

گزارش پروژه دوم

پروژه بررسی عملکرد TCP و UDP در شرایط مختلف کانال

- تشریح دقیق و مرحله به مرحله کد پایتون

پس از مشاهده فیلم های آموزشی و بررسی دقیق فایل PDF شامل پروژه، شرایط و لوازم لازم برای شروع کار را مهیا نمودم. با نصب سیستم عامل LINUX – UBUNTO کارم را آغاز کرد. از Pycharm به عنوان IDE خود استفاده کردم و کد پایتون را نوشتم. در کد پایتون با چند موضوع جدید روبرو بودم. اول از همه کار با کتابخانه Pyshark مورد اصلی بود. به دلیل نا شناخته بودن این کتابخانه، داک خیلی معروفی از این موجود نبود. با مطالعه سایت developers این کتابخانه به آدرس <https://github.com/KimiNewt/pyshark> اطلاعات کافی برای کد زدن را کسب نمودم. در ابتدای کد متغیری از نوع LiveCapture ساختم، و با استفاده از کلیدواژه های کتابخانه پایشارک، فیلتر هایی تعریف نمودم تا فقط از کارت شبکه لوپ بک و پورت ۴۰۴۰، بسته کیچر کند. سپس نیاز به چیزی مشابه دکمه شروع در وایرشارک بود، به این منظور با کمک متد sniff این را پیدا سازی کردم تا بتوانم پکت های این فیلتر مشخص فوق را مشاهده نمایم. در اینجا مدت زمان این عملیات را ۲۰ ثانیه تعریف نمودم تا پس از آن متوقف شود. بنابراین توانستم پکت ها را به درستی دریافت کنم. حالا باید تحلیل انجام میدادم. طبق خواسته متن پروژه نیازمند شمارش تعداد بسته های دریافتی، تعداد باز ارسال ها و گذردهی بودم، بنابراین با متدهایی این داده ها شمارش شد و به صورت خروجی برگردانده شد. بخش دیگر این کد به کار با فایل جایسان iperf اختصاص داده شد. علی الخصوص نمودارهای فرستنده و گیرنده توسط این فایل باید تحلیل و رسم میشد. بنابراین به یک کتابخانه دیگر هم احتیاج داشتم! کتابخانه matplotlib و بخش Pyplot آن کار رسم این نمودار ها را برایم انجام می دهد. بسیار خب، حالا من با خروجی جیسان گرفتن از iperf و دادن این فایل ها به متد ها و راهکار هایی که با جستجو و بررسی آرایه وار فایل جیسان آنرا تحلیل میکند، داده های لازم را با کمک متد های این کتابخانه به آن دادم و بدین ترتیب برای من نمودار ها را رسم شد و عملاً خروجی کد پایتون من این نمودار ها، که اولی نمودار فرستنده و دیگری نمودار گیرنده است، و همچنین پارامتر های ذکر شده در بالا، شد. بدین ترتیب کد پیاده سازی شد. سپس چهار ترمینال باز کردم و کد پایتون را در یکی، اسکریپت SC را در دومی و کلاینت و سرور را دو تای دیگر اجرا کردم و نتایج مطلوب نظر PDF صورت پذیرفت و موارد به پیوست در فایل موجود می باشد.

❖ پاسخ به سوالات متن در قسمت "برسی عملکرد TCP":

TCP	کانال ایده آل	کانال با تاخیر ثابت	کانال با تاخیر تصادفی	کانال با گم شدگی کم	کانال با گم شدگی زیاد
میانگین گذردهی فرستنده	۴۹۹۹۷۱۵۰۸۹۰۱۸۳۱۲۷	۷۲۲۷۳۵۰۱۳۰۵۵۴۰۹۱۹	۱۰۱۴۶۶۱۲۳۳۰۹۷۸۳۴۷۵	۲۲۰۴۷۸۱۵۷۷۰۵۰۶۶۸	۹۱۴۹۷۸۱۷۶۰۳۲۰۴۱۹۱
تعداد بسته دریافتی	۱۷۰	۲۰۸	۲۲۰	۲۷۹	۳۰۳
تعداد باز ارسال ها	۰	۰	۴۴	۶	۱۴

۱. افزایش تاخیر و گم شدن بسته ها باعث کاهش میانگین گذردهی فرستنده خواهد شد که این موضوع در جدول فوق هم قابل

مشاهده است. هر چه پکت ها با تاخیر بیشتری ارسال شوند، در سمت دریافت کننده هم دیرتر دریافت خواهد شد و این باعث

کاهش گذردهی است. همچنین هر چه تعداد پکت بیشتری گم شود، تعداد باز ارسال ها هم بیشتر میشوند و در نتیجه

گذردهی کاهش میابد.

