این حوزه، نیاز به تشخیص فعالیت‌های سطح بالا بدون نظارت است؛ زیرا روش‌های نظارت‌شده از

* رایانامه نویسنده مسئول: Dadashtabar@mut.ac.ir

1- Heterogeneous sensor fusion

شده تا بتوانیم با روش‌های نوین داده‌کاوی به استخراج الگوهای رفتاری و کشف فعالیت‌های سطح بالای یک حمله سایبری بپردازیم. Yan و همکاران برای کشف فعالیت‌های سطح بالا از روش‌های داده‌کاوی نوین مانند استفاده از آمار مربعات‌خی دو بهره گرفته‌اند. به غیر از روش‌های پیشین، آنالیز مؤلفه اصلی [۶]، الگوریتم‌های ژنتیک [۷] و شبکه‌های عصبی مصنوعی [۸]، استنتاج فازی [۹] همگی روش‌های کشف نفوذ معروفی هستند که در آنالیز ترافیک شبکه برای کشف فعالیت‌های سطح بالای یک حمله سایبری پیچیده استفاده می‌شوند اما اغلب از کشف روابط میان دنباله‌های دریافتی از شبکه غافل می‌مانند.

تا کنون از روش‌های مبتنی بر زبان در کشف ناهنجاری‌ها در موضوعات مختلف استفاده شده است. Adler و همکاران با ترکیب علوم پردازش زبان طبیعی و ابر داده^۲ قادر به تشخیص تحریف علمی در متون Wikipedia شده‌اند [۱۰]. در تحقیق Lekhac و همکاران با استفاده از پردازش زبان طبیعی به تحقیق و کشف جرائم در اطلاعات بزرگ پرداخته شده و سرعت و دقت در این روند بهبود یافته است [۱۱]. Peng و همکاران از روش‌های مبتنی بر زبان برای کشف فعالیت‌های سطح بالای انسانی در یک بازی پینگ‌پنگ استفاده کرده و نتایج آن‌را با استفاده از روش مدل مخفی مارکف مقایسه کرده‌اند [۱۲]. با الهام از این رویکردها ما به دنبال روشی هستیم که با استفاده از کشف فعالیت‌های سطح بالا به تشخیص ناهنجاری در یک شبکه منجر می‌شود.

تاکنون کمتر از روش‌های مبتنی بر زبان برای کشف نفوذ حملات سایبری استفاده شده است. تنها محققانی همچون Rieck [۱۳] پیشنهاد استخراج کردن ویژگی‌های زبان نظیر n-gram ها و کلمات را از روی حداکثر بار ارتباط و اعمال کشف ناهنجاری بدون نظارت را اعمال نمودند. تفاوت این کار با تحقیق ما در این است که آن‌ها فقط اطلاعات محلی دنباله‌های بایت را استفاده می‌کردند اما رویکرد ما وابستگی در فواصل دوربین الگوهای زبانی و فعالیت‌های حمله را به‌منظور تشخیص تهدیدهای سایبری مدل‌سازی می‌کند.

۳- روش پیشنهادی برای تشخیص فعالیت‌های

سطح بالای حملات سایبری

روش به‌کار گرفته‌شده برای تشخیص حملات سایبری روش حریم‌نامه‌ای است که با الهام از القاء گرامر در پردازش زبان طبیعی به‌دست آمده است. ما به دنبال ارتباط پنهان موجود بین هشدارها هستیم و قصد داریم با داشتن یک ترکیب منطقی از

تولیدشده در قالب و ساختار یک زبان طبیعی را تحلیل و درک نموده یا آن را تولید نمایند. از کاربردهای NLP می‌توان به فهم زبان طبیعی، برچسب‌گذاری اجزای کلام، رفع ابهام و استخراج روابط اشاره کرد. در روش پیشنهادی، علاوه بر استخراج روابط میان دنباله‌های دریافتی از شبکه، از

روش عمقی که در رفع ابهام در زبان طبیعی وجود دارد الهام می‌گیریم. زیرا روش‌های عمقی، دانش عمیق‌تری از کلمه را متصور می‌شوند. رفع ابهام از معنای یک کلمه برمی‌گردد به این‌که آن کلمه در چه جمله‌ای به‌کاررفته و با چه کلماتی هم‌نشین گردیده است، چراکه یک کلمه در یک جمله یک معنی می‌دهد، درحالی‌که همان کلمه در جمله دیگر معنای دیگری به خود می‌گیرد.

در نهایت، هدف ما کشف الگوهای رفتاری سطح بالای اصلی در ترافیک شبکه است که احتمال دارد نشانه‌های اولیه حملات سایبری باشند. سامانه ما قادر به کشف حملات سایبری در مراحل اولیه آن است به‌طوری‌که برخی از اقدامات دفاعی می‌توانند قبل از به خطر افتادن واقعی سامانه انجام شوند.

بعد از این‌که در بخش اول مزایای به‌کارگیری روش‌های پردازش زبان طبیعی و استخراج رفتارهای سطح بالا را بیان شد، در بخش دوم، کارهای مرتبط گذشته را با تمرکز بر روش‌های استفاده‌شده برای تشخیص حملات سایبری چندمرحله‌ای مرور می‌کنیم. در بخش سوم، مراحل روش پیشنهادی را به همراه نوآوری‌های اضافه‌شده به الگوریتم پایه شرح می‌دهیم. در بخش چهارم به ارائه نتایج آزمایش‌ها و ارزیابی عملکرد روش پیشنهادی می‌پردازیم و در بخش پنجم در رابطه با نتایج و کارهای آینده بحث می‌کنیم.

۲- کارهای مرتبط

در تحقیق انجام‌شده توسط Kremmer و همکاران، سامانه کشف نفوذی براساس یکپارچه‌سازی روش‌های کشف ناهنجاری و سوء استفاده ارائه شده است [۱]. در کار ارائه‌شده، الگوی فعالیت‌های یک حمله سایبری استخراج می‌شود. روش ارائه‌شده برای تشخیص الگوهای رفتاری در این تحقیق به مانند روش‌هایی که در تحقیقات Debar و Sperotto و همکاران بوده عمدتاً از روش‌های سنتی یعنی کشف فعالیت‌های سطح پایین استفاده می‌کنند [۲-۳]. ما به دنبال روشی هستیم به مانند آن‌چه در تحقیقات Lee و همکاران [۴] و Yan و همکاران [۵] پیشنهاد

