تشخیص نفوذ در شبکه‌های کامپیوتری مبتنی بر سیستم‌های فازی و الگوریتم جستجوی ممنوعه

مریم معین تقوی
مربی گروه مهندسی کامپیوتر ، دانشگاه آزاد اسلامی واحد تهران جنوب
Mmoeentaghavi@hotmail.com
مریم خادمی
استادیار گروه ریاضی کاربردی، دانشگاه آزاد اسلامی واحد تهران جنوب
khademi@azad.ac.ir

چکیده
با توجه به گسترش و توسعه سریع شبکه‌های کامپیوتری، نفوذ و حملات به آن‌ها افزایش یافته و به طرق و شیوه‌های مختلف انجام می‌شود. هدف از تشخیص نفوذ برای شناسایی استفاده غیرمجاز، سوء استفاده، و آسیب‌پذیری‌های ایجاد شده توسط کاربران داخلی و مهاجمان خارجی است. در این مقاله قصد داریم که سیستم تشخیص نفوذ از نوع سوء استفاده مبتنی بر سیستم فازی و الگوریتم جستجوی ممنوعه را ارائه کنیم. در ابتدا دانش موردنیاز خود را از سیستم فازی که مجموعه‌ای از قوانین if-then است، را کسب کرده و سپس الگوریتم جستجوی ممنوعه برای بهینه کردن مجموعه قوانین به‌دست آمده را بر روی مجموعه داده NSL-KDD پیاده و اجرا نمودیم. نتایج به‌دست آمده در مقایسه با نتایج موجود حاکی از آن است که روش پیشنهادی از صحت و کارایی مناسبی برخوردار است.

کلمات کلیدی: تشخیص نفوذ، سیستم فازی، الگوریتم جستجوی ممنوعه، Tabu Search

.

۱. مقدمه

همگام با رشد شبکه‌های کامپیوتری، حملات و نفوذها به این شبکه‌ها نیز گسترش یافته و به شکل‌های متعددی صورت می‌پذیرد. نفوذ مجموعه اقدامات غیرقانونی است که صحت، محرمانگی و یا دسترسی به یک منبع را به خطر می‌اندازد. نفوذگرها را می‌توان به دو دسته نفوذگرهای خارجی و داخلی دسته‌بندی کرد. نفوذگرهای خارجی کسانی هستند که اجازه استفاده از سیستم را ندارند، اما سعی می‌کنند که در سیستم راه یابند و نفوذگرهای داخلی کسانی هستند که برای دستیابی به سیستم اختیارات محدودی دارند، اما تلاش می‌کنند به منابعی که اجازه استفاده از آنها را ندارند، دسترسی پیدا کنند. در گذشته نفوذها و حملات بیش‌تر از ناحیه افرادی صورت می گرفت که علاقه‌مند بودند تا مهارت‌ها و توانایی‌های خود را آزمایش کنند، اما در حال حاضر، تمایل به نفوذ با انگیزه های مالی، سیاسی و نظامی بیش‌تر شده است. لذا ضرورت طراحی و ساخت سیستم‌های امن به مراتب بیش‌تر از گذشته احساس می‌شود.

به منظور مقابله با نفوذکنندگان به شبکه‌ها و سیستم‌های کامپیوتری، روش‌های متعددی تدوین شده است که روش تشخیص نفوذ نامیده می شوند. هدف از تشخیص نفوذ این است که استفاده غیرمجاز، سوء استفاده و آسیب رساندن به سیستم‌ها و شبکه‌های کامپیوتری توسط هر دو دسته کاربران داخلی و حمله‌کنندگان خارجی شناسایی شود. به‌طور کلی روش‌های تشخیص نفوذ به دو دسته اصلی تشخیص سوءاستفاده[1] و تشخیص رفتار غیرعادی[2] تقسیم می شوند. در روش تشخیص سوء استفاده از الگوهای نفوذ شناخته شده برای شناسائی نفوذها استفاده می‌گردد. اما در روش‌های تشخیص رفتار غیرعادی، رفتار عادی کاربران ملاک عمل قرار داده می‌شود و در نتیجه هر گونه رفتار مغایر با آن به عنوان تلاش جهت نفوذ به سیستم شناسائی می‌شود.