۲. بله، صد در صد. در سوال قبل هم دیدیم که افزایش تاخیر و گم شدن بسته ها و همچنین شرایط مختلف کانال، بسیار در این

تفاوت تاثیر گذارند.

۳. ما اکنون این موارد را در پروتکل TCP بررسی میکنیم و همانطور که میدانیم گم شدن بسته ها در این پروتکل خیلی اهمیت

دارد و هر گاه بسته ای گم شود، به منظور دریافت مجدد آن باز ارسال میشود. بنابراین هر چه میزان گم شدگی زیادتر باشد،

میزان باز ارسال ها هم بیشتر خواهد شد.

۴. در کانال با تاخیر تصادفی به دلیل آنکه بسته به Timeout میخورد، مجددا ارسال میشود و خب در این حالت مدام تعداد باز

ارسال های ما افزایش می یابند.

۵. نخیر، چنین نیست. همانطور که بالاتر گفتم، تعداد بسته های ارسالی وابسته به شرایط کانال و سایر موارد جانبی، تغییر

میکند. بنابراین تعداد باز ارسال ها هم به شرایط کانال بستگی دارد و ثابت نیستند و چنین شرایطی برای مقایسه وجود ندارد

و در نتیجه چنین مولفه ای، معیار خوبی برای مقایسه عملکرد نخواهد بود.

۶. بالاتر هم اشاره کردم که پروتکل TCP به دلیل قابل اعتماد یا Reliable بودنش، در هنگام مواجهه با گم شدگی بسته ها آنها را مجددا ارسال میکند و بنابراین گذردهی در این مسیر کاهش می یابد.

۷. همانطور که چند بار در بالا هم اشاره کردم، به دلیل آنکه در پروتکل TCP، که یک پروتکل قابل اعتماد هست، هستیم، در هنگام مواجهه با گم شدگی بسته ها، این پروتکل میکوشد تا این اثر را جبران نماید. پس بنابراین آنها را مجددا ارسال میکند و در نتیجه گذردهی در این مسیر کاهش می یابد.

❖ پاسخ به سوالات متن در قسمت "برسی عملکرد UDP":

کانال با گم شدگی زیاد	کانال با گم شدگی کم	کانال با تاخیر تصادفی	کانال با تاخیر ثابت	کانال ایده آل	UDP
۴۹۹۹۸۳۲۲۹۸.۰۶۹۵۶۵	۴۹۹۹۷۹۹۰۸۴۰۰۴۳۸۴۷	۴۹۹۹۶۲۲۰۸۳.۴۹۵۹۵۸	۴۹۹۹۴۹۰۱۶۳.۱۸۴۹۳۳	۴۹۹۹۵۲۳۶۹۵.۹۲۳۹۹۹	میانگین گذردهی فرستنده
۲۲	۲۰	۲۹	۲۱	۱۹	تعداد بسته دریافتی
۳	۱	۳	۰	۰	تعداد باز ارسال ها

- همانطور که از نامش پیداست، پروتکل UDP است، پروتکلی که اهمیت ویژه ای برای گم شدگی بسته ها قائل نیست و بنابراین تغییر چشمگیری در میانگین گذردهی احساس نمیشود. این به دلیل آن است که این پروتکل بر خلاف TCP از نوع قابل اعتماد و یا Reliable نیست و به همین دلیل گم شدگی بسته ها اثرآنچنانی بر میزان گذردهی نخواهند داشت.
- با توجه به داده های جدول، قابل مشاهده است که هر چه گم شدگی بیشتر باشد، گذردهی اندکی افزایش را شاهد خواهد بود. این افزایش نمیتواند مستدل باشد و تنها با تغییرات لحظه ای شرایط حاکم بر کانال قابل توجیه است.
- تصمیم سختی است! چرا که شاخصه های مهمی میتواند تاثیر گذار باشد. انتخاب یکی از این دو، کاملاً بستگی به نیاز و فعالیت ما خواهد داشت. اگر امنیت برایمان اولویت داشته باشد، TCP انتخاب بهتری است ولی اگر سرعت اتصالات برایمان حیاتی باشد، UDP بهتر از عهده این بر می آید! به طور کلی UDP برای شبکه های اجتماعی، VoIP و بازی های آنلاین بهتر است اما برای کارهای بانکی پیشنهاد نمی شود و از امنیت بالا برخوردار نیست. با اینکه انتخاب، انتخاب چالشی ای است اما من معتقدم که خوب است از پروتکل UDP که سریع تر است، استفاده کنیم مگر اینکه مشکلی با اتصال داشته باشیم.

ملوین مختاری (۹۸۳۱۱۴۳)