۱- استخراج روابط عبارت است از تعیین نسبت‌ها و ارتباطها میان هستارهای

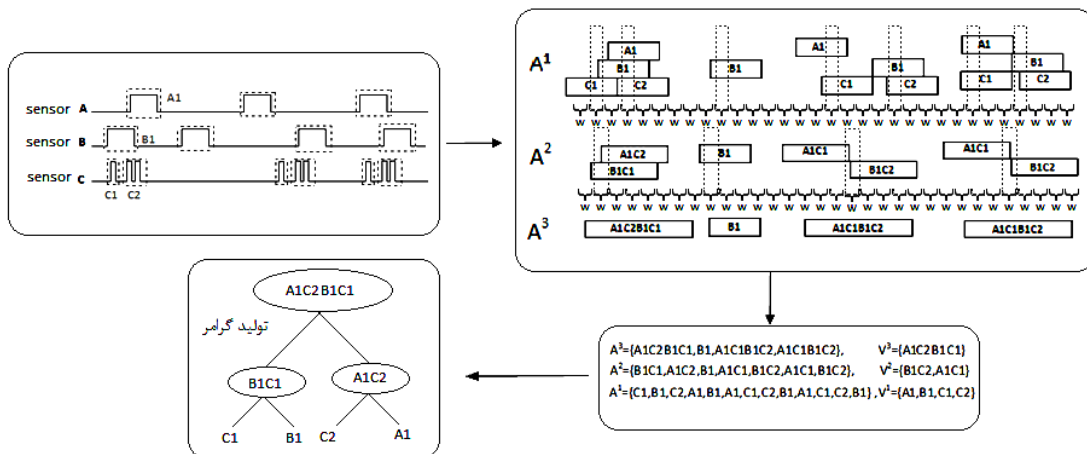
ادغام آن‌ها تصمیم‌گیری کرد (بالای سمت راست شکل). این پنجره از ابتدای دنباله شروع به حرکت روبه‌جلو کرده و فعالیت‌های نزدیک به هم را پیدا می‌کند. بدین ترتیب، می‌توان فعالیت‌های سطح بالا را استخراج نموده و با تولید گرامر از روابط به‌دست‌آمده، لغات سطح بالاتر را تولید کرد. در ادامه الگوریتم تولید گرامر را می‌بینیم:

```

Algorithm: grammar induction on network trace sequence
Input: S, network trace sequence obtained from MD_DBscan clustering
Input:  $\alpha T$ , merging threshold parameter
Input:  $\delta T$ , generalization threshold parameter
Output:  $G=\{V,R\}$ , discovered hierarchical grammar
Output: A, hierarchical network activities labeled using V
1 (A1,V1)= initialize (S)
2 (A',V')=(A1,V1)
3 l=1
4 While True do
5 l=l+1
6 (Al,Vl)= discover super-activities from (A',V') (merge each axl, ayl if collocation (axl, ayl) >  $\alpha T$ )
7 Break if |Vl|=0
8 For vi ∈ Vl, vj ∈ Vl do
9 Add edges(vi, vj) into R for all collocations
10 End
11 (A',V')= generalize vocabulary from (Al,Vl)
12 for each ai in Al
13 assign ai to cluster ci
14 Compute distance between clusters (maximum distance)
15 Find the closest (most similar) pair of clusters and merge them into a single cluster in V'
16 Compute distances (similarities) between the new cluster and each of the old clusters.
17 Repeat steps 15,16 until all distances are greater than (1- $\delta T$ )
18 End
19 V=(V1...Vl-1)
20 Return G={V,R},A
    
```

هشدارها به مانند آنچه در کشف bigramها در متون و اسناد انجام می‌شود، کار کنیم. در این روش، ابتدا فعالیت‌های سطح بالا از طریق در پراتز قرار دادن دو زیرفعالیت که از لحاظ آماری مکرراً در کنار هم قرار گرفته‌اند، شناسایی می‌شوند در مرحله بعد، فعالیت‌های مشابه به لحاظ شکلی یا معنایی را به فعالیت‌های سطح بالاتر تعمیم می‌دهیم. برای استفاده از رویکرد مبتنی بر زبان در یادگیری بدون نظارت ساختار فعالیت‌ها از روابطی که در ادامه براساس شکل (۱) توضیح داده شده است استفاده می‌کنیم. با فرض داشتن حسگرهای A و B و C که هر کدام دارای فعالیت‌های مرتبط هستند، ما متن فعالیت‌ها در سطوح مختلف را به شکل زیر تعریف می‌کنیم: $A=\{A^1, \dots, A^l\}$ که در آن $A^l=\{a_1^l, \dots, a_{|A^l|}^l\}$ و در آن هر a_i^l بیانگر فعالیت نام در سطح l است. در این جا A^1 نشان‌دهنده فعالیت‌های پایین‌ترین سطح است، در حالی که A^l نشان‌دهنده فعالیت‌های بالاترین سطح است. با توجه به A، در پایین شکل (۱) بعد از کشف با هم گذاری‌ها و خوشه‌بندی سلسله مراتبی، با داشتن درخت روابط، یک گرامر فعالیت $G=\{V,R\}$ تعریف می‌کنیم که در آن، V بیان‌گر روابطی است که ساختار گرامر را مشخص می‌کنند. به‌طور خاص، $V=\{V^1, \dots, V^l\}$ است که در آن، V^l بیان‌گر فهرست واژگان فعالیت l امین سطح معنایی است به‌طوری‌که هر فعالیت سطح l به شکل $a_i^l \in \{V^1, \dots, V^l\}$ است؛ بنابراین، هر رابطه مستقیم $(V_i^{l1} \rightarrow V_j^{l2})$ اگر $l_1 > l_2$ یا یک فعالیت ترکیبی 1 یا یک فعالیت تعمیم‌یافته 2 از V_j^{l2} است.

یافتن الگوهای تکرار شونده در سری‌های زمانی می‌تواند به ایجاد یک روش برای ساختن فرهنگ واژگان اولیه کمک کند. در بالای سمت چپ شکل (۱) مشاهده می‌شود که به زیر دنباله‌های مشابه برجسب‌هایی برای ساختن فرهنگ واژگان انتساب داده شده که با توجه به قرار گرفتن در یک پنجره زمانی می‌توان برای

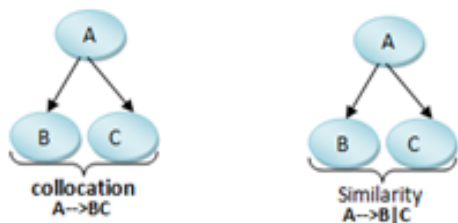


شکل (۱): رویکرد مبتنی بر زبان در یادگیری بدون نظارت براساس القاء گرامر G

تشخیص رفتار خاص حملات سایبری را ارائه دهیم از روابط و

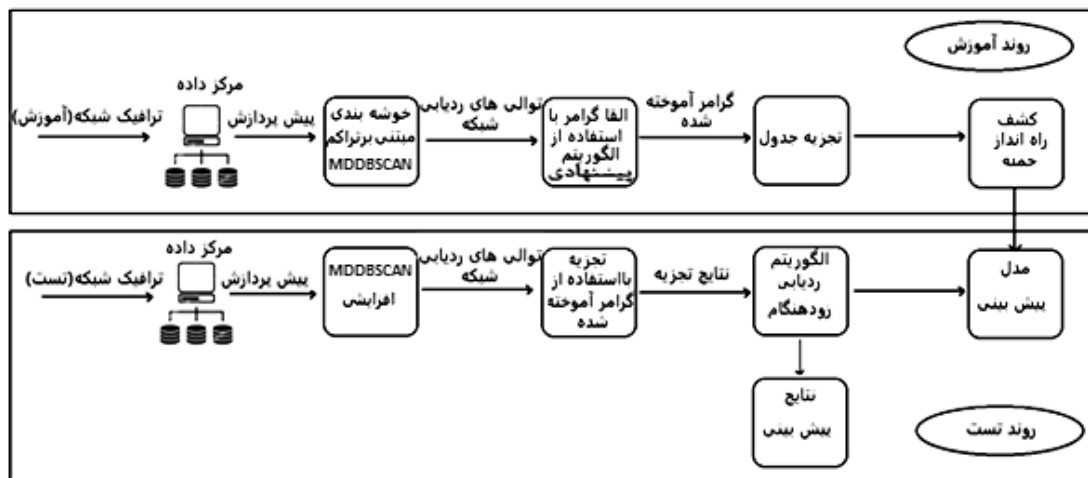
- 1- Super Activity
- 2- Generalized Activity