برای سیستم‌های تشخیص نفوذ روش‌های مختلفی مانند یادگیری ماشین با نظارت[3] و یادگیری بدون نظارت[4] وجود دارد که معروف‌ترین روش‌های با نظارت ، دسته‌بندی است که مهم‌ترین الگوریتم‌های آن Naïve Bayes، C4.5، Random Forest, SVM و و k_Nearest Neighbor (Kruczkowski, 2014) و معروف ترین روش‌های بدون نظارت خوشه بندی[5] است که مهم‌ترین الگوریتم‌های آن عبارتست از : C-means، Y-means، EM، SOM (Liu, 2011). هم‌چنین شبکه‌های عصبی ، فازی منطقی ، الگو ریتمهای ژنتیک درخت تصمیم[6] نیز می‌توانند حملات را شناسایی کنند.

سیستم‌های فازی مبتنی بر قوانین فازی if-then در سیستم‌های تشخیص نفوذ با مؤفقیت بیشتری مورد استفاده قرار می‌گیرند. در این پژوهش، مجموعه‌ای از قوانین if-then فازی و نتایج آن نیز جمع‌آوری شده که نشان می‌دهد آیا داده نرمال و یا غیرنرمال است. Gloverدر سال ۱۹۸۹-۱۹۹۰ در (Glover, 1997) الگوریتم جستجوی ممنوعه (Tabu Search) را به عنوان روشی برای گریز از گرفتار شدن در نقاط بهینه محلی در جستجوها ارائه داد که هدف آن، به‌دست آوردن لیست مسیرهای ممنوعه در همسایگی یک مسیر برای دوری جستن از چرخیدن در حلقه بین مسیرها است.

در این تحقیق، سیستم تشخیص نفوذ مبتنی بر سیستم فازی و الگوریتم جستجوی ممنوعه که براساس روش تشخیص سوءاستفاده است را بر مجموعه داده‌های NSL-KDD (Habibi, 2015) پیاده‌سازی نمودیم. استفاده از الگوریتم جستجوی ممنوعه در سیستم فازی ، تلاش می‌کند که به‌طور مؤثر کشف و بهره‌برداری از فضای جستجو بزرگ را در ارتباط با مشکلات تشخیص نفوذ داشته باشد.

۲. روش پیشنهادی

در این بخش روشی براساس قوانین فازی if-then و الگوریتم جستجوی ممنوعه ارائه شده است. این روش شامل مراحل اصلی پیش‌پردازش، ارزیابی و دسته‌بندی است.

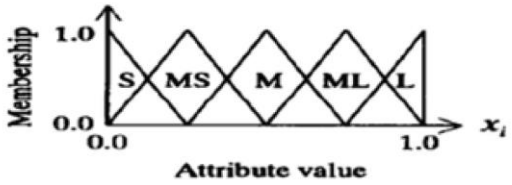
۲-۱. پیش‌پردازش

[1] Misuse Detection
[2] Abnormal Detection
[3] Supervised Learning
[4] Unsupervised Learning
[5] Clustering
[6] Decision tree

در ابتدا داده‌های جمع‌آوری شـده را مورد پیش‌پردازش قرار گرفته و داده‌های غیرمعتبر و اضـافی را از مجموعه داده‌ها حذف و یا با مقادیر مناسبی جایگزین کردیم. سپس با استفاده از روش‌های نرمال‌سازی داده‌ها را نرمال نمودیم.

۲-۲. ارزیابی

در مرحله فازی‌سازی، داده‌ها به قالبی قابل استفاده برای واکشی دانش تبدیل می‌شوند. در این مرحله برای هر متغیر، یک مجموعه فازی، براساس یکی از روش‌های فازی‌سازی داده‌ها ایجاد خواهد شد. فازی‌سازی داده‌ها بر مبنای فازی‌سازی پنج زبانی به ترتیب از اعداد ۱ تا ۶ برای نشان دادن مقادیر S[۷] MS[۸] ، M[۹] ، ML[۱۰] ، L[۱۱] و DC[۱۲] استفاده می‌شود (شکل ۱).



