به نام خدا

گزارش پروژه گروه اول

تشخیص حملات DDos به کمک اطلاعات موجود در packetهای شبکه

هادی فاضلی نیا – مهدی نادری

---

۱. مقدمه

پروژه مورد نظر در مورد تشخیص حملات DDos به کمک اطلاعات دریافتی از هر packet می‌باشد تا به این واسطه قادر به دفع این نوع حملات از دستگاه‌های حساس باشیم.

با توجه به این که پیش‌گیری از چنین حملاتی می‌تواند قابل توجه بسیاری از شرکت‌ها و سازمان‌ها باشد و ممکن است به عنوان یک چالش به این موضوع نگاه کنند، به نظر می‌رسد این موضوع یک موضوع کاربردی و مفید می‌باشد که می‌تواند در صورت پیاده‌سازی به شکل صحیح و قابل توجیه، قابل ارائه به بسیاری از افراد نیز باشد، از این رو با توجه به علاقه‌ای که به این زمینه داشتیم و همچنین کارایی آن، تصمیم گرفتیم یک مسئله داده کاوی در این زمینه را بررسی کنیم.

در این زمینه کارهای دیگری نیز در قالب مقالاتی به انتشار رسیده‌اند که اکثرا هم دقت‌های بسیار خوبی در حد حتی ۱۰۰ درصد ارائه می‌دهند. پس ما نیز انتظار داریم که پس از انجام کامل مراحل این پروژه، دقتی خوب در حد سایر کارهای مشابه داشته باشیم.

۲. دیتاست مورد استفاده و کارهای مشابه

در جهت پیش‌برد کار خود تصمیم به استفاده از دو دیتاست مختلف گرفتیم که تشابه هر دو در این مورد است که حاوی اطلاعاتی از پکت‌های مختلفی هستند که یا مربوط به یک حمله‌ی DDos هستند و یا به اصطلاح سالم هستند و مربوط به حمله DDos نیستند. در زیر، لینک دسترسی به صفحه هر کدام از دو دیتاست آورده شده است:

Dataset 1: DDoS Botnet Attack on IOT Devices

Dataset 2: CSE-CIC-IDS2018

قابل توجه است که دیتاست دوم شامل چندین دیتاست است که حاوی اطلاعات پکت‌های شبکه در طی روزهای جداگانه می‌باشد و با توجه به محدویتی که در منابع وجود داشت، تنها از دیتاست مربوط به تاریخ 2018-02-03 گرفتیم. (لینک دانلود)

همچنین قابل توجه است که روی دیتاست اول، کار مشابهی صورت گرفته و با استفاده از تکنیک‌های داده‌کاوی نتایج مناسبی به دست آمده است که این کار در قالب یک مقاله نیز منتشر شده است. لینک دسترسی به مقاله مورد نظر در زیر آمده است:

Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset

۳. تشریح گام‌های موضوع پروژه در قالب فرآیند CRISP-DM

۳.۱- درک تجاری مسئله: همانطور که در ابتدا نیز ذکر شد، بازار هدف این موضوع می‌تواند شرکت‌ها و سازمان‌هایی باشند که در معرض چنین حملاتی می‌توانند قرار بگیرند و در صورتی که درک درستی از خواسته آن‌ها وجود داشته باشد و در صورت پیاده سازی پروژه در جهت خواسته‌های آن‌ها، این پروژه می‌تواند مسیر درستی را تا به دست آوردن نتایج مناسب طی کند.

۳.۲- درک مناسب از داده مورد نیاز: بر اساس آنچه از نتایج مرحله اول به دست می‌آید یک دید کلی نسبت به آنچه که باید انجام شود، به دست خواهد آمد و اگر مسئله به درستی درک شده باشد قادر خواهیم بود تا دیتاست را مطابق با آنچه که نیاز است جمع آوری کنیم، نه کمتر و نه بیشتر از حد نیاز.