به طوری که $a_i^1 \in \{V^1U...UV^1\}$ پس، تعریف می‌کنیم که هر رابطه مستقیم $(V_i^{11} \rightarrow V_i^{12})$ اگر $I_1 > I_2$ و V_i^{11} یک فعالیت ترکیبی^۴ یا یک فعالیت تعمیم‌یافته از V_i^{12} است. همان‌طور که بیان شد گرامر G یک گرامر مستقل از متن است. گرامرهای مستقل از متن در زبان‌شناسی کاربردهای زیادی دارند و برای توصیف ساختار جملات و کلمات در زبان طبیعی به کار می‌روند. در گرامرهای مستقل از متن قواعد به شکل یک به یک، یک به چند و یک به هیچ تعریف می‌شوند و سمت چپ هر قاعده همیشه یک غیرپایانه قرار دارد. قواعد از یک نماد شروع به نام S آغاز می‌شوند. در گرامر مورد استفاده ما تمام نمادهای غیرپایانی در مجموع نمادهای شروع قرار می‌گیرند و قواعد می‌توانند از هر کدام از غیرپایانه‌ها شروع شوند. در مرحله اول ساخت گرامر که تمرکز به باهم‌گذاری لغات سطح پایین و ادغام آن‌ها در یک لغت داریم می‌توانیم قاعده تشکیل شده را بدین شکل تعریف کنیم: $A \rightarrow BC$ در مرحله دوم که بعد از اندازه‌گیری شباهت لغات آن‌ها را خوشه‌بندی کرده و لغات جدید را تعمیم می‌دهیم، می‌توانیم قواعد را به این شکل تعریف کنیم: $A \rightarrow B|C$



شکل (۲): ساخت قاعده تولید از درخت روابط (سمت چپ). این قاعده بدین معنا است که غیرپایانه A هم‌زمان B و C را در کنار هم تولید می‌کند. ساخت قاعده تعمیم از درخت روابط (سمت راست). این قاعده بدین معنا است که غیرپایانه A می‌تواند B یا C را تولید کند.

تعاریفی که در ادامه آمده است، استفاده می‌کنیم: فرض می‌کنیم سری‌های زمانی ترافیک شبکه‌ای به صورت $S = \{S_1, \dots, S_N\}$ به طول N باشند. هر رکورد S_i با دریافت اطلاعات از حسگرهای تشخیص نفوذ d بعدی به صورت $S_i = \{O_i^1, \dots, O_i^d\}$ تعریف می‌شود. دنباله ردیابی^۱ حاصل از ترافیک شبکه‌ای اخذ شده را می‌توان به صورت $T_i = F_M(S_i)$ در آن، نشان داد که در آن، $T_i = F_M(S_i)$ و F_M در این رابطه تابع نگاشتی است که دریافت اطلاعات از حسگرهای d بعدی S_i را به یک نماد ردیابی واحد T_i ترافیک شبکه‌ای تبدیل می‌کند. به بیان دیگر، دنباله ردیابی T یک نمایش تک‌بعدی از ترافیک شبکه‌ای خام است که انجام محاسبات بیشتر را به دلیل کاهش ابعاد آسان‌تر می‌سازد. به علاوه، در این الگوریتم فعالیت شبکه‌ای چندسطحی^۲ را با یک مجموعه به شکل $A = \{A^1, \dots, A^L\}$ با L سطح تعریف نمودیم که $A^1 = \{a_{11}^1, \dots, a_{1M}^1\}$ و در آن هر a_i^1 بیان‌گر آمین فعالیت در سطح l است. توجه کنید که A^1 در این‌جا به معنای فعالیت‌های پایین‌ترین سطح (یعنی $A_1 = T$) است درحالی‌که A^L دلالت بر بالاترین سطح دارد؛ چنانچه بخواهیم یک نگاشت مفهومی از سطوح مختلف معنایی از زبان طبیعی در زبان شبکه ارائه کنیم می‌توانیم پایین‌ترین سطح فعالیت را برابر با ویژگی‌های کلی ارتباطات در شبکه در نظر بگیریم، سطح بالاتر از معنا که در زبان طبیعی کلمات هستند می‌توانند به فعالیت‌های واحد تجزیه‌ناپذیر در شبکه نگاشت شوند و سطح بالاتر که در زبان طبیعی جملات هستند به وظایف در شبکه که دنباله‌ای از عملیات هستند، نگاشت می‌شوند. گرامر فعالیت شبکه‌ای $G = \{V, R\}$ به صورت تعریف می‌شود که V دلالت بر فهرست واژگان فعالیت^۳ در سطوح مختلف با L سطح دارد و R دلالت بر روابطی که ساختار گرامری را مشخص می‌کنند. دقیق‌تر بگوییم، $V = \{V^1, \dots, V^L\}$ که V^L بیان‌گر فهرست واژگان فعالیت در سطح مفهومی L ام می‌باشد



شکل (۳): مراحل تشخیص فعالیت‌های سطح بالای یک حمله‌ی سایبری

- 1- Trace Sequence
- 2- Multi-Level Network Activity Text
- 3- Activity vocabulary

سطح پایین

برای کشف ابرفعالیت از میان فعالیت‌های شبکه‌ای سطح پایین، ابتدا فرض می‌کنیم که جفت فعالیت‌هایی که به‌طور مشترک در یک پنجره اتفاق می‌افتند ممکن است مؤلفه‌هایی از یک فعالیت ترکیبی باشند. برای یک جفت فعالیت معین $[v_i^1, v_j^1]$ ، فرکانس رخ دادن مرزی v_i^1 و v_j^1 و نیز فرکانس رخ دادن مشترک آن‌ها را در دنباله ردیابی جمع‌آوری می‌کنیم. بر اساس این فرکانس‌های مرزی و مشترک، باهم‌گذاری^۷ محاسبه می‌شود تا تعیین شود که آیا این دو تشکیل یک فعالیت ترکیبی را می‌دهند یا خیر. برای مسئله ما به‌طور خاص، باهم‌گذاری بین دو فعالیت اشاره می‌کند که رخ دادن هم‌زمان آن‌ها ناشی از وابستگی ذاتی بین آن‌ها است و صرفاً تصادفی نیست. اهمیت مجموعه آماری می‌تواند با ساخت جدول احتمال وقوع به ازای هر جفت فعالیت $[v_i^1, v_j^1]$ و محاسبه آمار مجذور کای (X^2) اندازه‌گیری شود [۱۷] که بعداً با یک پارامتر آستانه‌ای (نظیر $X^2_{0.05}(1)=3.841$) که مقادیر بحرانی آن در جدول آمار مربعات کای آورده شده) مقایسه شود تا برای ادغام $[v_i^1, v_j^1]$ و ساخت یک فعالیت ترکیبی تصمیم‌گیری کند.

آزمون کای دوی دومتغیره مواقعی استفاده می‌شود که اطلاعات جمع‌آوری شده به فراوانی مشاهده شده و طبقات دو متغیر تعلق دارد بدین معنی که یک گروه آزمودنی براساس دو متغیر در طبقات مختلف طبقه‌بندی می‌شوند و سؤالی که در چنین مواردی مطرح است، وجود یا عدم وجود ارتباط بین متغیرها است؛ به عبارت دیگر، هدف این آزمون پاسخگویی به این سؤال است که آیا رابطه معنی‌داری بین دو متغیر وجود دارد یا خیر.

برای آزمون این فرضیه غالباً مشاهدات در یک جدول توافق $(r \times c)$ که r تعداد سطرها و c تعداد ستون‌ها است جمع‌آوری می‌شوند. آماره آزمون مشابه آزمون نیکویی برآزش به صورت رابطه (۱) به دست می‌آید:

$$X^2 = \sum_{i,j} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (1)$$

مرحله سوم: تعمیم فهرست واژگان با استفاده از خوشه‌بندی سلسله مراتبی

کشف باهم‌گذاری زمانی تأثیر دارد که دنباله‌های تکرارشونده‌ای از زیرفعالیت‌ها^۸ وجود داشته باشند. دقیق بودن این تکرارها برای فعالیت‌های شبکه‌ای به دلیل از دست رفتن اطلاعات در زمان انجام خوشه‌بندی احتمال کمی دارد. به منظور کشف بهتر فعالیت‌های سطح بالا در چنین شرایطی، ما باید قبل از ساخت جداول

حمله، می‌پردازیم. در حقیقت به‌مانند آن‌چه در شکل (۳) نشان داده شده است، عمل می‌کنیم. در ادامه مراحل الگوریتم را شرح می‌دهیم.