شکل ۱. مجموعه‌های فازی با پنج زبانی

در این مطالعه، از روش فازی‌سازی تاکاگی سوگنو کانگ (TSK) استفاده شده است. برای ایجاد مجموعه قوانین اولیه، به هر مقدار زبانی مجموعه‌های فازی جزء اول هر قانون، یکی از نمادهایی که در بالا تعریف شـده‌اند را به صـورت تصـادفی نسـبت می‌دهیم. از آن‌جایی که مجموعه داده‌های استفاده شده دارای ۴۱ خصیصه است برای محاسبه درجه سازگاری هر رکورد از داده‌ها با قانون if-then فازی توسط حاصل ضرب تابع عضویت هر خصیصه استفاده کرده و در نهایت درجه سازگاری برای هر دسته از مجموع درجه سازگاری هر رکورد متعلق به این دسته محاسبه می‌گردد. پس از محاسبه درجه سازگاری هر دسته، دسته‌هایی که بیش‌ترین درجه سازگاری را داشته باشد انتخاب می‌شود و قانونی که منجر به ایجاد این دسته شده است به عنوان قانون بهینه ذخیره می‌شود.

پس از این که مجموعه قوانین اولیه ایجاد شدند به کمک روال زیر دسته نتیجه و درجه قطعیت هر یک از قوانین مشخص می‌شوند: محاسـبه درجه سـازگاری هر نمونه آموزشـی $x_p = (x_{p1}, x_{p2}, ..., x_{pn})$ با قانون if-then فازی R_j که به کمک عمل حاصل‌ضرب به‌دست می‌آید

$$\mu_{R_j}(x_p) = \mu_{A_{j1}}(x_{p1}) \times ... \times \mu_{A_{jn}}(x_{pn}), \quad p = 1, 2, ..., m \qquad (1)$$

6- small
7- medium small
8- medium
9- medium large
10-large
11- don't care

که $\mu_{A_{ji}}(.)$ تابع عضویت A_{ji}، p تعداد نمونه‌های آموزشی و n تعداد صفات هر نمونه است.

محاسبه مجموع درجه‌های سازگاری برای هر دسته:

$$\beta_{Class\ h}(R_j) = \sum_{x_p \in Class\ h} \mu_{R_j}(x_p) \quad ,h = 1,2,...,c \qquad (2)$$

که در اینجا c تعداد دسته‌ها است.

یافتن دسته نتیجه C_j که بیش‌ترین مقدار $\beta_{Class\ h}(R_j)$ را در بین c دارد.

$$\beta_{Class\ C_j}(R_j) = \max\{\beta_{Class\ 1}(R_j),...,\beta_{Class\ c}(R_j)\}. \qquad (3)$$

و اگر در بیش از یک دسته دارای بیش‌ترین مقدار باشد آنگاه دسته نتیجه به صورت یکتا مشخص نمی شود و در این حالت یک مقدار تهی را به عنوان دسته نتیجه قانون در نظر می‌گیریم.

هنگامی که دسته نتیجه C_j به کمک رابطه (3) تعیین شد، درجه قطعیت و $\overline{\beta}$ از روابط زیر به‌دست می‌آیند:

$$CF_j = \frac{\beta_{Class\ C_j}(R_j) - \overline{\beta}}{\sum_{h=1}^{c} \beta_{Class\ h}(R_j)} \qquad \overline{\beta} = \frac{\sum_{h \neq C_j} \beta_{Class\ h}(R_j)}{c-1} \qquad (4)$$

به کمک روال فوق برای هر قانون دسته نتیجه و درجه قطعیت هر قانون را مشخص می‌شود. برای یک الگوی جدید $x_p = (x_{p1},...,x_{pn})$ قانون برنده R_{j^*} به کمک رابطه زیر به‌دست می‌آید:

$$\mu_{j^*}(x_p).CF_{j^*} = \max\{\mu_j(x_p).CF_j : j = 1,2,...,N\} \qquad (5)$$

در ابتدا بهترین قوانین به همراه پارامترهای هر یک بررسی می شود. سپس با انتخاب یکی از قوانین به صورت تصادفی قسمت if قانون را تغییر می‌دهد. پارامترهای قانون جدید ایجاد شده (در صورتی که قبلاً در لیست ممنوعه وارد نشده بوده باشد) را محاسبه می‌کند و در صورتی که جواب حاصل بهتر از قانون فعلی بود آن را ذخیره کرده و به لیست قانون‌ها اضافه می‌نماید. هنگام افزوده شدن قانون جدید از آن‌جایی که تعداد لیست بهترین قوانین ثابت است قانونی که در لیست بدترین نتیجه را داشت از لیست بهترین قوانین حذف شده و به لیست ممنوعه افزوده می شود. و در نهایت آخرین قوانین بهینه را واکشی کرده و مجموعه قوانین اگر-آنگاه فازی باید دارای درجه قطعیت بالایی باشد. سپس از تابع هزینه زیر برای ارزیابی مجموعه قوانین استفاده می‌کنیم:

$$Cost(S) = m - \sum_{j=1}^{N} NCP(R_j) \qquad (6)$$

که m تعداد کل نمونه‌های مجموعه آموزشی است و $\sum_{j=1}^{N} NCP(R_j)$ تعداد نمونه‌هایی است که به درستی توسط مجموعه قوانین S دسته‌بندی شده‌اند و N تعداد قوانین اگر-آنگاه فازی موجود در مجموعه قوانین S است.

الگوریتم جستجوی ممنوعه مبتنی بر سیستم تشخیص نفوذ در شکل (۲) نشان داده شده است. شامل مراحل زیر است :۱- بخش ایجاد مجموعه اولیه قوانین اگر-آنگاه فازی و مقداردهی اندازه لیست ممنوعه (مقداردهی اولیه) ۲- بخش ارزیابی مجموعه قوانین اگر-آنگاه فازی فعلی به کمک تابع هزینه (ارزیابی) ۳- بخش اصلاح مجموعه قوانین جدید از مجموعه قوانین فعلی توسط اصلاح یکی از قوانین (اصلاح) ۴- بخش پذیرش قوانین جدید در صورتی هزینه آن از هزینه قوانین فعلی بیش‌تر باشد یا قانون اصلاح شده در لیست ممنوعه نبا شد (پذیرش) ۵- بخش خاتمه الگوریتم در صورت برآورده شدن شرط خاتمه. در غیراین صورت به بخش ۲ برمی‌گردیم (خاتمه).

Algorithm Tabu Search based IDS

// $S_{init}, S_{best}, S_{current}, S_{new}$ are the initial, best, current, and new set of fuzzy if - then rules.

// R_{new} is the new rule.

// k is the total number of iterations.

Begin

(1) $S_{current} = S_{init};$

(2) $S_{best} = S_{init};$

while ($i <= k$)

(3) $S_{new} = \text{modify}(S_{current});$

(4) if $Cost(S_{new}) < Cost(S_{best})$ then

(5) $S_{best} = S_{new}; Cost(S_{best}) = Cost(S_{new});$

(6) if $Cost(S_{new}) < Cost(S_{current})$ then

(7) $S_{current} = S_{new}; Cost(S_{current}) = Cost(S_{new})$

(8) elseif $R_{new} \notin TL$ then

(9) $S_{current} = S_{new}; Cost(S_{current}) = Cost(S_{new});$

(10) $TL = TL + R_{new};$

(11) $i = i + 1;$

End while

End.

شکل ۲ . الگوریتم پیشنهادی

۳. مجموعه داده

در سال ۱۹۹۸ برنامه‌ای به منظور ارزیابی تشخیص نفوذ تحت عنوان DARPA1998 توسط آزمایشگاه ام آی تی لینکلن ایجاد شد که هدف آن بررسی و ارزیابی تحقیقی در مورد تشخیص نفوذ بود. بدین‌منظور یک مجموعه از داده استاندارد ایجاد شد که

حملات مختلف شبیه سازی شده در محیط شبکه واقعی را در برداشت. در مسابقه تشخیص نفوذ KDD99 ، از این مجموعه داده استفاده کرد. در این تحقیق از مجموعه داده NSL-KDD استفاده شده که ویرایش جدید KDD99 است. این مجموعه داده به عنوان یک داده‌ی استاندارد برای ارزیابی سیستم‌های تشخیص نفوذ پذیرفته شده و مورد استفاده قرارگرفته است. این مجموعه داده در برگیرنده هر دو داده آموزشی و تستی است. در این تحقیق از کل داده آموزشی، استفاده شده است. هر رکورد در این داده یا یک رکورد نرمال است و یا به یکی از ۲۲ دسته مختلف از حملات تعلق دارد. تمامی این حملات به ۴ دسته اصلی تقسیم می‌گردند که عبارتند از: DoS، R2L، U2R و Probing .