۳.۳- آماده سازی داده‌ها: در این مرحله با توجه به درکی که از ابعاد کلی داده مورد نیاز در مرحله پیش به دست آورده‌ایم، باید دیتای مناسب را به شکل عملی جمع‌آوری کنیم. در این مورد دو دیتاست را در ابتدای کار معرفی کردیم که طبق جستجوهای ما، در حال حاضر از بهترین دیتاست‌هایی هستند که برای مسئله‌ی مورد نظر وجود دارند و هر دو دیتاست با توجه به نیازهای موجود، جمع‌آوری شده‌اند که با در نظر داشتن این مورد، انتظار می‌رود نتایج بسیار خوبی نیز در پایان کار ارائه دهند.

۳.۴- پیاده‌سازی مدل: همانطور که می‌دانیم هر مسئله، خواستار نوع خاصی از مدل‌ها خواهد بود. از طرفی مسئله‌ی ما یک مسئله classification است. پس به ابتدا به دنبال مدل‌هایی نسبتا ساده خواهیم رفت که این کار را برای انجام دهند و همینطور از تفسیرپذیری بالایی نیز برخوردار باشند تا پس از ایجاد مدل و آموزش دادن آن، مدل قابل تحلیل نیز باشد. آنچه که مد نظر ما هست استفاده از دو مدل Decision Tree و Random Forest است که اولی تفسیرپذیری بالایی دارد و دومی نیز مزیت‌هایی مانند انعطاف پذیری مدل و سادگی در استفاده را ارائه می‌دهد. همچنین مدل Random Forest از آن جهت برای کار ما مناسب است که قادر است با وجود دیتاست‌هایی با تعداد ابعاد بسیار زیاد نیز کار خود را به درستی انجام دهد و از این لحاظ دچار مشکل نشود چیزی که ممکن است در یک شبکه عصبی، مقداری چالش برانگیز باشد و نیاز به صرف زمان و هزینه بیشتری نیز خواهد داشت.

۳.۵- ارزیابی مدل: در این مرحله لازم است تا مدل یا مدل‌هایی که در مرحله قبل روی داده موجود پیاده شده‌اند ارزیابی شوند تا در صورت نیاز بهبود یابند و یا مدل‌های جایگزین آن‌ها را استفاده کنیم. همانطور که در ادامه خواهیم دید، با توجه به دیتاست‌های بسیار مناسبی که در اختیار داریم، هر دو مدل درخت تصمیم و Random

Forest روی هر دو دیتاست ما بسیار عالی عمل می‌کنند و نیازی به صرف زمان و هزینه برای بهبود آن‌ها دیده نمی‌شود.

۳.۶- گسترش و پشتیبانی: این مرحله نیز شامل این می‌شود که کار انجام شده در بازار هدف ارائه شود و در صورت استقبال از آن و خریده شدن آن توسط سازمان‌ها، لازم است تا همیشه پشتیبانی از آن نیز به عمل آید تا صورت مشاهده هر گونه خطایی در سیستم ارائه شده، در جهت رفع آن تلاش کنیم.

۴. مروری کلی بر گام‌های عملی پروژه

آنچه در ادامه گزارش می‌آید مروری کلی بر گام‌های عملی پروژه است تا با آنچه در این مسیر استفاده شده است آشنا باشیم. آنچه که در این مسیر برای هر دو دیتاست موجود استفاده شده است تقریبا مشابه است و در صورت وجود تفاوت در نحوه برخورد با یک دیتاست، به آن اشاره خواهیم کرد.

