****

**دانشگاه صنعتی شریف**

**دانشکده مهندسی کامپیوتر**

**پایان نامه کارشناسی**

**مهندسی کامپیوتر**

**تولید ترافیک هجوم ناگهانی کاربران و ارزیابی سرور‌ها**

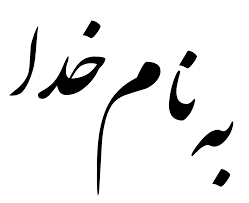
**نگارش:**

**علیرضا حسن پور**

**استاد راهنما:**

**دکتر امیرحسین جهانگیر**

بهار 1402



**چکیده**

حملات منع سرویس توزیع شده، نوعی از تهدیدهای امنیتی مرتبط با شبکه های کامپیوتری میباشند

که دسترس پذیری منابع شبکه را هدف قرار میدهند. یکی از ویژگیهای این نوع حملات حجم ترافیک

بالا و یا درخواست سرویس توسط تعداد زیادی مهاجم غیرمجاز که با هم شبکه ای از رباتها را تشکیل

میدهند، میباشد که باعث کاهش کارایی شبکه میشوند. مسئله تشخیص تعداد زیاد کاربران مجاز در

هجوم ناگهانی از درخواستهای غیرمجاز حمله، امروزه یکی از بزرگترین چالشهای پیش رو متخصصان

امنیت شبکه میباشد. روشهایی که تاکنون ارائه شده اند، عمدتاً یا کارایی لازم را نداشته‌اند و یا با بالا

رفتن دانش مهاجمین در تقلید رفتار کاربران مجاز، در کوتاه مدت پاسخگو بوده‌اند. روشهایی که عملکرد

بهتری نسبت به سایر روشها داشته‌اند اکثراً بر اساس استخراج ویژگیهای آماری عمل میکنند.

هدف از این پروژه تولید یک جریان ترافیک است که سالم (حمله منع سرویس نباشد) و طبیعی (ساختگی به نظر نرسد) باشد به طوری بتواند رفتار یک گروه از کاربران را شبیه سازی کند تا بتوان سامانه ها و سرور های موجود را پیش از قرار گرفتن زیر بار مورد ارزیابی قرار داد.

این پژوهش ابتدا به بررسی ویژگیهای ترافیک های حمله و هجوم ناگهانی کاربران می پردازد. سپس رفتار کاربران مختلف را با جریان رسیده از رباتهای حمله منع سرویس مقایسه میکند. این مقایسه شامل بررسی ویژگیهای مختلف آماری زمان بین بسته های پشت سر هم رسیده از یک مشتری (یک کاربر در هجوم ناگهانی و یا یک ربات در حمله منع سرویس( می باشد.

**واژه های کلیدی :**

حملات منع سرویس ، هجوم ناگهانی کاربران، ارزیابی کارایی و ترافیک شبکه .

**فهرست مطالب**

فصل 1: مقدمه

فصل 2 : مفاهیم اولیه و کار های پیشین

فصل 3 : روش پیشنهادی

فصل 4 : ارزیابی نتایج

فصل 5 : نتیجه گیری و کار های آینده

مراجع

فهرست جدول ها

فهرست شکل ها

فصل 1:

مقدمه

هجوم ناگهانی وقتی اتفاق میافتد که تعداد زیادی از کاربران در یک بازه زمانی کوتاه اقدام به دسترسی به منابع یک شبکه کنند. در این حالت سرویس دهنده ممکن است توان پاسخگویی به حجم عظیم درخواست دهندگان را نداشته باشد و با مشکل مواجه شود و حتی در برخی موارد به‌طور کامل از کار بیافتد. برای مثال، ورود تعداد زیادی کاربر به یک سایت خبری در هنگام وقوع رویدادی خاص می تواند نمونه ای از هجوم ناگهانی باشد. ازآنجایی که افزایش ترافیک رسیده به سرویس دهنده در حملات منع سرویس تا حد زیادی مشابه هجوم ناگهانی کاربران میباشد، تشخیص حملاتی که این هجوم را شبیه سازی و تقلید می کنند و یا همزمان با وقوع این رویداد اعمال می شوند بسیار دشوار است و نیازمند تحلیل و بررسی دقیق ترافیک شبکه و شناخت ویژگیهای مختلف هر یک از موارد فوق برای تفکیک آن ها از یکدیگر می باشد. تا آنجایی که در سالهای اخیر نوع جدیدی از حملات منع سرویس معرفی شده اند که به آنها حمله هجوم ناگهانی کاربران گفته میشود. این حملات به صورت توزیع شده انجام میشود و حجم ترافیک به صورت ناگهانی افزایش می یابد به طوریکه سیستم های تشخیص حمله موجود نمیتوانند تشخیص دهند که تعداد زیادی از کاربران که با آدرس های مختلف در حال درخواست سرویس هستند، کاربران مجاز سیستم هستند و یا با هدف خرابکارانه بخشی از یک حمله منع سرویس می باشند. همانطور که گفته شد ممکن است که این حملات همزمان با هجوم ناگهانی کاربران نیز اتفاق بیفتد. برای مثال، یک رقیب اقتصادی که از هجوم خریداران به سایت رقیب خود در ساعات مشخصی آگاه است، میتواند اقدام به یک حمله منع سرویس در آن زمان نموده که ترافیک خاص از ترکیب این حمله با هجوم کاربران میتواند بسیار بزرگ و غیر قابل روانه سازی باشد و پیش بینی های در نظر گرفته شده در مورد میزان سرویسدهی و پهنای باند را با خطا و اختلال مواجه کند تشخیص حملات با گستردگی بالا و تعداد حمله کننده زیاد و یا حملاتی که در حین هجوم ناگهانی کاربران اتفاق میافتند، یکی از چالشهای پیش روی متخصصان و محققین امنیت شبکه میباشد.