مرحله اول: استخراج نمادهای اولیه از ترافیک شبکه اخذشده با استفاده از MDDBSCAN

در این مرحله، ابتدا پیش‌پردازشی بر روی ترافیک شبکه‌ای، انجام می‌گیرد. ما براساس رویدادننگاری‌هایی^۱ که ورودی سامانه کشف نفوذ پیشنهادی هستند، به جمع‌آوری آمار و ویژگی‌های مرتبط، خواه پیوسته یا گسسته می‌پردازیم تا بتوانیم مفاهیم سطحی^۲ ترافیک شبکه را دریافت کنیم. این ویژگی‌ها برای هر ردیابی شبکه به مبانی ترانکشن‌های شبکه نظیر انواع پروتکل‌ها محدود نمی‌شوند بلکه ویژگی‌های سطح بالای ارتباطات شبکه‌ای نظیر مدت ارتباط و درصد خطاهای ارتباطاتی را نیز می‌توان تعریف کرد که به تشخیص ارتباطات عادی از حملات کمک می‌کنند. بعد از مرحله پیش‌پردازش داده، ترافیک شبکه با استفاده از الگوریتم‌های خوشه‌بندی^۳ به یک دنباله ردیابی تبدیل می‌شود. به طوری که ردیابی‌های شبکه‌ای مشابه توسط نماد یکسانی در دنباله ردیابی نمایش داده می‌شوند؛ یعنی، هر رکورد تراکنش ترافیک شبکه به یک نماد واحد مشابه یا یک کلمه در زبان طبیعی تبدیل می‌شود که منجر به یک دنباله ردیابی $T = \{T_1, \dots, T_N\}$ می‌شود. دنباله T بیان‌گر ترافیک شبکه شامل N تراکنش است که در آن، $T_i = F_M(S_i)$ و F_M تابع تبدیل از ترافیک شبکه‌ای خام به نمادهای ردیابی مبتنی بر نتایج خوشه‌بندی است. در همین مرحله برای انجام خوشه‌بندی ما از الگوریتم MD_DBSCAN^۴ که از الگوریتم‌های بهبودیافته MD_DBSCAN^۵ استفاده کرده‌ایم. الگوریتم MD_DBSCAN می‌تواند خوشه‌هایی با چگالی‌های متفاوت، یعنی خوشه‌هایی که در تراکم متفاوت باشند را پیدا کند. همچنین، در دنباله‌های جدید به علت پویایی فضای داده از الگوریتم IMD_DBSCAN^۶ استفاده کرده‌ایم. این الگوریتم برای به‌روزرسانی خوشه‌ها نیاز به پرس‌وجوی ناحیه‌ای بسیار کمتری دارد و دارای افزایش سرعت بسیار خوبی است. برای محاسبه شباهت هنگام خوشه‌بندی، به دلیل وجود ویژگی‌های پیوسته و گسسته در مجموعه دادگان، از معیار شباهت Gower [۱۶] در الگوریتم MD_DBSCAN استفاده شده است.

مرحله دوم: کشف فعالیت‌های ترکیبی مبتنی بر فعالیت‌های

1- Log
2- Shallow Semantics
3- Clustering
4- Multi_Density_Dbscan
5- Density Based Spatial Clustering Application With Noise
6- Incremental Md_Dbscan

7- Collocation
8- Sub-Activity

$$\theta_e = 2 * \frac{\text{precision} * \sqrt{\text{recall}}}{\text{precision} + \sqrt{\text{recall}}} \quad (6)$$

۲- شباهت زمینه (θ_x): در شباهت زمینه فعالیت‌هایی که در محیطی مشابه رخ می‌دهند را بررسی می‌کنیم. ما زمینه یک فعالیت را به صورت $\vec{c} = (n_1, \dots, n_{|V|})$ ارائه می‌کنیم که بیان‌گر تعداد دفعات رخ دادن مشترک هر یک از فعالیت‌ها با فعالیت مورد نظر است. سپس شباهت زمینه θ_x بین دو فعالیت را با فاصله کسینوسی بین بردارهای زمینه آن فعالیت‌ها به‌مانند آنچه در رابطه (۷) آمده است، در نظر می‌گیریم.

$$\theta_x = \frac{c_1 \cdot c_2}{|c_1| |c_2|} \quad (7)$$

۳- شباهت با معیار فاصله ویرایش (d_0): فاصله ویرایش یا فاصله لون‌اشتاین برای محاسبه میزان تفاوت میان دو رشته در علوم رایانه و نظریه داده‌ها استفاده می‌شود. فاصله ویرایش عبارت است از کمترین تعداد عملیات مورد نیاز برای تبدیل یک رشته به رشته دیگر. این عملیات شامل حذف، اضافه و تبدیل است. یک الگوریتم رایج برنامه‌نویسی پویا برای محاسبه فاصله لون‌اشتاین شامل استفاده از یک ماتریس $(m+1) \times (n+1)$ می‌شود، که n و m طول دو رشته هستند. این الگوریتم برپایه الگوریتم wagner_fischer برای ویرایش فاصله است. به‌عنوان مثال، برای تبدیل رشته سمت چپ به رشته سمت راست، سه مرحله عملیات نیاز داریم:

A6A2A3A3A7A4 → A1A2A3A3A2A4A5

مرحله اول: جایگزینی A1 به جای A6، مرحله دوم: جایگزینی A2 به جای A7 و مرحله سوم: اضافه کردن A5 در انتها؛ بنابراین، فاصله ویرایش بین این دو رشته ۳ است. برای نرمالیزه کردن معیار فاصله ویرایش در فاصله بین صفر و یک و تبدیل آن به معیاری برای شباهت از رابطه (۸) استفاده می‌کنیم:

$$\text{Edit distance similarity} = \frac{\max(\text{length}(v_1), \text{length}(v_2)) - \text{editdistance}(v_1, v_2)}{\max(\text{length}(v_1), \text{length}(v_2))} \quad (8)$$

خوشه‌بندی، نیازمند تجمیع شباهت‌های محتوا و زمینه و ویرایش در یک مقیاس شباهت واحد θ هستیم. براساس مشاهده، میانگین‌های حسابی، هندسی و هارمونیک، همگی گزینه‌های معقولی هستند. اگرچه به‌منظور این که یکی از شباهت‌ها (و نه لزوماً هر سه) را برای تعمیم کافی بسازیم، به میانگین حسابی نیاز داریم. این θ تجمیع سپس در یک الگوریتم خوشه‌بندی پیوند کامل^۵ استفاده می‌شود که طبق آن یک پیوند بین دو فعالیت

احتمال وقوع، فعالیت‌های مشابه را تعمیم دهیم. همان‌طوری که در [۱۲] پیشنهاد شده است، شباهت بین دو فعالیت در سطح یکسان می‌تواند برحسب شباهت محتوایی θ_e و شباهت متنی θ_x اندازه‌گیری شود. همچنین، در [۱۸] معیار فاصله ویرایش^۳ برای محاسبه شباهت دو فعالیت پیشنهاد شده است. ما این سه معیار را با محاسبه میانگین حسابی در قالب یک معیار شباهت واحد θ تجمیع می‌کنیم. این تجمیع θ سپس در یک الگوریتم خوشه‌بندی لینک کامل^۴ استفاده می‌شود که در آن یک لینک بین دو فعالیت وجود دارد تنها اگر θ شباهت آن‌ها بزرگ‌تر از آستانه θt باشد. در نتیجه، همه فعالیت‌های دسته‌بندی شده در خوشه یکسان به‌شدت شبیه به یکدیگر می‌باشند. دلیل استفاده از چند معیار شباهت، دستیابی به جنبه‌های متفاوتی از شباهت بین فعالیت‌ها است.