مجموعه داده مورد آزمایش در این تحقیق، مجموعه داده NSL_KDD است. این مجموعه داده برای حذف برخی مشکلات ذاتی مجموعه داده KDD99 ، گرچه مجموعه داده جدید نیز از مشکلات مجموعه داده قبلی بی‌اثر نبوده است و نمی‌تواند به‌طور کامل بیان‌کننده شبکه واقعی باشد، اما هنوز بهترین داده برای ارزیابی روش‌های مختلف تشخیص نفوذ است. تعداد نمونه‌های آموزشی و تستی در مجموعه داده‌ی NSL_KDD معقول‌تر از مجموعه داده‌های KDD99 است.

مزایا NSL_KDD نسبت به: KDD99

• مجموعه داده آموزشی شامل افزونگی رکورد نمی‌باشد. بنابراین طبقه‌بندی به سمت نمونه‌هایی با تعداد تکرار بیش‌تر بایاس نخواهد شد.

• نمونه‌های تکراری در مجموعه داده آزمایشی وجود ندارد بنابراین، کارایی یادگیرنده‌ها به‌وسیله‌ی روش‌هایی که نرخ شناسایی بیش‌تری روی نمونه‌های پرتکرار دارند بایاس نخواهد شد.

•تعداد نمونه‌های انتخاب شده از هر گروه سختی به‌طور معکوس با درصد رکوردها در دیتا ست KDD اصلی متناسب است. در نتیجه نرخ طبقه‌بندی روش‌های یادگیری ماشین بیش از پیش متفاوت خواهد بود،که این امر موجب ارزیابی صحیح‌تری از تکنیک‌های متفاوت یادگیری می‌شود.

•تعداد نمونه‌ها در داده‌های آموزشی و آزمایشی معقول است، و اجرای آزمایشات روی مجموعه کامل بدون نیاز به انتخاب تصادفی بخش کوچکی از داده‌ها را فراهم می‌کند. بنابراین ارزیابی نتایج کارهای تحقیقاتی متفاوت قابل مقایسه و ارزیابی خواهد بود.

جدول1 . تعداد رکوردهای موجود در مجموعه های داده

مجموعه داده	تعداد رکورد آموزشی	تعداد رکورد آزمایشی
KDD99	۴۹۴۰۲۱	۳۱۱۰۲۷
NSL-KDD	۱۲۵۹۷۳	۲۲۵۴۴

در جدول (۲) خصوصیات مجموعه داده NSL_KDD را مشاهده می کنید .

جدول2 . جدول اسامی خصیصه‌ها NSL_KDD

(نوع خصیصه‌ها با حروف discrete (گسسته) و continuous (پیوسته) نشان داده شده است)

Feature name		Type	Description
1	Duration	continuous	length (number of seconds) of the connection
2	protocol_type	discrete	type of the protocol, e.g., tcp, udp, etc.
3	Service	discrete	network service on the destination, e.g., http, telnet, etc.
4	src_bytes	continuous	number of data bytes from source to destination
5	dst_bytes	continuous	number of data bytes from destination to source
6	Flag	discrete	normal or error status of the connection
7	Land	discrete	1 if connection is from/to the same host/port; 0 otherwise
8	wrong_fragment	continuous	number of ‘‘wrong’’ fragments
9	Urgent	continuous	number of urgent packets
10	Hot	continuous	number of ‘‘hot’’ indicators
11	num_failed_logins	continuous	number of failed login attempts
12	logged_in	discrete	1 if successfully logged in; 0 otherwise
13	num_compromised	continuous	number of ‘‘compromised’’ conditions
14	root_shell	discrete	1 if root shell is obtained; 0 otherwise
15	su_attempted	discrete	1 if ‘‘su root’’ command attempted; 0 otherwise
16	num_root	continuous	number of ‘‘root’’ accesses
17	num_file_creations	continuous	number of file creation operations
18	num_shells	continuous	number of shell prompts
19	num_access_files	continuous	number of operations on access control files
20	num_outbound_cmds	continuous	number of outbound commands in an ftp session
21	is_hot_login	discrete	1 if the login belongs to the ‘‘hot’’ list; 0 otherwise
22	is_guest_login		1 if the login is a ‘‘guest’’login; 0 otherwise
23	Count	continuous	number of connections to the same host as the current connection in the past two seconds
24	serror_rate	continuous	% of connections that have ‘‘SYN’’ errors