۴.۱- آمار کلی داده‌های موجود

همانطور که قبلا ذکر شد با دو داده حجیم سر و کار داریم که آمار کلی هر یک در زیر آمده است:

Dataset 1: 47 columns , 1927101 rows

Dataset 2: 80 columns , 1048575 rows

در زیر نیز دیتاست اول از این نظر که هر ستون آن نماینده چه اطلاعاتی از پکت است بررسی می‌شود:

Feature	Description
pkSeqID	Row Identifier
Stime	Record start time
flgs	Flow state flags seen in transactions
flgs_number	Numerical representation of feature flags
Proto	Textual representation of transaction protocols present in
proto_number	network flow
saddr	Numerical representation of feature proto
sport	Source IP address
daddr	Source port number
dport	Destination IP address
pkts	Destination port number
bytes	Total count of packets in transaction
state	Totan number of bytes in transaction
state_number	Transaction state

ltime	Numerical representation of feature state
seq	Record last time
dur	Argus sequence number
mean	Record total duration
stddev	Average duration of aggregated records
sum	Standard deviation of aggregated records
min	Total duration of aggregated records
max	Minimum duration of aggregated records
spkts	Maximum duration of aggregated records
dpkts	Source-to-destination packet count
sbytes	Destination-to-source packet count
dbytes	Source-to-destination byte count
rate	Destination-to-source byte count
srate	Total packets per second in transaction
drate	Source-to-destination packets per second
attack	Destination-to-source packets per second
category	Class label: 0 for Normal traffic, 1 for Attack Traffic
subcategory	Traffic category
	Traffic subcategory

تعدادی از ستون‌ها نیز به شکل دستی به کمک ستون‌های موجود تولید شده‌اند تا مسئله پس از حل شدن، از دقت بیشتری برخوردار باشد که ستون‌های مورد بحث در زیر معرفی شده‌اند:

Feature	Description
TnBPSrcIP	Total Number of bytes per source IP
TnBPDstIP	Total Number of bytes per Destination IP
TnP_PSrcIP	Total Number of packets per source IP
TnP_PdstIP	Total Number of packets per Destination IP
TnP_PerProto	Total Number of packets per protocol
TnP_Per_Dport	Total Number of packets per dport
AR_P_Proto_P_SrcIP	Average rate per protocol per Source IP
AR_P_Proto_P_SrcIP	Average rate per protocol per Destination IP
N_IN_Conn_P_SrcIP	Number of inbound connections per source IP
N_IN_Conn_P_DstIP	Number of inbound connections per destination IP
AR_P_Proto_P_Sport	Average rate per protocol per spor

AR_P_Proto_P_Dport	Average rate per protocol per dport
Pkts_P_State_P_Protocol_P_DestIP	Number of packets grouped by state of flows and protocols per destination IP
Pkts_P_State_P_Protocol_P_SrcIP	Number of packets grouped by state of flows and protocols per source IP

همچنین یک مورد قابل توجه این است که در هیچ یک از دو دیتاست مقدار NULL وجود ندارد و تنها در دیتاست دوم بعضی سطرها شامل مقدار بی‌نهایت هستند که نحوه برخورد با آن‌ها در ادامه خواهد آمد.

۴.۲- انجام فرآیندهای اکتشافی داده‌ها (EDA)

ابتدای کار برای اینکه ستون‌های بدون کاربرد را از کار کنار بگذاریم یک نگاه اجمالی به ستون‌ها شده است و در نتیجه ستون‌هایی که صرفا نمایانگر id و یا شماره سطر بوده‌اند از داده کنار رفته‌اند.

سپس با نوع دوم ستون‌ها مواجه هستیم که ستون‌هایی هستند که شاید به نظر بیاید اطلاعات مفیدی را شامل می‌شوند اما در تعریف مسئله ما و در تشخیص یک پکت مربوط به حمله DDos، هیچ مورد سودمندی را در اختیار نخواهند گذاشت، به عنوان مثال ستون‌هایی مانند saddr و daddr که تنها بیان می‌کنند پکت از چه آدرسی و به سمت چه آدرسی ارسال شده است، ستون‌هایی بدون فایده هستند. پس هر ستونی که چنین استدلالی برای آن برقرار بوده است را از کار کنار گذاشته‌ایم.