۱- شباهت محتوایی (θ_e): فعالیت‌های مشابه از نظر محتوا شامل دو حالت اصلی را تشکیل می‌دهد. حالت اول، وقتی است که فعالیت‌های چندگانه، هر دو تکرارهای زیرفعالیت یکسانی باشند. نظیر A0A0A0A0A0 و A0A0A0A0A0. حالت دوم وقتی است که دو فعالیت تنها در بخش جزئی از ترکیب خود باهم تفاوت دارند. برای مثال، A0A0A0A0A1 و A0A0A0A0A0. برای در نظر گرفتن این دو حالت برای دو عبارت (فعالیت) مشابه V_1 و V_2 ابتدا n-gram آن‌ها را محاسبه کرده‌ایم. فرض کنید $|V_1| \leq |V_2|$ و ۲ باشد.

$$\text{Precision} = \frac{|n\text{Gram}(v_1) \cap n\text{Gram}(v_2)|}{|n\text{Gram}(v_1)|} \quad (2)$$

$$\text{Recall} = \frac{|n\text{Gram}(v_1) \cap n\text{Gram}(v_2)|}{|n\text{Gram}(v_2)|} \quad (3)$$

که در آن‌ها، صورت کسر نشان‌دهنده 2-gram های مشترک بین رشته V_1 و V_2 می‌باشند.

برای به‌دست آوردن شباهت، دقت و بازخوانی با یک میانگین هارمونیک که در رابطه (۴) نشان داده شده است، ادغام می‌شوند تا F1 از رابطه (۵) به‌دست آید.

$$(a_1, \dots, a_n) = \frac{n}{\frac{1}{a_1} + \dots + \frac{1}{a_n}} \quad (4)$$

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (5)$$

درحالتی که دو عبارت از نظر محتوا بسیار شبیه بوده لیکن از نظر طول متفاوت هستند، رابطه (۵) را به شکل رابطه (۶) نرمالیزه می‌کنیم:

- 1- Content Similarity
- 2- Context Similarity
- 3- Edit Distance
- 4- Complete Link Algorithm

نمادهای آغازین معتبر یک اشتقاق از گرامر G باشند. در نهایت، نتایج در یک ساختار قابل استفاده مجدد به نام چارت ذخیره می‌شوند. وقتی گرامر G القاء می‌شود و بعد از اجرای الگوریتم CYK همه n سازنده معتبر سطح l یعنی $C^l = \{C_1^l, \dots, C_n^l\}$ در طول دنباله آموزش پیدا می‌شوند، دنباله آموزش را می‌توان توسط مجموعه‌ای از زوج‌های (c_i^l, w_i^l) به ازای سطح l نشان داد که W پنجره‌ای با طول ثابت است که شامل دنباله فعالیت‌های شبکه‌ای است که به c_i^l مربوط می‌شود و بعد از c_i^l رخ می‌دهد (شکل ۴).

الگوریتم CYK برای تشخیص سازندگان به شکل زیر استفاده شده است:

```

algorithm:structured trigger discovery with modified
CYK parsing
input: S, trace sequence of N symbols  $S_1, \dots, S_N$ 
input: the set of M non-terminal symbols  $R_1, \dots, R_M$ 
input: maxSpanLen, the maximum length of parsing
span to be considered
output: the location of all valid constituent with
corresponding derivation start symbols
1 let P be an  $N \times N \times M$  array of Booleans.
  Initialize all elements of P to false.
2 For (i=1; i<=n; i++) do
3 For each unit production  $R_m S_i$  do
4  $P[i, 1, m] = \text{True}$ 
5 End
6 End
7 // i iterates over the length of the span
8 For (i=2; i<= maxSpanLen; i++) do
9 // j iterates over the start of the span
10 For (j=1; j<=n-i+1; j++) do
11 // k iterates over the partition of the span
12 For (k=1; k<=i; k++) do
13 For each generalization rule  $R_A R_B$  do
14 If  $P[j, k, B]$  then
15  $P[j, k, A] = \text{true}$ 
16 End
17 End
18 For each production rule  $R_A R_B R_C$  do
19 If  $P[j, k, B]$  and  $P[j+K, i-k, C]$  then
20  $P[j, K, A] = \text{true}$ 
21 End
22 End
23 End
24 End
25 End
26 If any of  $P[j, i, x]$  is true (x is iterated over
the set of starting symbols R) then
27 Return
end
    
```

وجود دارد تنها اگر شباهت آن‌ها \emptyset بزرگ‌تر از یک آستانه \emptyset_{thr} باشد. در نتیجه، همه فعالیت‌هایی که باهم در یک خوشه دسته‌بندی می‌شوند، همگی بسیار شبیه به یکدیگر هستند.

مرحله چهارم: کشف "نشانه‌های شروع رفتار"

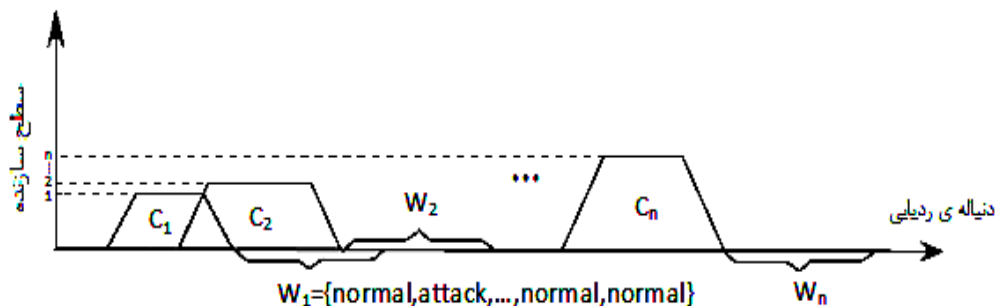
وقتی گرامر ترافیک شبکه القاء گردید، می‌توان از قواعد گرامری برای تجزیه^۱ دنباله‌های ردیابی شبکه استفاده کرد و در سطوح مختلف در طول دنباله ردیابی با استفاده از الگوریتم‌های تجزیه، همه ساختارهای معتبر زیردرخت تجزیه را به دست آورد که در پردازش زبان طبیعی (NLP) به عنوان سازندگان^۲ نامیده می‌شوند.

برای کشف سازندگان در متن، از الگوریتم تجزیه چارت CYK بهره می‌بریم که این الگوریتم از برنامه‌نویسی پویا استفاده می‌کند. الگوریتم CYK یک الگوریتم اشتقاق است که روی گرامرهای مستقل از متن عمل می‌کند. این الگوریتم برای تشخیص عضویت یا عدم عضویت یک رشته خاص در یک زبان مستقل از متن به کار می‌رود. روش این الگوریتم بالا به پایین بوده و از الگوریتم‌های پویا است. نسخه استاندارد این الگوریتم فقط روی گرامرهایی به شکل فرم نرمال چامسکی^۳ عمل می‌کند. البته تمام گرامرهای مستقل از متن قابلیت تبدیل به فرم نرمال چامسکی را دارند. زمان اجرایی این الگوریتم در بدترین حالت گرامر موجود است. هدف ما از اجرای این الگوریتم، یافتن تمام رشته‌های موجود در دنباله S است که در زبان مستقل از متن وجود دارند، علاوه بر آن، مکان این رشته‌ها در دنباله برای ما حایز اهمیت است. همچنین، به دلیل وجود قواعد تعمیم در گرامر، که قواعد یک به یک می‌باشند، قواعد به شکل فرم چامسکی نیست. با این توصیف، ما نیاز به ایجاد تغییراتی در الگوریتم اصلی داریم که در نهایت آن‌را به شکل زیر بازسازی می‌کنیم: فرض کنید گرامر مستقل از متن القاء شده یک تاپل چهار جزئی $G = (N, \Sigma, P, R_s)$ باشد که N, Σ, P, R_s به ترتیب بیان‌گر «مجموعه نمادهای غیر پایانی»، «مجموعه نمادهای پایانی»، «مجموعه قواعد تولید گرامر» و «نماد شروع شناسایی شده» باشند. می‌خواهیم بدانیم آیا دنباله‌ای از نمادهای w وجود دارد که به $L(G)$ تعلق داشته باشد یا خیر. به این ترتیب می‌توانیم همه سازنده‌های معتبر سطوح مختلف را در دنباله ردیابی پیدا کنیم. $L(G)$ همه دنباله‌هایی است که در تعداد متناهی از گام‌هایی از نماد آغازین متعلق به R_s قابل استخراج است که در مورد ما معادل N است یعنی همه نمادها در فهرست واژگان فعالیت سطح بالا می‌توانند