25	rerror_rate	continuous	% of connections that have ‘‘REJ’’ errors
26	same_srv_rate	continuous	% of connections to the same service
27	diff_srv_rate	continuous	% of connections to different services
28	srv_count	continuous	number of connections to the same service as the current connection in the past two seconds
29	srv_serror_rate	continuous	% of connections that have ‘‘SYN’’ errors
30	srv_rerror_rate	continuous	% of connections that have ‘‘REJ’’ errors
31	srv_diff_host_rate	continuous	% of connections to different hosts
32	dst_host_count	continuous	count for destination host
33	dst_host_srv_count	continuous	srv_count for destination host
34	dst_host_same_srv_rate	continuous	same_srv_rate for destination host
35	dst_host_diff_srv_rate	continuous	diff_srv_rate for destination host
36	dst_host_same_src_port_rate	continuous	same_src_port_rate for destination host
37	dst_host_diff_host_rate	continuous	diff_host_rate for destination host
38	dst_host_serror_rate	continuous	serror_rate for destination host
39	dst_host_srv_serror_rate	continuous	srv_serror_rate for destination host
40	dst_host_rerror_rate	continuous	rerror_rate for destination host
41	dst_host_srv_rerror_rate	continuous	srv_serror_rate for destination hos

۴. نتایج تجربی

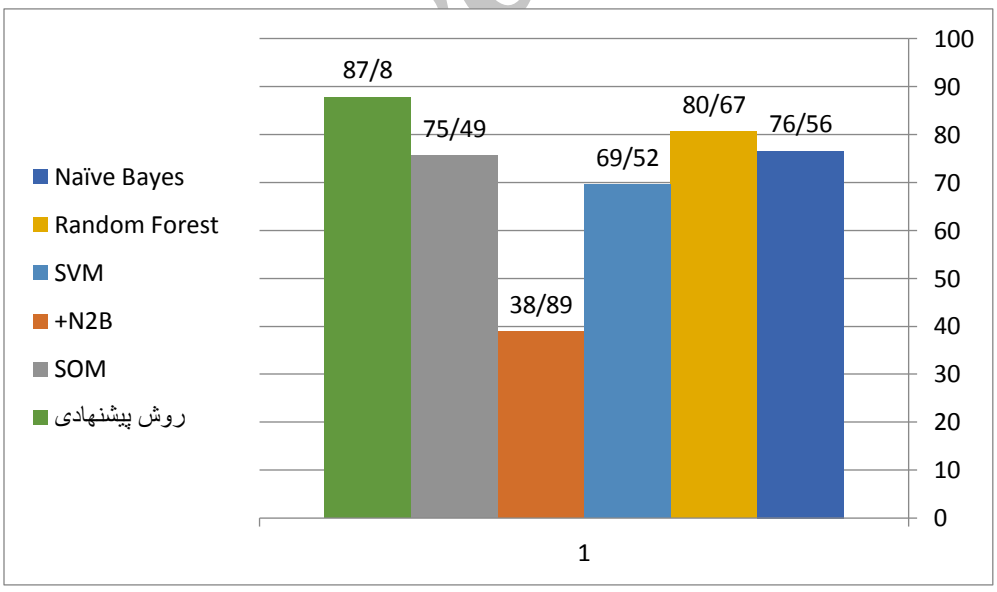
روش پیشنهادی بر روی مجموعه داده NSL_KDD اجرا شده است(Habibi, 2015). در اینجا از مجموعه‌ای با تعداد 125973 نمونه از تمام دسته‌ها به عنوان مجموعه داده آموزشی و 22544 نمونه به عنوان مجموعه داده تستی استفاده شده است که این رقم برابر با تعداد نمونه‌های موجود در فایل تست مجموعه داده NSL-KDD است.