مورد بعدی که در داده‌ها به چشم می‌خورد، ستون‌های دسته‌ای بودند که در دیتاست اول بیشتر نمایان بودند و در دیتاست دوم از قبل این ستون‌ها بررسی شده بودند و روی آن‌ها کارهایی مانند encode کردن انجام شده بود. پس در مورد دیتاست اول کاری که انجام شده است تبدیل ستون‌های دسته‌ای مانند flgs (که نوع flag یک پکت را نشان می‌دهد) به متغیرهای dummy است.

مورد دیگری که در این مرحله می‌توانست به ما کمک کند تا از تعداد ستون‌ها کم کنیم و حجم داده را کاهش دهیم و همچنین از ورود چندباره یک نوع اطلاعات به مدل‌ها جلوگیری کنیم، بررسی correlation بین ستون‌های موجود در داده‌ها بود، به این ترتیب که از بین هر دو ستونی که دارای مقدار absolute correlation بیشتر از ۹۵ درصد بودند، یکی را از داده‌ها حذف کردیم که این کار در دیتاست اول منجر به حذف ۷ ستون و در دیتاست دوم منجر به حذف ۲۳ ستون شده است.

کار دیگری که می‌توانست در این مرحله برای کاهش تعداد ستون‌ها انجام شود، استفاده از dimension reduction با استفاده از الگوریتم PCA بود، منتهی مورد قابل توجه این است که ستون‌ها موجود در داده‌ها با یکدیگر correlation چندان قوی ندارند و اکثرا مستقل از یکدیگر هستند، پس اعمال الگوریتم‌هایی مانند PCA روی داده بیشتر باعث کاهش کیفیت داده می‌شدند و این مورد ما را از انجام چنین موردی منصرف کرد و تصمیم به ادامه کار با هر تعداد ستونی که موجود بود (هر چند زیاد) گرفتیم.

مورد آخر که این مورد تنها در مورد دیتاست اول به کار رفته است، Resampling داده‌هاست. در ستون هدف دیتاست اول، ۱۹۲۶۶۲۴ سطر مربوط به پکت‌هایی هستند که مربوط به حمله DDos هستند و تنها ۴۷۷ ستون مربوط به پکت‌های سالم هستند. ابتدا قصد بر این بود تا این ناتوازنی را به کمک روش SMOTE برطرف کنیم اما با توجه به حجم داده‌ی موجود و محدود بودن منابع سخت‌افزاری موجود، هیچ‌گاه موفق به انجام این کار نشدیم و به ناچار از روش ساده‌ای که در طول درس نیز آموزش داده شده بود استفاده کردیم تا این مورد به هر شکل کنترل شود. از فرمول زیر استفاده شد تا در نهایت به نسب ۰.۳، داده‌ها شامل پکت‌های سالم باشند.

$$x = \frac{p(records) - rare}{1 - p}$$

۴.۳- پیش‌پردازش داده‌ها

در این زمنیه، یک مورد که در تنها در مورد دیتاست دوم انجام شد، حذف سطرهایی بود شامل مقدار بی‌نهایت بودند، از آنجایی که تعداد این سطرها در مقایسه با تعداد کل سطرها بسیار ناچیز بود، از انجام کارهایی مانند جایگزینی مقادیر بی‌نهایت با مقدار mod در ستون مورد نظر، اجتناب کردیم و هر سطری که چنین مقادیری را شامل بود، کاملا از داده حذف شد.

در نهایت هر دیتاست به کمک min-max-scaler نرمال‌سازی شده است و به نسبت ۰.۲، داده اصلی را به مجموعه‌های train و test تقسیم کرده‌ایم.