برای شناسایی "نشانه‌های شروع رفتار" سازنده که فعالیت‌های حمله در حال وقوع احتمالی را پیش‌بینی می‌کنند،

1- Trigger
2- Parse
3- Constituents
4- Chomsky Normal Form (CNF)

و یا تصویری که از عمل در ذهن خود تجسم می‌کند. در واقع، Triggerهایی که ما در دنباله آموزش پیدا می‌کنیم، سازندگانی هستند که با رفتار شبکه همبستگی خاصی دارند و معمولاً بعد از پدیدارشدن در دنباله، رفتار خاصی را به دنبال دارند که مقدار این همبستگی با رفتار را می‌توان از روش‌های آماری که در ادامه گفته می‌شود، محاسبه کرد.



شکل (۴): سازندگان سطوح مختلف و پنجره‌های فعالیت پیش روی آن‌ها در دنباله‌ی ردیابی

$$\frac{\left(8 - \left(10 * \frac{11}{18}\right)\right)^2}{\left(10 * \frac{11}{18}\right)} + \frac{\left(2 - \left(10 * \frac{7}{18}\right)\right)^2}{\left(10 * \frac{7}{18}\right)} + \frac{\left(3 - \left(8 * \frac{11}{18}\right)\right)^2}{\left(8 * \frac{11}{18}\right)} + \frac{\left(5 - \left(8 * \frac{7}{18}\right)\right)^2}{\left(8 * \frac{7}{18}\right)}$$

$$= 11.61 \geq \alpha_{0.95} = 0.384 \quad (9)$$

مرحله پنجم: کشف حمله سایبری در مرحله اولیه حمله

در این مرحله ابتدا همبستگی بین هر "نشانه شروع رفتار" سازنده شناسایی شده در دنباله ردیابی را با انواع حملات احتمالی بررسی می‌کنیم؛ یعنی با هر بار خواندن یک نماد ردیابی جدید تست، قادر به یافتن همه "نشانه شروع رفتار" سازنده معتبر سطوح مختلف به ازای هر پنجره فعالیت شبکه خواهیم بود.

فرض کنید که یک "نشانه شروع رفتار" سازنده معتبر C در دنباله تست شناسایی شود که W پنجره فعالیت مرتبط با آن است. ابتدا با مراجعه به جدول نگاشت‌های ذخیره شده که در گام کشف "نشانه شروع رفتار" تهیه کرده‌ایم، به دنبال همه انواع فعالیت‌های شبکه‌ای که ممکن است با C آغاز شوند، می‌گردیم که آن‌را با A نشان می‌دهیم. سپس، به منظور انجام پیش‌بینی خود، آمار مجذور کای را بین هر جفت (C, Ai) که Ai ∈ A با آستانه‌های ثابت مقایسه می‌کنیم. در این جا فرض می‌کنیم که A حاوی هم برچسب منفی (به معنای «عادی») و هم برچسب مثبت (به معنای «حمله»، «DoS») است زیرا اگر C فقط برای

مجدداً جدول احتمال رخداد را ایجاد کرده و آمار مجذور کای (X²) را برای اندازه‌گیری همبستگی بین هر سازنده و رفتار شبکه محاسبه می‌کنیم. نشانه‌های شروع رفتار یا Triggerها در علم NLP به معنای نمایه‌هایی است که قبل از انجام هر کار، عامل آن را در ذهن خود می‌سازد. به عبارت دیگر، این نمایه‌های داخلی مانند حرف‌هایی است که عامل قبل از انجام عمل با خود می‌گوید

با جستجوی در دنباله ردیابی آموزش از چپ به راست، فرکانس‌های مشترک همه جفت‌های (C_i⁺, A_i) به ازای سطح l را جمع‌آوری می‌کنیم که A_i ∈ W_i⁺ بیان‌گر برچسب نماد ردیابی است که می‌تواند دودویی (نظیر «عادی» یا «حمله») یا چندتایی (نظیر «عادی»، «DoS»، «probe») باشد. چنین تعدادی در یک جدول فرکانس مشترک^۱ (سمت چپ شکل ۵) ثبت می‌شود. جدول فرکانس حاشیه^۲ (وسط شکل ۵) نیز به‌طور مشابه ساخته می‌شود. سپس جدول احتمال وقوع (سمت راست شکل ۵) بر طبق جداول فرکانس مشترک و مرزی ساخته می‌شود. آمار مجذور کای هر جفت (C_i⁺, A_i) براساس جدول احتمال وقوع محاسبه می‌شود که برای استفاده بعدی یعنی پیش‌بینی رفتار در حال وقوع شبکه با داشتن دنباله جدید ردیابی در یک جدول، ذخیره می‌شود.

جدول فرکانس مشترک			جدول فرکانس حاشیه			جدول احتمال وقوع		
تکرار	نوع فعالیت	سازنده	تکرار	نوع فعالیت	سازنده			
8	normal	162	10	-----	162		162	~16
2	attack	162	8	-----	721	normal	8	3
3	normal	721	11	normal	-----	attack	2	5
5	attack	721	7	attack	-----		10	8
								18

شکل (۵): مثال از نحوه محاسبه آمار مجذور کای (X²)، جداول فرکانس مشترک و مرزی و جداول احتمال وقوع با استفاده از رابطه (۱)

1- Joint Frequency Table

2- Marginal Frequency Table

۴- نتایج و بحث

در ادامه به شرح آزمایش و روش‌های ارزیابی آن می‌پردازیم.

۴-۱- آزمایش‌ها

در شبیه‌سازی الگوریتم از دادگان KDD99 برای آموزش و تست الگوریتم پیشنهادی استفاده شده است [۱۵] که بر روی این دادگان، ابتدا نمونه‌گیری و سپس خوشه‌بندی با استفاده از الگوریتم MDDBSCAN اجرا شده است. از خروجی این مرحله برای ساخت گرامر همان‌گونه که گفته شد، استفاده شده است. در نهایت، برای ارزیابی دقت تشخیص فعالیت‌های سطح بالای یک حمله سایبری از ارزیابی گرامر ساخته‌شده نهایی بهره بردیم.

۴-۲- ارزیابی

ارزیابی نتایج آزمایش‌ها در دو مرحله برای خوشه‌بندی MDDBSCAN و برای گرامر ساخته‌شده نهایی انجام شده است.

۴-۲-۱- ارزیابی خوشه‌بندی MDDBSCAN

برای این که ارزیابی MDDBSCAN روی این دادگان را داشته باشیم و میزان بهبود نتایج آن را نسبت به الگوریتم DBSCAN بسنجیم، در یک آزمایش از ۸۰٪ داده‌ها (یک‌بار به شکل تصادفی و بار دیگر به شکل ترتیبی) به‌عنوان داده آموزشی استفاده کردیم و در دادگان تست، دادگان بدون برچسب را برحسب رأی اکثریت خوشه برچسب زدیم (نرمال و حمله). سپس نتایج به‌دست‌آمده را با نتایج به‌دست‌آمده از الگوریتم DBSCAN روی همین دادگان با $\minpts=4$ و $Eps=0.02$ مقایسه کردیم. برای این مقایسه از سه معیار Fowlkes.Mallows و Jaccard و درصد خطا استفاده کرده‌ایم.

$$Fowlkes.Mallows_index = \sqrt{\frac{TP}{TP+FP} \cdot \frac{TP}{TP+FN}} \quad (10)$$

جدول (۱): نتایج ارزیابی خوشه‌بندی با معیار Fowlkes Mallow

KDD99	KDD99	
۸۰٪ - ۲۰٪	۸۰٪ - ۲۰٪	دادگان الگوریتم
آموزشی - تست (ترتیبی)	آموزشی - تست (تصادفی)	
۰/۹۴	۰/۸۷	DBSCAN
۰/۹۸	۰/۹۱	MDDBSCAN

$$Jaccard_Index = \frac{|A \cap B|}{|A \cup B|} = \frac{TP}{TP+FP+FN} \quad (11)$$

شروع فعالیت‌های عادی امکان‌پذیر باشد، نیازی به گرفتن آمار از آن نداریم و با دیدن C در دنباله می‌توانیم به‌سادگی فعالیت بعدی را پیش‌بینی کنیم.

