The Hybrid Technique for DDoS Detection with Supervised Learning Algorithms

2019

75 citations

Computer Networks

فریمورکی نوین و هیبریدی بر پایه دیتااستریم مبتنی بر یادگیری افزایشی برای شناسایی حملات منع خدمت توزیع شده ارائه می‌دهیم. بار پردازشی را بر اساس منابع موجود برای پردازش سریع بین سمت کلاینت و پروکسی تقسیم می‌کنیم. (برخلاف اکثر روش‌های قبلی که متمرکز[1] بودند). از الگوریتم‌های یادگیری ماشین مختلف در سمت پروکسی(سرور) استفاده می‌کنیم: random forest, decision tree, MLP, K-NN . و نتیجه می‌گیریم که random forest بهترین نتایج را ارائه می‌دهد.

روشهای دفاع به سه نوع در سمت مبداء، قربانی و یا شبکه و میان افزارها می‌باشند. روش‌های سمت قربانی با وجود مزیت‌هایشان به دلیل حجم بالای حملات از عدم سازگارپذیری در پهنای باند بالا رنج می‌برند. همانطور که می‌دانیم روش ما بر مبنای دفاع در شبکه می‌باشد که می‌تواند ترافیک را فیلتر کند[2] یا نرخ آن را محدود کند[3]. این مکانیزم می‌تواند منبع حملات را با همکاری مشترک بین آداپتورهای شبکه پیداکند. در این مکانیزم آداپتورها ترافیک متخاصم و معمولی را با هم دریافت می‌کنند، لذا فیلترینگ گزینه مناسبی نمی‌باشد. در نتیجه گزینه بهتر تعریف محدودیت برای ترافیک می‌باشد. به طور مثال:

- فیلترینگ بسته مبتنی بر مسیر
- شناسایی و فیلتر مسیریاب‌های متخاصم

مشکلات شبکه‌های امروزی داده های بزرگ[4] و استریم داده‌ها[5] می باشند. برای حل مشکل دو راهکار موجود می‌باشد:

- پردازش موازی: الگوریتم را به چندین بخش تقسیم می کنند.
- پردازش افزایشی[6]:

پس راه حل، استفاده از یادگیری افزایشی می‌یابد. اما بایستی سرعت پردازش بیشتری نسبت به الگوریتم‌های batch داشته باشند. لذا راهکارهایی که از این روش‌ها استفاده می‌کنند، داده را در یک بازه زمانی و تنها یکبار می‌خوانند.

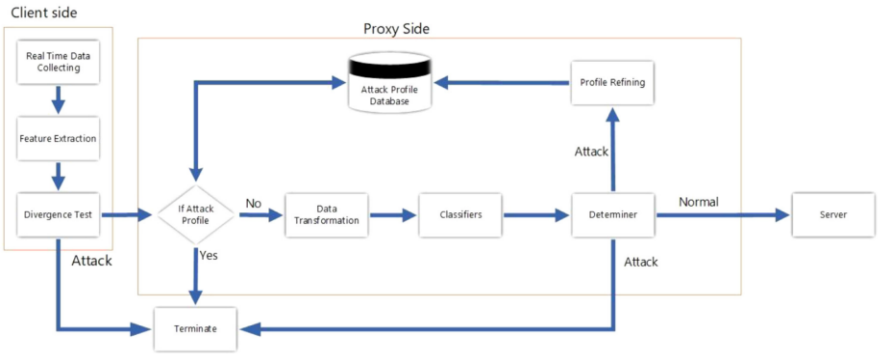
راهکار ارائه شده:

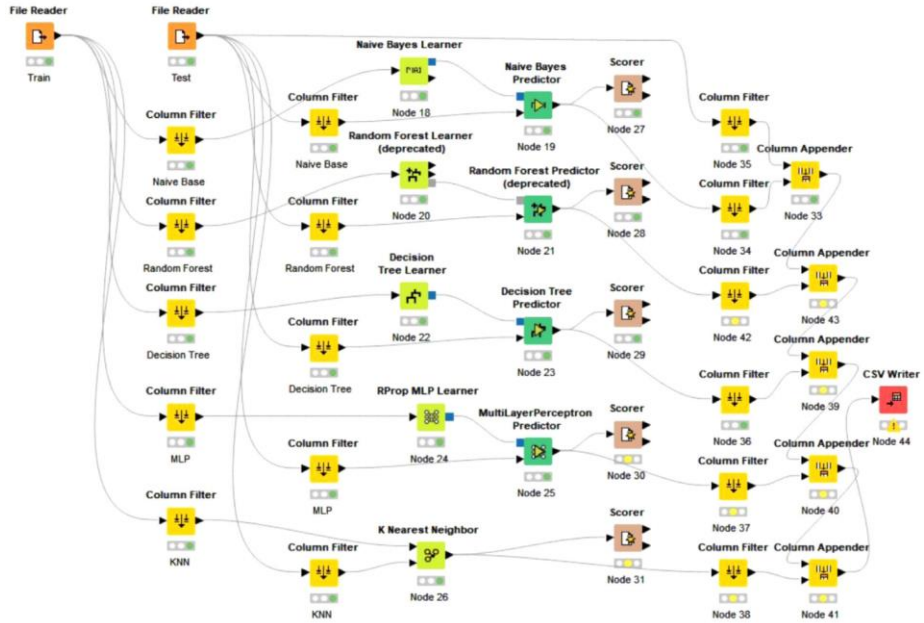
در این راهکار علاوه بر استفاده از مزیت تمامی الگوریتم‌های یادگیری با کمک بهره گیری از همه آنها، از یک determiner برای بهبود نتایج استفاده می‌کنیم. ادمین در determiner چندین خط قانون طبقه بندی کننده برای شرایط مختلف می‌تواند تعریف کند.

[1] Centeralized
[2] Traffic filtering
[3] Rate limit
[4] Big-data
[5] DataStream
[6] Incremental Processing

سمت کلاینت: طبقه‌بندی کننده[7]ها را آموزش می‌دهیم و در بخش forward selection ویژگی‌های مناسب برای هر الگوریتم را به صورت جدا پیدا می‌کنیم. ابتدا با یک مجموعه ویژگی تهی شروع می‌کنیم و در هر دور اجرا یک ویژگی برای بهبود کارایی مدل اضافه می‌کنیم و تا جایی ادامه می‌دهیم که اضافه کردن ویژگی تاثیری نداشته باشد.

سمت پروکسی: پس از انجام پردازش سمت کلاینت، برای جلوگیری از سربار محاسباتی، ابتدا داده‌های دریافتی با پایگاه داده پروفایل حمله بررسی می‌شود و در صورتی که با هیچ پروفایلی مطابقت نداشت، داده‌ها را با فرمت مناسب به سمت پروکسی ارسال می‌کند که از یادگیری ماشین برای پردازش استفاده می‌کند. برای پردازش از الگوریتم‌های ساده بیز، جنگل تصادفی، درخت تصمیم، MLP و K-NN استفاده می‌کنیم و سپس یک تعیین‌کننده الگوریتم بر نتایج اجرا می‌شود که بر اساس خطمشی پیکربندی‌شده خود تصمیم می‌گیرد و نتیجه بهتری را ارائه می‌دهد. رویکرد مبتنی بر یادگیری افزایشی با تک تک داده های ورودی، پروفایل را به روز می‌کند. از طرف دیگر، اینجا تنها از یک پایگاه داده پروفایل استفاده می‌کنیم که از تراکم پروفایل ها جلوگیری می‌کند. در ادامه، پایگاه داده با نتایج جدید بهبود می‌یابد





[7] Classifier