ما در فصل 2 به مفاهیم اولیه و مرور کارهای پیشین می پردازیم. سپس در فصل 3 ، روش پیشنهادی را ارائه می‌کنیم. در فصل 4 به ارزیابی نتایج اختصاص دارد و در نهایت در فصل 5 به نتیجه‌گیری و جمع‌بندی کارهای انجام شده و ارائه چشم‌انداز کارهای آتی می‌پردازیم.

فصل 2 :

**مفاهیم اولیه و کار های پیشین**

یک روش و سیستم تشخیص حملات منع سرویس باید ویژگیهای زیر را دارا باشد:

1. تشخیص سریع

ازآنجایی که حملات منع سرویس بسیار سریع عمل می‌کنند و در زمان بسیار کمی حجم ترافیک بسیار بالا میرود، اگر عمل تشخیص به موقع انجام نشود، ممکن است بعد از تشخیص، توانایی مقابله با حمله وجود نداشته باشد. به عبارتی، سیستم وقتی متوجه حمله منع سرویس شود که حمله تأثیر مورد نظر خود را گذاشته و برای مقابله با آن دیر شده است . بنابراین باید دقت کرد که لایه های دفاعی را قبل از سرویس دهنده مورد نظر قرار داد. برای مثال اگر بتوان حمله را در یک مسیریاب تشخیص داد، میتوان عمل مقابله با آن را در آن مسیریاب و یا حتی قبل تر انجام داد و سرو یس دهنده با استفاده از مسیریاب های دیگر تعبیه شده به کار خود ادامه دهد. این مسئله اهمیت تشخیص حمله منع سرویس در لایه های پایینتر شبکه به خصوص لایه شبکه و لایه پیوند داده را بیش از پیش روشن می‌کند. تشخیص حمله در لایه های بالاتر مانند لایه کاربرد نیازمند بررسی محتویات بسته ها در یک سرویس گیرنده است که نیازمند صرف منابع مختلف و زمان است.

1. قابلیت اطمینان

روش پیاده سازی شده باید به درستی میان ترافیک رسیده از کاربران مجاز هنگام هجوم ناگهانی کاربران و ترافیک حمله تمایز قائل شود. هرچند ممکن است در عمل این مساله امکانپذیر نباشد، اما در روش مورد استفاده، باید تعداد تشخیصهای منفی نادرست و مثبت نادرست، صفر باشد. در عمل میتوان به این شرط بسنده کرد که این موارد حداقل باشند.

1. امکانپذیری (عملی بودن)

روش مورد نظر باید قابلیت پیاده سازی در دنیای واقعی را با توجه به زیرساخت ها و همبندی ها ی موجود در شبکه ها، داشته باشد. برای مثال، روشهایی که نیازمند نگهداری حجم عظیم اطلاعات هستند و یا نیاز به انجام پردازش های سنگین روی داده دارند، ممکن است کارایی لازم را نداشته باشند و پیاده سازی آنها از لحاظ عملی امکانپذیر نباشد.

1. تشخیص بی‌درنگ

روش مورد نظر باید به گونه‌ای باشد که به صورت بی‌درنگ و به محض دریافت ترافیکهای غیرطبیعی عملیات تشخیص خود را آغاز کند. ازآنجایی که ویژگی حمله منع سرویس، بالا بردن حجم ترافیک است، در صورت گذر زمان و عدم تشخیص بیدرنگ، حجم ترافیک بالا رفته و حمله به میزان زیادی اثرگذار خواهد بود. بنابراین هرگونه تأمل اضافی در تشخیص ترافیک حمله ممکن است حتی در صورت تشخیص، مقابله با حمله را غیرممکن سازد.

1. انعطاف‌ پذیری

روش مورد نظر باید توانایی تشخیص حملات مختلف را داشته باشد و مختص به یک حمله خاص نباشد. مهاجم ممکن است پروتکل مورد استفاده یا روش خود را تغییر دهد. این تغییر روش نباید بر روی سیستم تشخیص تأثیر خاصی داشته باشد و یا بتوان با حداقل تغییرات در سیستم آن را با تغییرات انجام شده توسط مهاجم، سازگار کرد.

اساس کارهایی که تا کنون در این مورد انجام گرفته اند، چه روشهایی که در سطح جریان و چه روشهایی که مقایسه ها در سطح کاربران انجام می دهند، بر پایه ویژگیهای آماری جریان قرار دارد. طبق تحقیقی که انجام گرفته است، مهاجم در بهترین حالت تنها میتواند از 30 درصد سیستم هایی که به ربات حمله آلوده شده اند و تحت کنترل درآمده اند، برای تولید ترافیک حمله استفاده کند. دلیل این امر نیز این است تمامی این سیستمها درآن واحد روشن و در دسترس نیستند و بعلاوه هماهنگی و بکار گرفتن همه این رباتها با توجه به اینکه روی شبکه های گوناگون با پهنای باندهای مختلف و زیرساختها ی مختلف قرار دارند دشوار میباشد. لذا ویژگیهای آماری ترافیکی که توسط یک کاربر مجاز در هجوم ناگهانی کاربران تولید میشود، با ترافیکی که توسط یک برنامه خودکار یا نیمه خودکار که با دریافت دستور حمله، چه به صورت زمان بندی شده و چه به صورت دریافت دستور حمله، اقدام به تولید ترافیک میکند تفاوت هایی دارد.

**هجوم ناگهانی کاربران**

منظور از هجوم ناگهانی کاربران درخواست تعداد زیادی کاربر به یک سرویس دهنده وب در یک بازه

زمانی کوچک است به طوریکه تعداد درخواستهای کاربران به سرویس دهنده، به صورت نمایی افزایش

می‌یابد. این در حالی است که سرویس دهنده موردنظر در حالت عادی، این تعداد سرویس گیرنده نداشته و

ممکن است برای آن برنامه ریزی نشده باشد که این باعث اختلال در عملکرد سخت افزاری و نرم افزاری آن خواهد شد. به همین دلیل است که این پدیده را جزو ناهنجاری‌(1)ها ترافیک شبکه دسته بندی میکنند.