در نهایت، ما قادر به ارائه نتایج پیش‌بینی مبتنی بر الگوریتم زیر خواهیم بود که در آن $normalThr$ و $attackThr$ بیانگر دو مقدار آستانه‌ای آمار مجذور کای هستند. برای مثال اگر آمار مجذور کای جفت سازنده فعالیت (C,Ai) بزرگ‌تر از آستانه مربوط به خود باشد، پیش‌بینی مطابق با Ai تعیین می‌گردد.

الگوریتم تصمیم‌گیری نهایی برای پیش‌بینی نتایج به شکل زیر خواهد بود:

Algorithm: prediction of network behavior on the trace sequence

Input: S, the trace sequence, indexed by [0,1,...,N-1]

Input: chart, derived from CYK parsing on the sequence

Input: chiSqrMapping, the mapping from each constituent trigger to its X^2 statistics at each vocabulary level

Input: window size

Input: maxLevel, the maximum level of constituent

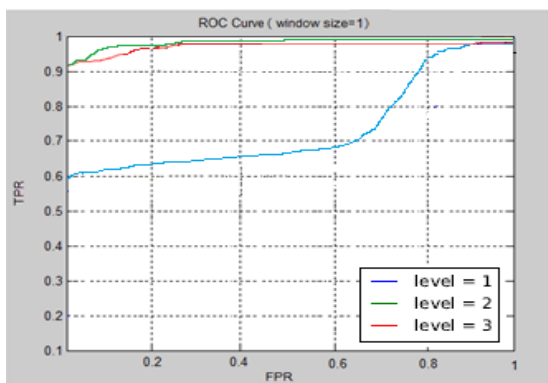
Output: prediction label for each valid trigger along the test sequence

```

1   startIndex=0
2   while startIndex <= N-windowSize do
3     triggerFound= false
4     prediction window=
5     Stest [startIndex+windowSize]
6     possible Triggers= find all valid constituent
       triggers for prediction window by looking
       up chart
7     target Trigger=get the constituent trigger up
       to level maxLevel from possible Triggers
8     A= get all possible network activity types
       that can be triggered by target trigger
9     If normal ∈ A then
10    If chiSqr Mapping
11    [(target Trigger,normal)] > normalThr then
12    Prediction= normal
13    Else
14    Prediction= attack
15    End
16    Else
17    Maxchisqr=max(getChisqr
       Values(chisqrMapping,A))
18    If maxchisqr > attachThr then
19    Prediction= attack
20    Else
21    Prediction= normal
22    End
23    End
24    startIndex= startIndex+1
25    end

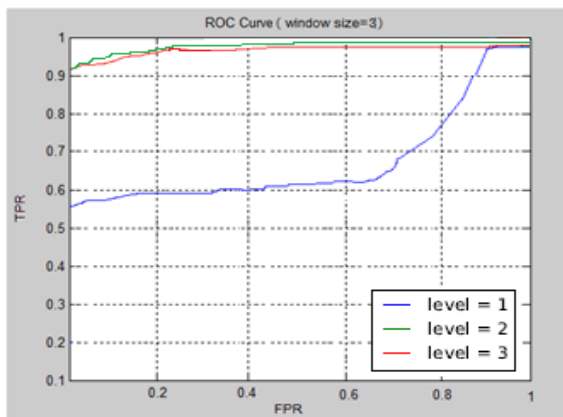
```

فاصله بیشتر اما با دقت کمتر می‌شود.



شکل (۸): نتایج الگوریتم و مقایسه نتایج به دست آمده از لغات سطوح

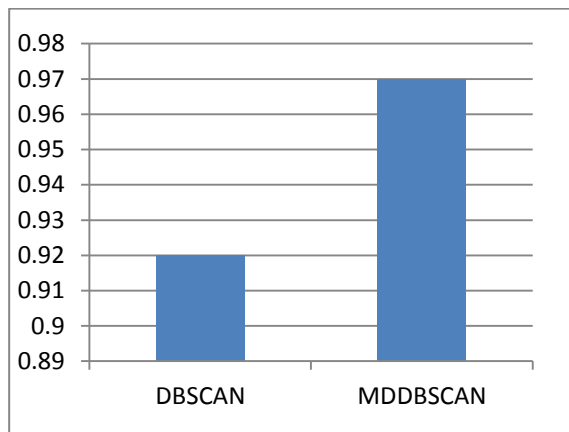
۱، ۲ و ۳



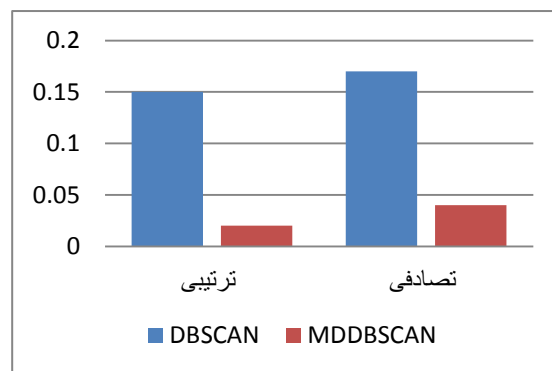
شکل (۹): نتایج افزایش اندازه پنجره پیش‌بینی تا اندازه ۳

۵- نتیجه‌گیری

هدف اصلی این مقاله تشخیص بهتر و سریع‌تر حملات سایبری چندمرحله‌ای است. بدین منظور از مدل‌سازی ساخت یافته رفتاری با الهام‌گرفتن از پردازش زبان طبیعی استفاده شده است. در روش پیشنهادی علاوه بر استخراج روابط میان دنباله‌های دریافتی از شبکه، از روش عمقی که در رفع ابهام در زبان طبیعی وجود دارد، الهام گرفته‌ایم. ما برای انجام کار خود از دادگان KDD99 که بیشترین دادگان مورد استفاده در این حوزه می‌باشد، استفاده کرده‌ایم. به‌علت داشتن ویژگی‌های پیوسته و گسسته در دادگان KDD99، ما برای خوشه‌بندی اولیه از معیار شباهت Gower استفاده کردیم که هر دو نوع ویژگی را در رابطه خود در برمی‌گیرد. خروجی این خوشه‌بندی لغات ابتدایی یا سطح اول در دنباله زبان شبکه را تشکیل می‌دهد. با تناظری که بین زبان طبیعی و زبان شبکه تعریف کردیم می‌توان با ترکیب لغات سطوح پایین‌تر به لغات سطوح بالاتر دست پیدا کرد. برای



شکل (۶): نتایج ارزیابی خوشه‌بندی براساس معیار میانگین Jaccard



شکل (۷): نتایج ارزیابی خوشه‌بندی با معیار درصد خطا

۴-۲-۲- ارزیابی الگوریتم

بعد از انجام خوشه‌بندی با هر بار کشف لغات در سطوح یک تا سه، الگوریتم ارائه‌شده برای تشخیص سریع حملات، اجرا شد و برای هر لغت کشف‌شده، تشخیص به‌صورت حمله یا نرمال صورت گرفت. در الگوریتم ارائه‌شده به متغیر آستانه حمله و نرمال مقادیر مختلفی دادیم و درنهایت، نتایج کلی الگوریتم را در نمودار ROC، با هر دو آزمایش با پنجره‌های پیش‌بینی ۱ و ۳ رسم کردیم.