میانگین دقت در روش پیشنهادی برای جداسازی نمونه‌های نرمال از غیرنرمال بین ۸۷ تا ۹۱ درصد است (دقت شود این صحت در حالت آموزش دو کلاسه برای روش پیشنهادی است یعنی حالتی که فقط به روش پیشنهادی حمله بودن یا نبودن را آموزش دهیم). مجموعه قوانین دربازه ۳۰ تا ۸۰ قانون متفاوت است و بهترین نتایج از تعداد قوانین بین ۴۰ تا ۵۰ حاصل می‌شود که در نمونه تست بیان شده ۴۴ قانون برای دسته‌بندی استفاده شده که صحت ۸۷٪ را در پی داشته است و در حالت کلی میزان تشخیص نفوذ در این سیستم برابر ۸۷ درصد نفوذ صورت گرفته است. در جدول (۳) نیز میزان دقت تشخیص نفوذ روش پیشنهادی با روش‌هایی دیگر نشان داده شده است .

جدول۳ . مقایسه میزان تشخیص نفوذ به‌کار گرفته شده در NSL-KDD دیتاست

روش‌های مورد مقایسه	میزان تشخیص نفوذ
Naïve Bayes	۵۶/۷۶
Random Forest	۶۷/۸۰
SVM	۵۲/۶۹
Multinomial Naïve Bayes (+ N2B)[46]	۸۹/۳۸
SOM[44]	۴۹/۷۵
روش پیشنهادی	۸۰/۸۷

در شکل (۳) نیز مقایسه میزان نرخ تشخیص نفوذ روش پیشنهادی را با سایر روش‌های موجود آورده شده است.



شکل۳. نمودار مقایسه روش پیشنهادی با سایر روش ها

با توجه به جدول(۳) و هم‌چنین شکل (۳) روش پیشنهادی دارای بالاترین دقت تشخیص نفوذ نسبت به روش‌های کلاسیک و چند روش جدید است. این امر به این دلیل است که روش پیشنهادی با مقداردهی اولیه مناسب، تابع ایجاد مفید و تابع ارزیابی دقیق به

خوبی در فضای حالت مسأله از راه حل‌های بهینه محلی گریخته و به سمت راه حل بهینه سراسری حرکت می‌نماید و به این صورت در شناسایی حالات نرمال یا حمله بودن مؤفق‌تر عمل می‌کند.

۵ . نتیجه‌گیری

در این مقاله روشی برای تشخیص نفوذ در شبکه‌های کامپیوتری مطرح شده است. دانش موردنظر را توسط فرایند داده‌کاوی و به کمک روش دسته‌بندی به‌دست آوردیم. مجموعه قوانین اولیه توسط قوانین if-then فازی نمایش داده شده است. ارزیابی مجموعه قوانین به‌دست آمده توسط معیارهای دقت سیستم دسته‌بندی مبتنی بر قوانین و هم‌چنین بالا بودن قابلیت تفسیر چه از نظر کم بودن تعداد قوانین و چه از نظر کوتاه بودن قوانین حاصله، انجام شده است. سپس به کمک الگوریتم TS، مجموعه قوانین فازی به‌دست آمده، بهبود یافت و در نهایت مجموعه قوانین بهینه حاصل شد. برای بررسی و ارزیابی مورد مطالعاتی کشف نفوذ، روش پیشنهادی بر روی مجموعه داده NSL-KDD اعمال شده است. مجموعه داده آموزشی مورد استفاده دارای ۱۲۵۹۷۳ نمونه از چهار نوع مختلف حمله و یک نمونه نرمال بود و تعداد قوانین موجود در مجموعه قوانین بین ۴۰ تا ۸۰ متغیر بود که بهترین حالت تعداد ۶۸ بود. سپس مجموعه قوانین برای دسته‌بندی نمونه‌های آزمایشی مورد استفاده قرار داده شد. مجموعه آزمایشی شامل ۲۲۵۴۴ نمونه بود و برای ارزیابی نتایج به‌دست آمده برای مجموعه داده‌های آزمایشی از پارامتر تشخیص نفوذ استفاده شد. پس از آن روش پیشنهادی را با روش‌های NB، Random Forest، SVM وSOM که بر روی مجموع داده NSL-KDD تست شده‌اند و روش MNB(+N2B) مقایسه کردیم و ارزیابی و مقایسه نتایج نشان داد که روش پیشنهادی دارای بهترین میزان دقت در تشخیص انواع حملات در بین روش‌های موجود است.