رویدادهایی همچون رقابتهای ورزشی جهانی و بازیهای المپیک، منتشر شدن نسخه جدید از یک

نرم افزار محبوب مانند نسخه‌های جدید یک سیستم عامل یا وقوع حوادث سیاسی و اجتماعی مانند

حملات تروریستی و... نمونه هایی از وقایعی هستند که می توانند هجوم ناگهانی کاربران را به سایتها ی

مرتبط با آنها در پی داشته باشند .

اگر تعداد درخواستهای رسیده از کاربران در دقیقه، به‌صورت نمایی رشد کند، هجوم ناگهانی اتفاق افتاده است. این مسئله در رابطه 1 نشان داده شده است.

که در این رابطه بیانگر میانگین نرخ درخواستها در بازه زمانی ti میباشد. طبق این رابطه اگر تعداد

درخواستها از مقداری مشخص که با توجه به درخواستهای قبلی هر کارگزار منحصربه فرد ، متفاوت میباشد، بیشتر شود، هجوم ناگهانی کاربران اتفاق افتاده است. البته به طورکلی، رشد تعداد درخواست های کاربران در یک بازه ی زمانی تا حدی که سرویس دهنده مجبور باشد برای ادامه سرویس دهی خود، عملیات خود را مدیریت و تنظیم کند، هجوم ناگهانی نامیده میشود.

**تاریخچه ی هجوم ناگهانی کاربران**

اصطلاح رویداد هجوم ناگهانی کاربران در سال 1970، سالها قبل از ابداع شبکه اینترنت، توسط نویسنده ای به نام لاری نیون ، در یک رمان داستانی علمی تخیلی به نام پرواز اسب ، مورد استفاده قرار گرفت. در داستان این کتاب، دانشمندی ماشین زمانی اختراع میکند که افراد به وسیله آن میتوانند به گذشته، زمانی که رویداد خاصی اتفاق افتاده است، سفر کنند. اوج داستان زمانی اتفاق می افتد که جمعیت انبوهی برای سفر به زمان رویداد خاصی هجوم می آورند و این هجوم ناگهانی باعث میشود که تغییراتی در آن رویداد رخ داده و گذشته تغییر کند که این موضوع باعث بروز ناهنجاری و آشوب هایی میشود.

سالها بعد و با ظهور شبکه اینترنت این اصطلاح، به دلیل مشابهت فراوان این داستان با هجوم کاربران به یک کارگزار وب که باعث افزایش درخواستهای رسیده به آن، به صورت نمایی میشود، وارد دنیای فناوری اطلاعات شد.

**اثر SlashDot**

اثر اسلش‌دات به افزایش موقت ترافیک به یک وب‌سایت اشاره دارد، که می‌تواند زمانی رخ دهد که یک وب‌سایت پربازدید، پیوندی به سایت یا وبلاگ کوچک‌تر ارسال می‌کند، بنابراین افزایش بی‌سابقه‌ای در ترافیک ایجاد می‌کند. اگر افزایش ترافیک بسیار زیاد باشد، باعث کاهش سرعت سایت یا غیرقابل دسترس شدن آن می شود.

در این حالت پدیده هجوم ناگهانی کاربران برای سایت مذکور اتفاق میافتد. سایت SlashDot یک سایت خبری در مورد آخرین اخبار تکنولوژی و نرم افزار است که بسیاری از لینک های خبری آن، به سایت کوچک دیگر ارجاع داده میشود. به همین دلیل این پدیده که برای اولین بار در این سایت اتفاق افتاده است، به اثر SlashDot معروف است .

دسته بندی های مختلفی در مورد انواع هجوم ناگهانی انجام شده است. برای مثال،تقسیم بندی هجوم ناگهانی به دو نوع قابل پیشبینی و غیر قابل پیشبینی، عادی و بحرانی، دو نوع از این دسته بندی ها هستند . برای مثال، هجوم کاربران در مواقعی مانند وقتی که زمان معینی برای انتشار محصولی جدید، مانند نسخه جدید یک سیستم عامل ، گوشی هوشمند و ... تعیین شده است، از پیش محتمل است. اما در مواقعی که یک حادثه غیرمنتظره مانند وقوع یک حمله تروریستی و یا حوادثی مانند طوفان و ... اتفاق میافتد، هجوم ناگهانی کاربران قابل پیش بینی نیست.

در مواقع قابل پیش بینی می توان جلوی از کار افتادن سرویس دهنده را گرفت و یا تا حد زیادی از کاهش کارایی آن جلوگیری کرد. اما در برخی مواقع، به خصوص وقتی که این پدیده غیر قابل پیش بینی است، هجوم ناگهانی کاربران باعث از کار افتادن سرویس دهنده می شود و هزینه های زیادی را در بر دارد. این نوع از هجوم ناگهانی کاربران، بحرانی نامیده میشود.

**نمونه هایی از هجوم ناگهانی کاربران در سالهای اخیر**

در 29 جولای 2015 که تاریخ عرضه سیستم عامل ویندوز10 ، از سوی شرکت مایکروسافت اعلام شده بود، شماری زیادی از کاربران برای به روزرسانی سیستم عامل های خود به این نسخه، به کارگزارهای این شرکت هجوم آوردند. باوجود اینکه این شرکت این هجوم را پیش بینی کرده بود و شبکه های توزیع محتوایی با پهنای باند40 ترابایت در ثانیه برای آن در نظر گرفته بود، اما باز هم تعداد زیادی از کاربران، در ساعات اولیه موفق به به روزرسانی سیستم عامل خود نشدند. البته این هجوم از قبل پیش بینی شده بود و بسیاری از سایتهای خبری دنیا با تیتر "امروز پرترافیک ترین روز تاریخ اینترنت خواهد بود " به استقبال آن رفته بودند.