شکل (۸) نتایج آزمایش با استفاده از پنجره پیش‌بینی با اندازه ۱ است. همان‌طور که در شکل دیده می‌شود نتایجی که از لغات سطوح دو و سه به دست آمده تفاوت زیادی با نتایجی که از سطوح اول به دست آمده دارد.

شکل (۹) فوق نتایج آزمایش با استفاده از پنجره پیش‌بینی با اندازه ۳ است. همان‌طور که انتظار می‌رود با افزایش اندازه پنجره پیش‌بینی دقت نتیجه افت می‌کند. زیرا با افزایش اندازه پنجره پیش‌بینی، ارتباط معنادار بین سازنده‌ها و رفتار شبکه کمتر خواهد شد. در واقع، افزایش اندازه پنجره منجر به تشخیص با

- [7] M. Saniee Abadeh, J. Habibi, and C. Lucas, "Intrusion detection using a fuzzy genetics-based learning algorithm," pp. 414-428, 2007.
- [8] G. Wang, J. Hao, J. Ma, and L. Huang, "A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering," pp. 6225-6232, 2010.
- [9] A. J. Rashidi, K. Dadashtabar Ahmadi, and F. Samsami Khodadad, "Projection of Multistage Cyber Attack Based on Belief Model and Fuzzy Inference," *Journal of electronical & cyber defence*, vol. 3, no. 2, 2015. (In Persian)
- [10] B. Thomas Adler, L. de Alfaro, S. M. Mola-Velasco, P. Rosso, and A. G. West, "Wikipedia Vandalism Detection: Combining Natural Language, Metadata, and Reputation Feature," *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 277-288, 2011.
- [11] N.-A. Le-Khac, M.-T. Kechadi, and M. Banerveld, "Performance Evaluation of a Natural Language Processing approach applied in White Collar crime investigation," *School of Computer Science & Informatics, University College Dublin Belfield, Dublin, Ireland*, 2014.
- [12] H.-K. Peng, P. Wu, J. Zhu, and J. Ying Zhang, "Helix: Unsupervised Grammar Induction for Structured Activity Recognition," *11th IEEE International Conference on Data Mining*, pp. 1195-1199, 2011.
- [13] K. Rieck and P. Laskov, "Detecting unknown network attacks using language models," *Third international conference on Detection of Intrusions*, pp. 74-90, 2006.
- [14] Sunoallah, Wesam Ashour and Saad, "Multi Density DBSCAN," *International Conference on Intelligent Data Engineering and Automated Learning*, pp. 446-453, 2011.
- [15] A. Özgür and H. Erdem, "A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015," *Ankara: s.n.*, Apr. 2016.
- [16] J. C. Gower, "A general coefficient of similarity and some of its properties," pp. 857-871, 1971.
- [17] W. G. Cochran, "The X2 test of goodness of fit," *Annals of Mathematical Statistics*, vol. 25, pp. 315-345, 1952.
- [18] Peterson, A. C. Lin, and L. Gilbert, "Activity Pattern Discovery from Network Captures," *IEEE Symposium on Security and Privacy Workshop*, 2016.

ترکیب لغات از فرمول آماری مربعات کای دو استفاده کردیم که با داشتن یک حد آستانه برای تصمیم‌گیری ترکیب یا عدم ترکیب دو لغت از سطح پایین‌تر به ما کمک کرد. در ادامه کار، با خوشه‌بندی سلسله مراتبی لینک کامل این ترکیبات را به لغات مشابه نیز تصمیم دادیم تا روابط و لغات جدید بیشتری تولید کنیم. در نهایت، با به‌دست آوردن لغات سطوح مختلف و روابط بین آن‌ها با استفاده از الگوریتم CYK توانستیم لغات را در دنباله آموزش پیدا کرده و به همراه پنجره رویدادهای بعد از آن در یک جدول ذخیره کنیم. برای تشخیص زودهنگام حملات، تصمیم‌گیری برای هر لغت به‌منظور قرار گرفتن در دسته حمله یا دسته نرمال صورت گرفت. همان‌طور که در نمودارهای ROC مشاهده شد، نتایجی که لغات سطح دوم و سوم در تشخیص حملات داشتند نسبت به لغات سطح یک تفاوت و بهبود قابل توجهی داشت اما لغات سطح سوم نسبت به سطح دوم در به‌دست آوردن نتایج کارایی بهتری نداشته‌اند. علت این موضوع می‌تواند تنوع و گوناگونی بیشتر لغات در سطوح بالاتر باشد. این موضوع در زبان طبیعی نیز به چشم می‌خورد که هرچه سطح لغات بالاتر باشد، گوناگونی و پراکندگی در آن‌ها بیشتر خواهد بود. این موضوع می‌تواند در کارهای آینده مورد بررسی و تحقیق بیشتری قرار بگیرد.

۶- منابع

- [1] R. Kremmerer and G. Vigna, "Intrusion Detection: A Brief History and Overview," 2002.
- [2] D. Marc Dacier and A. wespi, "Towards a taxonomy of intrusion-detection systems," 1999.
- [3] A. Sperotto G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, "An Overview of IP Flow-Based Intrusion Detection," pp. 343-356, 2010.
- [4] W. Lee, S. J. Stolfo, "A Framework for Constructing Features and Models for Intrusion Detection Systems," *ACM Transaction on Information and System Security* 3, pp. 227-261, 2000.
- [5] X. Yan and J. Ying Zhang, "Early Detection of Cyber Security Threats using Structured Behavior Modeling," *ACM Transaction on Information and System Security*, vol. V, 2013.
- [6] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," *New York: s. n, conference on Applications, technologies, architectures, and protocols*, pp. 219-230, 2004.

Detection of advanced Cyber Attacks, Using Behavior Modeling Based on Natural Language Processing

K. Dadashtabar Ahmadi*, M. Kheirkhah, A. J. Rashidi

*Malek-Ashtar University of Technology

(Received: 01/12/2017, Accepted: 03/02/2018)

ABSTRACT

The complex and persistent attacks of network have been made up of numerous hidden stages. One of the reasons for the ineffectiveness of intrusion detection systems against these attacks is the use of a defense mechanism based on low-level network traffic analysis, in which the hidden relationships between alerts are not addressed. Our assumption is that there is a hidden structural information in traffic data, and we want to define rules in network traffic similar to linguistic rules and use it to describe the patterns of malicious network activity. In this way, the discovery of misuse and anomalous patterns can be well treated as the problem of learning syntactic structures and semantic fragments of the “network language”. In this paper, for the first time in cybersecurity, a new clustering is used named as the clustering of MD_DBSCAN; one of the most advanced types of DBSCAN clustering. In addition, a greedy algorithm inspired by the induction of grammar in natural language processing has been used to recognize high-level activities and define the relations between activities in different levels, by integrating low-level activities. In the recognition section of high-level activities of the proposed algorithm, for the first time, similarity edition criterion in hierarchical clustering has been added to the existing criteria in the base algorithm. According to ROC curves the results show that the accuracy of detection in higher-level activities are about 30% higher than low-level activities. Also by choosing the best setting for threshold parameters in attack detection algorithms, we had the highest F1 score in different levels from 1 to 3: 72.3 , 96.2, 96.4. which means that in general we have had the improvement of about 0.2 compared to the base algorithm.

Keywords: Behavior, Advanced Persistent Threats, Cyber Attacks, Data Fusion, Natural Language Processing

* Corresponding Author Email: Dadashtabar@mut.ac.ir