۶ . مراجع

Angurala, M. G. (2015). Different Attacks in the Network: A Review. *International Journal of Advanced Engineering, Management and Science (IJAEMS)*, 1-3.

Bahamida, B. B. ((2014)). Intrusion Detection Using Fuzzy Meta-Heuristic Approaches. *International Journal of Applied Metaheuristic Computing (IJAMC),*, 39-53.

Chattemvelli, R. S. (2012). GA Approach for Network Intrusion Detection. *International Journal of Research and Reviews in Information Sciences (IJRRIS)*.

Elkan, C. (2000). Results of the KDD'99 classifier learning. *ACM SIGKDD Explorations Newsletter*, 63-64.

Glover, F. L. (1997). *Tabu Search.* Norwell, MA, USA: Kluwer Academic.

Habibi, A. (2015). *University of New Brunswick*. Retrieved from The NSL-KDD Data Set: http://nsl.cs.unb.ca/NSL-KDD/

Ibrahim, L. M. (2013). A comparison study for intrusion database (Kdd99, Nsl-Kdd) based on self-organization map (SOM) artificial neural network,. *Journal of Engineering Science and Technology*, 107-119.

Kruczkowski, M. N.-S. (2014). Comparative Study of Supervised Learning Methods for Malware Analysis. *JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY*, 24-33.

Liu, Y. W. (2011). On using tabu search for fuzzy clustering analysis. *International Journal of Applied Metaheuristic Computing (IJAMC), 6*, 6821-6828.

Naik, D. S. (2014). A Review on Image Segmentation Clustering. *International Journal of Computer Science and Information Technologies(IJCSIT)*, 3289 - 3293.

Tsang, C. H. (2005). Anomaly intrusion detection using multi-objective genetic fuzzy system and agent-based evolutionary computation framework. *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM'05)*.

Zhang, Y. W. (2012). Tabu Search Particle Swarm Optimization used in Cluster Analysis. *Journal of Science, 1*, 6-12.

Intrusion Detection in computer networks based Fuzzy systems and Tabu search

Maryam Moeen Taghavi
Instructor of Department of Computer Engineering, Islamic Azad University South Tehran Branch,Iran, Mmoeentaghavi@hotmail.com
Maryam Khademi
Assistant Professor of Department of Applied Mathematics, Islamic Azad University South Tehran Branch,Iran, khademi@azad.ac.ir

Abstract. Due to the rapid development of computer networks, Intrusions and attacks into these networks have grown, and occur in various ways. To resist against hackers and intrusive behaviors, several algorithms have been introduced in literature known as intrusion detection methods. The aim of intrusion detection is to identify unauthorized use, misuse, and vulnerability made by internal users or external attackers. The proposed method, based on misuse detection, extracts required knowledge from fuzzy system which is a set of fuzzy if-then rules, and performs the intrusion detection process. . In fact, the mentioned knowledge is considered as a fuzzy rule base which is optimized during the data mining process by an optimization algorithm according to some criteria such as accuracy and comprehensibility. Tabu search algorithm is employed to optimize the obtained set of fuzzy rules. Finally, the proposed method is implemented and applied to the NSL-KDD dataset which contains some information about normal and intrusive behaviors in computer networks. The results are compared with those of well-known methods, and show the competitive accuracy and efficiency.

Keywords: Tabu search, Fuzzy rule extraction, Intrusion detection, Optimization