در سال 2012 ، طوفان "سندی" بخش وسیعی از سواح شرقی ایالات متحده را درنوردید. در پی این حادثه طبیعی، در عرض تنها چند ساعت، استفاده از اینترنت در جهان113 درصد افزایش یافت. سایت نتفلیکس با افزایش ناگهانی 150 درصدی ترافیک مواجه شد. همچنین استفاده از سرویسهای اسکایپ 112 درصد افزایش یافت. این مورد که در مواجهه با یک رویداد طبیعی رخ داد، یکی از بزرگترین موارد هجوم ناگهانی کاربران در سالهای اخیر محسوب میشود .

در سال 2010 و در طی برگزاری جام جهانی در آفریقای جنوبی، حجم عظیمی از کاربران به سایت توییتر هجوم آوردند، به طوریکه حجم توییتهای کاربران در طول یک بازی به 750 توییت در ثانیه رسید. البته ماجرا به همینجا ختم نشد و با به ثمر رسیدن گل در یک مسابقه، این سایت با 2940 توییت در ثانیه مواجه شد که باعث از سرویس خارج شدن کل سایت به مدت چندین دقیقه شد که خسارت های فراوانی به آن وارد کرد. البته این اتفاق در طول بازیهای المپیک و بازیهای زمستانی نیز، بسیار اتفاق می‌افتد. یکی از بزرگترین و اولین موارد هجوم ناگهانی کاربران، در سال1998 و در طول بازی های نیمه نهایی جام جهانی فرانسه اتفاق افتاد که باعث شد سایت اصلی رقابت ها برای مدتی ترافیک بسیار سنگینی را شاهد باشد.

در سال 2014سایت فروش مایکروسافت آفیس، به دلیل هجوم کاربران برای خرید، از سرویس خارج شد، این اتفاق برای سایت اپل که نسخه جدید سیستم عامل iOS خود را برای دانلود در اختیار کاربرانش قرار داده بود، در سال 2013 نیز رخ داد.

سایتهای خبری بزرگ دنیا در حوادثی مانند یازده سپتامبر و همچنین حوادث بمبگذاری در لندن در سال 2008 با حجم عظیمی از ترافیک از سوی بازدیدکنندگان خود مواجه شدند. موارد دیگری ازجمله رویدادهایی در سایت های خبری که خبر از درگذشت یک شخصیت معروف، یک خبر اقتصادی سیاسی مهم و ... میدهند، نمونه هایی از هجوم ناگهانی کاربران در سال های اخیربوده اند.

از سرویس خارج شدن سرویس دهنده ی شرکت های بزرگ اینترنتی جهان در هنگام هجوم ناگهانی کاربران در سالهای اخیر نشان میدهد که باوجود قابل پیشبینی بودن در برخی موارد و اتخاذ تمهیدات مختلف برای مقابله با این پدیده، هنوز هم این مسئله نیازمند تحقیقات بیشتر و ارائه راهکارهای کارآمدتری برای فائق آمدن بر آن است.

**شباهت ها و تفاوت های هجوم ناگهانی کاربران و حملات منع سرویس**

مواردی چون بالا رفتن تعداد درخواستها و افزایش حجم ترافیک به صورت ناگهانی، بالا رفتن زمان پاسخ دهی و ناپایداری سیستم و درنهایت از دسترس خارج شدن سیستم، ویژگی های مشترک بین حمله منع سرویس و هجوم ناگهانی کاربران هستند. اما باید در نظر داشت که حمله منع سرویس توسط منبع مهاجم و با اهداف خرابکارانه صورت می گیرد. همچنین با وجود افزایش توانایی ها و دانش مهاجمین برای تقلید عملکرد کاربران مجاز، هنوز ازنظر پراکندگی آدرسهای اینترنتی، تعداد کاربران منحصربه فرد ، الگوی درخواست ها و بسیاری از موارد دیگر، تفاوت هایی بین این دو پدیده وجود دارد که از آنها می توان برای تشخیص حملات و پیشگیری از آنها استفاده کرد.

اولین تفاوتی که ازنظر ترافیکی بین این دو پدیده وجود دارد نحوه افزایش و ثبات حجم ترافیک است. در هجوم ناگهانی کاربران، نرخ ترافیک در واحد زمان به صورت نوسانی می باشد که نمودار آن به صورت امواج زیگزاگی مشاهده می شوند. درحالیکه ترافیک حمله منع سرویس پس از رسیدن به میزان مشخصی، معمولاً نرخ ثابتی را در پیش می گیرد. دو نمودار شکل 1 بیانگر این تفاوت هستند



**شکل 1 نرخ ترافیک در واحد ثانیه برای هجوم ناگهانی کاربران )راست) و حمله منع سرویس )چپ(**

دلیل این تفاوت، ماهیت دو ترافیک می باشد. در هجوم ناگهانی کاربران، پس از اینکه کاربران متوجه

افت کیفیت سرویس دهی شوند و یا با از کار افتادن های متوالی سرویس دهنده مواجه شوند، به دو دسته

تقسیم میشوند، دسته ای بر ارسال درخواست های خود اصرار می ورزند و دسته ای نیز دلسرد شده و سعی

میکنند زمان دیگری را برای دسترسی مورد نظر خود انتخاب کنند. با خروج این دسته از کاربران، کاربران دیگر وضع بهتری در سرویس دهی پیدا می کنند و به خواسته های خود میرسند. البته همانطور که گفته شد دلیل هجوم ناگهانی کاربران، رویداد خاصی است که طبیعتاً با گذشت زمان و اطلاع کاربران در مورد آن، حجم درخواست ها به تدریج کاهش می یابد. اما همانطور که گفته شد، حمله منع سرویس توسط رباتها حمله و به صورت برنامه ریزی شده و زمانبندی شده انجام میشود. بنابراین حمله در لحظه‌های خاص آغاز شده و با توجه به هدف آن و مکانیزم های دفاعی در مقابل آن، مدت زمان مشخصی به طول می انجامد. البته در حملات جدیدتر مهاجمان سعی می کنند افزایش نرخ ترافیک به صورت ناگهانی نباشد، اما کنترل افت ترافیک در پایان حمله با توجه به مکانیزم های دفاعی و همچنین نیاز به هماهنگی و در دسترس بودن رباتها، هنوز مساله ای دشوار برای مهاجمان است. شکل 2 نمودار مربوط به تعداد درخواستها و همچنین تعد اد آدرسهای اینترنتی خلاصه شده منحصربه فرد را برای دو مورد، هجوم ناگهانی کاربران و حمله منع سرویس نشان میدهد.



