

به نام خدا

دانشگاه صنعتی اصفهان - دانشکده مهندسی کامپیوتر

درس شبکه‌های کامپیوتری ۲ - بهار ۱۴۰۴

طراحی، شبیه‌سازی و تحلیل پیشرفته توپولوژی‌های شبکه با SDN و Mininet

مرحله ۱ - توپولوژی پایه Single (۲۰٪)

ابتدا یک توپولوژی ساده (Single) ایجاد کنید. برای این منظور، فایل `base_topo.py` را به گونه‌ای بنویسید که شامل یک سوئیچ (`s1`) و تعدادی میزبان (`h1` تا `hn`) باشد. تعداد میزبان‌ها (`N`) برابر با رقم آخر شماره دانشجویی شما به اضافه عدد ۴ خواهد بود.

پس از ساخت توپولوژی و اجرای اسکریپت، وارد محیط CLI شوید و دستورات زیر را اجرا کنید:

`pingall` و `dump`، `links`، `net`، `nodes`

خروجی هر یک از دستورات را در فایل به نام `outputs/base_single.txt` ذخیره کنید و در گزارش نهایی، هر خروجی را به طور واضح تحلیل و توضیح دهید. سپس پایداری لینک بین میزبان `h1` و سوئیچ `s1` را بررسی کنید. برای این منظور ابتدا با دستور `link s1 h1 down` لینک را قطع و پس از اجرای مجدد دستور `pingall`، تغییرات جدول `MAC` و `RTT` را ثبت کنید. سپس لینک را با دستور `link s1 h1 up` مجدداً برقرار کرده و تغییرات را تحلیل نمایید. علاوه بر این، اسکریپت خود را طوری توسعه دهید که بتوانید تعداد میزبان‌ها را به صورت پارامتری و با دستور زیر تعیین کنید:

`sudo python3 base_topo.py --hosts 10`

خروجی دستور `dump` در این حالت را نیز توضیح دهید.

در پایان این مرحله به این سؤال مفهومی پاسخ دهید:

چرا قطر شبکه (diameter) در توپولوژی Single همواره برابر با ۲ است، در حالی که در توپولوژی Linear با همان تعداد میزبان ممکن است قطر شبکه بزرگ‌تر شود؟

مرحله ۲ - مقایسه پنج توپولوژی مرسوم (۲۵٪)

در این مرحله، پنج توپولوژی معروف شامل `Linear`، `Minimal`، `Tree`، `Torus` و `Full` را از نظر ساختار و عملکرد بررسی کنید.

برای هر یک از توپولوژی‌ها، ابتدا یک نمودار واضح و دقیق به کمک ابزارهایی مانند `Graphviz` یا `NetworkX` رسم کنید. در هر نمودار، درجه هر نود (`Node Degree`)، قطر شبکه (`Diameter`) و قابلیت تحمل خرابی (`Fault Tolerance`) را مشخص کنید.

سپس برای هر یک از این توپولوژی‌ها، یک اسکریپت جداگانه در `Mininet` بنویسید. برای هر توپولوژی دستورات `pingall` و تست عملکردی `iperf3` را هم در حالت `TCP` و هم در حالت `UDP` اجرا نمایید. نتایج به دست آمده را برای میانگین و انحراف

معیار RTT و Throughput استخراج کرده و در قالب جدول، نمودار ستونی (Bar Chart) و نمودار جعبه‌ای (Box-Plot) ارائه دهید.

در نهایت، به صورت مقایسه‌ای در ۱ تا ۲ پاراگراف، مزایا، معایب و کاربردهای واقعی هر یک از این توپولوژی‌ها را توضیح دهید.

مرحله ۳ - تحلیل ترافیک و امنیت (۳۰٪)

یکی از توپولوژی‌های مرحله قبل (ترجیحاً توپولوژی Tree) را انتخاب کرده و آن را به یک کنترلر SDN مانند Ryu یا POX متصل کنید.

در کنترلر، سیاست‌های زیر را پیاده‌سازی کرده و کدهای مربوطه را در گزارش ضمیمه کنید:

- محدود کردن پهنای باند یکی از میزبان‌ها (QoS)
 - هدایت ترافیک HTTP و SSH از مسیرهای متفاوت (Load Balancing ساده)
- سپس با استفاده از ابزار tcpdump یا Wireshark، ترافیک شبکه را در حداقل دو میزبان و یک سوئیچ ضبط کنید و فایل‌های ضبط‌شده (pcap) را ذخیره نمایید.

یکی از سناریوهای زیر را برای ایجاد ناهنجاری انتخاب کنید و به صورت مصنوعی آن را ایجاد کنید:

- حمله UDP Flood توسط ابزار hping3
 - جعل آدرس MAC (حمله ARP Spoof)
 - قطع ناگهانی لینک شبکه و بررسی بازیابی مسیرها
- پس از ایجاد ناهنجاری، مراحل تشخیص، تحلیل و دفاع را به‌طور کامل شرح دهید و نمودار زمانی تغییرات پهنای باند شبکه را نیز در گزارش ضمیمه نمایید.

تحویل پروژه و مستندات

تمام اسکریپت‌های نوشته‌شده به زبان Python و فایل‌های پیکربندی را در یک مخزن Git یا فایل ZIP تحویل دهید. گزارش پروژه را در قالب PDF و به زبان فارسی، به طول ۸ تا ۱۵ صفحه و با فونت بی‌نازنین سایز ۱۳ تهیه نمایید. نام فایل گزارش باید به‌صورت زیر باشد:

CN2_402xxxxxx_Name.pdf

این پروژه کاملاً فردی است و هرگونه شباهت غیرمنطقی باعث نمره صفر خواهد شد.