۴.۴- ایجاد مدل‌های مناسب

همانطور که قبلا ذکر شد ما در مورد دیتاست اول از دو مدل decision tree و random forest استفاده کردیم و در مورد دیتاست دوم نیز از random forest بهتره بردیم. دلایل استفاده از هر مدل نیز به این صورت است که decision tree تفسیرپذیری بالایی دارد و مدلی ساده‌ای جهت پیاده سازی است و دومی نیز مزیت‌هایی مانند انعطاف پذیری مدل و سادگی در استفاده را ارائه می‌دهد. همچنین مدل Random Forest از آن جهت برای کار ما مناسب است که قادر است با وجود دیتاست‌هایی با تعداد ابعاد بسیار زیاد نیز کار خود را به درستی انجام دهد.

۴.۵- اجرای فرآیندهای آموزشی و نتایج آن

پس از آنکه هر مدل روی داده‌های train آموزش داده شده است و نتایج روی داده‌های تست بررسی شده است، شاهد این هستیم که در رابطه با دیتاست اول به کمک هر دو مدل، دقت ۱۰۰ درصد را خروجی می‌گیریم و هیچ مقدار False Positive با False Negative نیز وجود ندارد. در مورد دیتاست دوم دقت به مقدار بسیار کمی کاهش می‌یابد و برابر با ۹۹ درصد است و ۴۶ مقدار False Positive و ۲۰ مقدار False Negative داریم که عملکرد بسیار خوبی است.

همچنین قابل ذکر است که متود classification_report از کتابخانه sklearn، این مقادیر را به درستی برای ما نمایش نمی‌داد و ما برای اطمینان از صحت بیشتر، از متود precision_recall_fscore_support نیز استفاده

کرده‌ایم. آنچه که در رابطه با دلیل عملکرد نادرست متود classification_report جویای آن شدیم در لینک زیر به تفصیل بررسی شده است:

classification_report and sklearn confusion_matrix: values do not match?

۴.۶- بررسی مدل به دست آمده و تفسیر نتایج

دقت مدل‌ها روی داده تست در قسمت قبلی بررسی شد که دقت‌های بسیار مناسبی را نیز ارائه می‌دهند. مورد دیگری که قابل بحث بیشتری است، تفسیر مدل Decision Tree در رابطه با دیتاست اول است که اصلا این مدل بیشتر به همین خاطر استفاده شده است که طبق نتایج به دست آمده سه متغیر تاثیر به سزایی در پیش بینی نتایج دارند که ابتدا متغیر N_IN_Conn_P_DstIP است که از بین تمام متغیرها بیشتر تاثیرگذار است و سپس دو متغیر TnP_PDstIP و TnP_PerProto که البته متغیر N_IN_Conn_P_DstIP در مقایسه با دو متغیر دیگر بازم هم بسیار تاثیرگذارتر است.

۴.۷- پیاده سازی راه حل‌های جدید برای افزایش دقت

از آن‌جا که با راه حل‌هایی که تا کنون بیان کردیم دقت‌های بسیار خوبی به دست آوردیم، تصمیم بر آن شد که همین روش‌ها را ارائه دهیم. می‌شد مدل‌های شبکه عصبی را نیز امتحان کرد اما از طرفی زمان بسیاری که صرف آموزش آن‌ها می‌شد و همچنین احتمال زیاد overfit شدن مدل‌ها ما را از پیاده‌سازی شبکه عصبی منصرف کرد.

۴.۸- نتیجه گیری نهایی

همانطور که در ابتدا نیز ذکر شده دیتاست‌هایی که در این زمینه موجود هستند، این امکان را فراهم می‌آورند که به کمک مدل‌های نه چندان پیچیده، پیش‌بینی‌هایی با دقت‌های بسیار بالا به دست آیند همانطور که در سایر کارهای مشابه نیز قابل مشاهده است. از نقاط قوت فرآیند طی شده می‌توان به این مورد اشاره کرد که تقریبا تمام مراحلی که در یک فرایند داده‌کاوی استاندارد باید طی شوند، در کار ما نیز تا حد ممکن و با توجه به شکل داده، پیاده‌سازی شدند که نتیجه آن نیز با مشاهده دقت‌ها قابل توجه است.