**شکل 2 تعداد درخواستها و تعداد آدرسهای - اینترنتی خلاصه شده در هجوم ناگهانی کاربران**

**)بالا( و حمله منع سرویس )پایین(**

همانطور که مشاهده می شود در هر دو مورد تعداد درخواست ها به طور ناگهانی افزایش یافته است،

اما آنچه در این بین متفاوت است، تعداد خوشه های آدرس خلاصه شده می باشد. در هجوم ناگهانی

کاربران، تعداد خوشه های منحصربه فرد بعد از خلاصه سازی، با روزها و ساعات قبل تفاوتی ندارد و حتی

نسبت به بعضی روزها مقدار کمتری نشان میدهد. این مطلب نشان دهنده این است که فقط تعداد درخواستها نسبت به روزهای مشابه افزایش یافته است. نه تعداد آدرس و خوشه های منحصربه فرد. اما در مورد حمله منع سرویس، مشخص است که مهاجم از آدرس های اینترنتی جعلی استفاده کرده است که به صورت یکنواخت و تصادفی تولید شده اند. به همین علت هنگام خلاصه سازی تعداد خوشه های منحصربه فرد تقریباً برابر تعداد آدرس های موجود است. منظور از خلاصه سازی، پیدا کردن بزرگترین زیر شبکهای است که چند آدرس مختلف روی آن قرار دارند.

مهمترین تفاوت این دو پدیده از منظر سرویس دهنده، نحوه مواجهه با آنهاست. در هنگام وقوع

هجوم ناگهانی، باید تمهیداتی اندیشیده شود تا باوجود افزایش حجم ترافیک و مشکلات ناشی از آن،

رضایت کاربران تأمین شود و درخواستهای آنان پاسخ داده شود. این کار را می‌توان با افزایش قابلیت های

سخت افزاری، استفاده از شبکه های محتوا توزیع شده و ... انجام داد. اما در مورد حمله منع سرویس، آنچه

اهمیت دارد جلوگیری از رسیدن درخواست های غیرمجاز و عدم پاسخ به آنها از جانب سرویس دهنده

است. این کار میتواند با استفاده از پازل های گرافیکی ، مسدود‌کردن آدرسهای مهاجمین و رباتها

در مسیریاب های بالادستی، استفاده از تله عسل و ... انجام داد.

در جدول 1، مقایسه میان حمله منع سرویس و هجوم ناگهانی کاربران به صورت تجمیعی آورده شده است:

|  |  |
| --- | --- |
| **هجوم ناگهانی کاربران** | **حمله منع سرویس** |
| سرویس‌دهنده و شبکه با حجم زیادی از ترافیک دریافتی اشباع می‌شوند. | سرویس‌دهنده و شبکه با حجم زیادی از ترافیک دریافتی اشباع می‌شوند. |
| ترافیک به‌وسیله کاربران مجاز ارسال می‌شود و پاسخ‌دهی به آن ضروری است. | ترافیک به‌وسیله کاربران غیرمجاز ارسال می‌شود و نیازی نیست به آن پاسخ داده شود. |
| به دو دسته فابل پیش‌بینی و غیرقابل پیش‌بینی تقسیم میشود و در پی علاقه کاربران در مورد رویدادی خاص رخ می‌دهد. | غیرقابل پیش‌بینی است و به‌واسطه استفاده مهاجم از سیستم آلوده به ربات که به آن زامبی گفته‌ می‌شود ، حمله رخ می‌دهد. |

**جدول 1 مقایسه کلی ویژگی های حمله منع سرویس و هجوم ناگهانی کاربران**

حال می‌خواهیم به ابزار‌های تولید ترافیک و ارزیابی شبکه بپردازیم :

1. **سلنیوم**

سلنیوم یک ابزار تست خودکار است. ما می‌توانیم هر برنامه نرم‌افزار و هر اپلیکیشن تلفن همراه را با استفاده از Selenium تست کنیم. سلنیوم مجموعه‌ای از ابزار‌های نرم‌افزاری مختلف است که هر کدام با روشی متفاوت از اتوماسیون تست پشتیبانی می‌کند. بیشتر مهندسان QA Selenium بر روی یک یا دو ابزار که بیشترین نیاز‌های پروژه را برآورده می‌کنند تمرکز می‌کنند، با این حال یادگیری تمام ابزار‌ها، گزینه‌های مختلفی را برای نزدیک شدن به رفع مشکلات مختلف اتوماسیون تست به شما خواهد داد. هدف آن، مجهز بودن به مجموعه‌ای از توابع تست است که نیاز‌های تست هرگونه اپلیکیشن وب را برطرف می‌کند. یکی از ویژگی‌های کلیدی Selenium پشتیبانی از اجرای تست‌ها در چندین پلتفرم مرورگر است.

Selenium یک ابزار تست خودکار متن باز برای برنامه‌های وب در تمامی مرورگر‌های مختلف پلتفرم‌ها است. تست انجام شده با استفاده از ابزار سلنیوم معمولاً به عنوان تست سلنیوم نامیده می‌شود. از آنجایی که سلنیوم مجموعه‌ای از ابزار‌های مختلف است، پس دارای توسعه‌دهندگان مختلف است.

پیش‌نویس اولیه‌ی برنامه‌ی WebDriver API W۳C منتشر شد سلنیوم یک فریمورک تست کاربردی وب است که به شما اجازه می‌دهد تا در بسیاری از زبان‌های برنامه‌نویسی مثل جاوا، سی شارپ، Groovy، پرل، پی اچ پی، پایتون و روبی تست‌های مورد نظر را بنویسید. سلنیوم در ویندوز، لینوکس و سیستم عامل MAC استفاده می‌شود.

**معرفی تست خودکار**

خودکارسازی تست نرم‌افزار با استفاده از ابزار‌های تخصصی برای کنترل اجرای تست‌ها و مقایسه نتایج واقعی با نتایج مورد انتظار می‌باشد. معمولاً تست‌های رگرسیون، که اقدامات تکراری هستند، به صورت خودکار انجام می‌شود. ابزار تست نه تنها به ما در انجام تست‌های رگرسیون کمک می‌کند، بلکه به ما کمک می‌کند تا تولید داده‌ها، نصب محصول، تعامل GUI، لاگ کردن نقص‌ها و غیره را به طور خودکار انجام دهیم.

تست خودکار یک تکنیک می‌باشد که یک اپلیکیشن یا نرم‌افزار برای پیاده‌سازی کل چرخه عمر نرم‌افزار در زمان کم از آن استفاده و بهره وری و اثربخشی را برای نرم‌افزار تست فراهم می‌کند. تست خودکار یک تکنیک اتوماتیک است که تست‌کننده‌ها آن را مستقیماً می‌نویسند و از نرم‌افزار مناسب برای تست نرم‌افزار استفاده می‌کنند. معمولاً، تستر‌ها اسکریپت‌های تست و موارد تست را با استفاده از ابزار خودکارسازی بکار می‌گیرند و سپس آن‌ها را به سوئیت‌های تست گروه‌بندی می‌کند. هدف اصلی تست خودکار، افزایش کارایی تست و توسعه ارزش نرم‌افزاری است.

**ویژگی‌های سلنیوم (Selenium)**

* سلنیوم یک چارچوب تست وب منبع باز و قابل حمل است.

سلنیوم IDE بدون نیاز به یادگیری زبان برنامه‌نویسی آزمون، قابلیت پخش و ضبط را برای نوشتن تست فراهم می‌کند.

این می‌تواند به عنوان پیشرو تست بستر مبتنی بر ابر در نظر گرفته شود که به آزمایش‌کنندگان کمک می‌کند تا اقدامات خود را ضبط کنند و آن‌ها را به عنوان یک اسکریپت قابل استفاده مجدد با یک رابط کاربری ساده برای درک و استفاده آسان صادر کنند.

سلنیوم از سیستم عامل‌های مختلف، مرورگر‌ها و زبان‌های برنامه‌نویسی پشتیبانی می‌کند. در زیر لیست آمده است:

زبان‌های برنامه‌نویسی: C#، Java، Python، PHP، Ruby، Perl و JavaScript

سیستم عامل: Android، iOS، Windows، Linux، Mac، Solaris.

مرورگر‌ها: Google Chrome، Mozilla Firefox، Internet Explorer، Edge، Opera، Safari و غیره.

همچنین از اجرای تست موازی پشتیبانی می‌کند که باعث کاهش زمان و افزایش کارآیی تست‌ها می‌شود.

سلنیوم برای جمع‌آوری کد منبع می‌تواند با چهارچوب‌هایی مانند mavenیکپارچه شود.

سلنیوم همچنین می‌تواند با آزمایش چارچوب‌هایی مانند TestNG برای تست برنامه و تولید گزارش‌ها یکپارچه شود.

* سلنیوم در مقایسه با سایر ابزار‌های تست اتوماسیون به منابع کمتری نیاز دارد.

WebDriver API در سلنیوم یکی از مهمترین اصلاحات انجام شده در سلنیوم است.

درایور وب سلنیوم به نصب سرور احتیاج ندارد، اسکریپت‌های تست به طور مستقیم با مرورگر تعامل دارند.

دستورات سلنیوم از نظر کلاس‌های مختلف طبقه‌بندی می‌شوند که درک و اجرای آن ساده‌تر است.

کنترل از راه دور سلنیوم (RC) در رابطه با WebDriver API با عنوان سلنیوم ۲. ۰ شناخته می‌شود. این نسخه برای پشتیبانی از صفحات وب پر جنب و جوش و آژاکس ساخته شده است.

**مولفه‌های سلنیوم (Selenium)**

سلنیوم یک سیستم منبع باز است که به آزمایش برنامه‌های مختلف وب کمک می‌کند. این قابل استفاده در مرورگر‌های مختلف است و کاربرد آن بسیار آسان است. این مجموعه ابزار‌هایی است که روش‌های مختلف آزمایش را ارائه می‌دهد. چهار مؤلفه وجود دارد که از سلنیوم تشکیل شده است. آن‌ها سلنیوم IDE، سلنیوم RC، درایور وب و شبکه هستند. این زبان از زبان‌های مختلف مانند جاوا، PHP، روبی و پایتون پشتیبانی می‌کند. سلنیوم مستقل بودن از سکو از مزیت دیگری برخوردار است. این کتابخانه دارای کلاس‌ها، روش‌ها و رابط‌های مختلفی است. این دلیل اصلی استفاده سلنیوم توسط سازمان‌های مختلف است.

* محیط توسعه یکپارچه سلنیوم (IDE)
* کنترل از راه دور سلنیم (RC)
* WebDriver
* شبکه سلنیوم (GRID)
* محیط توسعه یکپارچه سلنیوم (IDE)

این ساده‌ترین مجموعه موجود در مجموعه سلنیوم است. سلنیوم IDE اجازه ضبط و پخش اسکریپت‌ها را می‌دهد. با استفاده از پسوند Firefox پیاده‌سازی می‌شود. با استفاده از سلنیوم می‌توانید تست‌های مختلفی را ضبط، ویرایش و اشکال زدایی کنید. تست‌های ایجاد شده با داشتن تست‌های مختلف در مجموعه‌های مختلف به راحتی قابل تنظیم است. عملکرد ضبط و پخش در نمونه‌سازی ابزار کمک می‌کند و سهولت یادگیری را فراهم می‌کند. این به عنوان یک راه حل کلی برای توسعه و نگهداری سوئیت‌های مختلف عمل می‌کند. داشتن دانش پایه در مورد HTML، JavaScript و DOM می‌تواند در استفاده از بهترین سلنیوم IDE به آزمایش‌کننده کمک کند. وقتی آزمایش‌کننده موارد تست را ایجاد می‌کند، ویژگی تکمیل خودکار این موارد را نشان می‌دهد. این به وارد کردن سریع دستورات کمک می‌کند و باعث صرفه‌جویی در وقت می‌شود. همچنین شانس ورود به دستورات نامعتبر را کاهش می‌دهد.

سرور سلنیوم به قصد استفاده از درایور وب بستگی دارد. اگر قرار است مرورگر و موارد آزمایش بر روی همان دستگاه اجرا شوند، دیگر نیازی به اجرای سرور سلنیوم نیست. درایور وب مستقیماً اجرا خواهد شد. برای جدا کردن تست‌ها بر روی چندین ماشین و ماشین مجازی می‌توانید از شبکه سلنیوم استفاده کنید. هنگامی که تستر از راه دور یا از طریق یک مرورگر خاص به هم وصل می‌شود، به سرور سلنیوم نیاز خواهد بود. رابط درایور وب روش‌های مختلفی را که می‌تواند برای تست پایان به پایان استفاده شود را اعلام می‌کند. کلاس آزمایشی ایجاد شده باید در اجرای خاص مرورگر استفاده شود. این کلاس تست می‌تواند مرورگر وب را کنترل کند و همچنین عناصر مختلفی را که می‌توان از صفحات HTML بارگیری کرد، انتخاب کنید. مرورگر وب API‌هایی را ارائه می‌دهد که به کنترل از راه دور برنامه‌های وب تست رابط کاربر کمک می‌کنند.

Selenium-WebDriverاز مرورگر‌های زیر همراه با سیستم عامل‌های این مرورگر‌ها پشتیبانی می‌کند:

* Google Chrome
* Internet Explorer
* Vista
* Windows
* Firefox
* Safari
* Opera
* HtmlUnit
* Phantomjs
* Android
* iOS

1. Wireshark

Wireshark یک تحلیل‌گر بسته‌های شبکه رایگان و منبع باز است. این ابزار برای رفع اشکالات شبکه، تحلیل، توسعه نرم‌افزار و پروتکل‌های ارتباطی و آموزش استفاده می‌شود. در ابتدا به نام Ethereal شناخته می‌شد، اما در ماه مه سال 2006 به دلیل مسائل مربوط به علامت تجاری به Wireshark تغییر نام داده شد.

Wireshark یک برنامه چندسکویی است که در نسخه‌های کنونی از این ابزار از کتابخانه Qt برای پیاده‌سازی رابط کاربری استفاده می‌کند و از pcap برای ضبط بسته‌ها استفاده می‌کند. این برنامه بر روی سیستم‌عامل‌های لینوکس، macOS، BSD، Solaris و برخی سیستم‌عامل‌های مشابه یونیکسی و ویندوز مایکروسافت اجرا می‌شود. همچنین نسخه‌ای بدون رابط گرافیکی به نام TShark نیز وجود دارد. Wireshark و دیگر برنامه‌های همراه با آن مانند TShark، نرم‌افزارهای رایگانی هستند که تحت شرایط مجوز عمومی GNU نسخه 2 یا نسخه‌های بعدتر منتشر می‌شوند.

قابلیت‌ها

Wireshark بسیار شبیه به tcpdump است، اما دارای رابط کاربری گرافیکی و گزینه‌های مرتب‌سازی و فیلتر کردن یکپارچه است.

Wireshark به کاربر امکان می‌دهد کنترل‌کننده‌های رابط شبکه را در حالت promiscuous قرار دهد (اگر توسط کنترل‌کننده رابط شبکه پشتیبانی شود)، بنابراین او می‌تواند تمام ترافیک قابل رویت در آن رابط را شامل ترافیک یونیکستی که به آدرس MAC کنترل

‌کننده رابط شبکه ارسال نشده است، مشاهده کند. با این حال، هنگام ضبط بسته‌ها با استفاده از یک تحلیل‌گر بسته در حالت promiscuous روی یک پورت در یک سوئیچ شبکه، تمام ترافیک از طریق سوئیچ لزوماً به پورتی که ضبط انجام می‌شود ارسال نمی‌شود، بنابراین ضبط در حالت promiscuous برای مشاهده تمام ترافیک شبکه کافی نیست. آینه‌سازی پورت یا انواع دیگری از سیم‌کشی شبکه، امکان ضبط در هر نقطه‌ای از شبکه را فراهم می‌کند. تپ‌های ساده و بی‌حرکت بسیار مقاوم در برابر دستکاری هستند.

در لینوکس، BSD و macOS، با استفاده از libpcap نسخه 1.0.0 یا بالاتر، Wireshark 1.4 و نسخه‌های بعدی می‌توانند کنترل‌کننده‌های رابط شبکه بی‌سیم را در حالت نظارتی قرار دهند.

اگر یک دستگاه از راه دور بسته‌ها را ضبط کند و بسته‌های ضبط شده را با استفاده از پروتکل TZSP یا پروتکل استفاده شده توسط OmniPeek به یک دستگاهی که Wireshark در آن اجرا می‌شود، ارسال کند، Wireshark این بسته‌ها را تجزیه و تحلیل می‌کند، بنابراین می‌تواند بسته‌های ضبط شده در یک دستگاه از راه دور را در زمان ضبط تحلیل کند.

1. Locust

Locust یک ابزار تست کاربری ساده است که برای اجرای آزمایشی وب‌سایت -یا سیستم‌های دیگر- ساخته شده تا مشخص شود که آن وب‌سایت چه تعداد کاربر را می‌تواند در لحظه هَندل کند. ایدهٔ پیاده‌سازی این ابزار به این شکل است که در طول یک پروسهٔ تست وب‌سایت، گروهی از اصطلاحاً «ملخ‌ها» می‌توانند به وب‌سایت شما حمله کنند. رفتار هر ملخ (یا همان کاربر فرضی)، توسط شما تعریف شده و پروسهٔ هجوم آوردن از طریق یک رابط کاربری نظارت می‌شود. این جنگ آزمایشی به شما کمک خواهد کرد تا تنگناهای کد خود را قبل از ورود کاربران واقعی شناسایی و رفع کنید.

کاربران فرضی یا ملخ‌ها به طور کامل مبتنی بر رویداد (Event) هستند و در نتیجه، امکان پشتیبانی از هزاران نفر از کاربران به طور هم‌زمان در یک سیستم واحد وجود دارد. این ابزار در مقایسه با بسیاری از دیگر سرویس‌های مبتنی بر رویداد، از اصطلاحاً Callback استفاده نمی‌کند.

در واقع، هر حملهٔ ملخ‌ها به سایت شما داخل سندباکس خودش اجرا می‌شود که چنین فرایندی به شما اجازه می‌دهد تا به صورتی کاملاً شفاف، الگوریتم‌های مختلفی در #پایتون بنویسید بدون این که با کال‌بک‌ها، سورس‌کد خود را شلوغ کنید. برخی از مهم‌ترین ویژگی‌های این ابزار عبارتند از:

ارسال سناریوهای تست کاربران در پایتون: نیازی به رابط کاربری پیچیده یا کدهای عجیب و غریب XML ندارید؛ مثل هر پروژهٔ پایتون دیگری، به سادگی می‌توانید کد بزنید.

توزیع پذیری/مقیاس پذیری (پشتیبانی از صدها هزار کاربر): Locust از آزمایش‌های لود شدن بر روی چندین دستگاه پشتیبانی می‌کند. با توجه به این که لوکاست مبتنی بر رویداد است، می‌توانید هزاران نفر از کاربران را در یک پروسه هَندل کنید. بخشی از دلیل چنین سیاستی این است که حتی اگر شما اقدام به شبیه‌سازی تعداد بالایی از کاربران نمایید، همهٔ آن‌ها به طور جدی به سیستم شما هجوم نمی‌آورند؛ اکثر مواقع، کاربران نمی‌داند که چه کاری می‌خواهند انجام دهند. به عبارت دیگر، ریکوئست‌های در لحظه هرگز به معنای تعداد کاربران آنلاین وب‌سایت نیست!

رابط کاربری مبتنی بر وب: فرانت‌اند Locust با HTML/JS کدنویسی شده که تمام جزئیات مربوطه را به صورت در لحظه (Real-time) نشان می‌دهد و در صورت نیاز هم قادر خواهید بود که آن را کاستومایز کنید.

قابلیت تست هر سیستمی را دارد

با این که لوکاست مبتنی بر وب است، اما می‌توان آن را برای تست تقریباً هر نوع اپلیکیشنی به کار برد. فقط یک کلاینت برای آزمایش آن چیزی را که می‌خواهید تست کنید نوشته و با ملخ‌ها به آن هجوم ببرید!

اپن‌سورس است: لوکاست اپن‌سورس است؛ لذا به سادگی می‌توان آن را هک کرده و بسته به نیازهای خود این ابزار را دستخوش تغییر ساخت و جالب است بدانیم که پیچیدگی‌های ابزارهای تست فعلی، دلیل اصلی ایجاد Locust بوده است.

)) Locust یک ابزار تست عملکرد و بار شبیه‌سازی شده است که به صورت منبع باز و رایگان در دسترس است. این ابزار برای ارزیابی عملکرد سیستم‌های نرم‌افزاری و تست بار شبکه مورد استفاده قرار می‌گیرد. با استفاده از Locust، می‌توان تعداد بالایی از کاربران همزمان و تراکم بالای تراکنش‌ها را شبیه‌سازی کرده و عملکرد سیستم را در شرایط مختلف مورد ارزیابی قرار داد.

این ابزار با استفاده از زبان برنامه‌نویسی پایتون پیاده‌سازی شده است و قابلیت توسعه و سفارشی‌سازی فراوانی دارد. استفاده از مفهوم "کد به عنوان ساز" به کاربران اجازه می‌دهد تست‌های دقیق و پیچیده را با استفاده از قابلیت‌های برنامه‌نویسی پایتون ایجاد کنند.

Locust امکاناتی مانند مدیریت کاربران همزمان، شبیه‌سازی رفتار کاربران و ترافیک واقعی، ثبت و پیگیری نتایج آزمایش‌ها، مانیتورینگ عملکرد سیستم در طول زمان و ایجاد گزارش‌های جامع را فراهم می‌کند.

این ابزار با دارا بودن رابط کاربری کاربرپسند و قابلیت مدیریت آسان، برای کاربرانی که با مفاهیم تست عملکرد آشنایی کمی دارند نیز مناسب است. همچنین، Locust از پروتکل HTTP و WebSocket پشتیبانی می‌کند و می‌تواند با سیستم‌هایی که از این پروتکل‌ها استفاده می‌کنند، ارتباط برقرار کند.

استفاده از Locust در فرآیند توسعه نرم‌افزار، باعث بهبود کیفیت و پایداری سیستم می‌شود و به ت

یم‌های توسعه و عملیات کمک می‌کند تا با تمرکز بر عملکرد سیستم، مشکلات و ضعف‌ها را بهبود دهند و اطمینان حاصل کنند که سیستم به درستی عمل خواهد کرد.

با اینکه Locust ابزاری رایگان است، اما دارای جامعه فعالی از توسعه‌دهندگان است و به‌روزرسانی‌ها و پشتیبانی منظمی دارد. این ابزار با مجوز MIT منتشر شده است و قابلیت انتشار، استفاده و تغییرات آزاد را به کاربران می‌دهد.((