بسم‌الله الرحمن الرحيم

دانشگاه آزاد اسلامی

واحد تهران جنوب

دانشکده فنی و مهندسی

پایان‌نامه کارشناسی

مهندسی رشته – گرایش

مهندسی فناوری اطلاعات (IT)

عنوان:

معرفی و ارزیابی ترافیک و امنیت در شبکه های تعریف شده نرم افزاری

استاد راهنما:

سرکار خانم دکتر بختیاری

نام و نام خانوادگی دانشجو:

ثنا باقری خوزانی

شماره دانشجویی:

۹۲۱۱۷۱۰۱۶۸

تاریخ فارغ‌التحصیلی ۱۳۹۹/۰۲

سپاسگزاری

خداوند سبحان را سپاسگزارم که به بنده حقیر توفیق انجام و اتمام پژوهش حاضر را عنایت فرمود.

از تمامی اساتید، دوستان و خانواده‌ام که در تمام دوران تحصیلی مرا حمایت کرده‌اند تقدیر و تشکر می‌کنم و این پایان‌نامه را تقدیم می‌کنم به خانواده‌ام که پشتیبان همیشگی من در زندگی بوده‌اند.

فهرست مطالب

فهرست جداول

فصل دوم

فهرست اشکال

فصل اول

فصل سوم

چکیده فارسی

شبکه‌های مبتنی بر نرم‌افزار شبکه نرم‌افزاری تعریف‌شده[1] یک شبکه جدید است. تا قبل از ورود این شبکه جدید، پیاده‌سازی شبکه نیاز به مهارت و تخصص زیادی داشت و هزینه نصب و استفاده از سیستم‌های پیچیده متشکل از چند فنّاوری، هزینه‌بر بود. ایده اصلی مفهوم شبکه نرم‌افزاری تعریف‌شده در سال ۱۹۹۶ و برای ایجاد مدیریت کاربر روی ارسال و دریافت در گره‌های شبکه مطرح شد. برای پیاده‌سازی این شبکه به ۲ عامل نیاز داریم:

۱- یک معماری منطقی مشترک، بین تمامی سوییچ‌ها، روترها و دیگر تجهیزات وجود داشته باشد.

۲- به یک پروتکل امن برای تجهیزات شبکه نیاز است. مثل پروتکل‌های: Ipsilon 1996، IETF 2000، Ethan 2007 و OpenFlow 2008

[1] SDN

مقدمه

با توسعه مداوم و عمیق اپلیکیشن محاسبات ابری و اینترنت اشیا، معماری شبکه سنتی قادر به برآورده نمودن الزامات زمینه‌های صنعت حال حاضر، مانند سیستم‌های فیزیکی–سایبری، نسل پنجم شبکه‌های بی‌سیم (5G) و اینترنت وسایل نقلیه نیست؛ بنابراین برخی از محققان پیشنهاد نموده‌اند که شبکه‌های مبتنی بر نرم‌افزار برای محیط صنعتی که ظرفیت انعطاف‌پذیری و نوآوری اینترنت اشیا در آن افزایش خواهد یافت، عملی و کاربردی خواهد بود.

امنیت و مهندسی ترافیک هر دو از موضوعات پژوهشی اپلیکیشن‌های شبکه نرم‌افزاری تعریف‌شده درزمینه محیط‌های صنعتی هستند. در این پایان‌نامه قصد بررسی کلی شبکه‌های نرم‌افزاری تعریف‌شده و سپس امنیت و ترافیک این شبکه‌ها را دارم.

فصل اول:
معرفی شبکه‌های نرم‌افزاری تعریف‌شده

۱-۱ شبکه نرم‌افزاری تعریف‌شده چیست؟

شبکه تعریف‌شده توسط نرم‌افزار، روشی برای مدیریت شبکه‌هایی است که در آن بخش کنترل و بخش انتقال از هم جدا هستند. برای این کار از نرم‌افزار مدیریت توابع شبکه به‌وسیله یک نقطه کنترل متمرکز انجام می‌شود. شبکه نرم‌افزاری تعریف‌شده یک رویکرد مکمل برای مجازی‌سازی توابع شبکه[۱] برای مدیریت شبکه است.

شبکه نرم‌افزاری تعریف‌شده نمای متمرکزی از شبکه را ارائه می‌دهد و یک کنترل‌کننده در شبکه نرم‌افزاری تعریف‌شده مانند «مغز» شبکه عمل می‌کند. در این شبکه با استفاده از لایه‌های مجازی، سوییچ‌های مجازی، کنترلر مرکزی، استانداردهای ارتباطی و رابط‌های سطح بالا سعی می‌کنند برخی از کارهای کنترلی و مدیریتی شبکه را در لایه‌های بالاتر به‌صورت نرم‌افزاری انجام دهند. به زبان دیگر شبکه نرم‌افزاری تعریف‌شده وابستگی به سخت‌افزار را کاهش داده و قابلیت‌های نرم‌افزاری و هوشمندی شبکه را افزایش می‌دهد. از این رهگذر سازمان‌ها و اپراتورها می‌توانند اقدام به برنامه‌ریزی و برنامه‌نویسی برای شبکه خود کرده و قابلیت‌های سفارشی و اختصاصی را به وجود آورند که نتیجه آن ارائه سرویس‌های جدیدی برای مشتریان است. یکی از مشهورترین پروتکل‌های مورداستفاده در کنترلرهای شبکه نرم‌افزاری تعریف‌شده پروتکل «اوپن فلو[۲]» است.

محدودیت در فناوری شبکه‌هایی که اکنون استفاده می‌شود، همچنین پیچیدگی، وابستگی به فروشنده و هزینه‌های بالا ازجمله مشکلاتی است که با آن‌ها مواجه هستیم؛ درحالی‌که شبکه نرم‌افزاری تعریف‌شده متمرکز، به‌راحتی قابلیت تغییر متناسب با نیاز کسب‌وکار را دارد. این موضوع باعث می‌شود هزینه‌ها کاهش یابد و همچنین انعطاف‌پذیری و نوآوری بیشتری را برای شبکه فراهم می‌آورد.

مجازی‌سازی شبکه[۳] بخشی از حرکت به سمت شبکه‌های شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه است که با جدا نگه‌داشتن صفحه کنترل از صفحه انتقال، شبکه را از سخت‌افزار جدا می‌کند و یک شبکه مجازی را در بالای شبکه فیزیکی اجرا می‌کند. این کار باعث ایجاد یک سیستم پویاتر می‌شود که از یک صفحه مرکزی قابل‌کنترل تشکیل‌شده است و نیاز به تنظیمات دستی را کاهش می‌دهد. از شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی شبکه می‌توان در شبکه‌های ابری، 5G و... استفاده کرد.

[۱] NFV
[۲] Open Flow
[۳] NV: NETWORK VIRTUALIZATION

۱-۱-۱ تاریخچه شبکه نرم‌افزاری تعریف‌شده

گروه IETF با بررسی روش‌های مختلف برای جداسازی صفحه کنترل از صفحه داده، در سال ۲۰۰۴ یک استاندارد با عنوان «جداسازی صفحه انتقال و کنترل» به همراه معماری "SOFT ROUTER"پیشنهاد کرد. هدف اصلی شبکه نرم‌افزاری تعریف‌شده جداسازی بخش کنترل از سوئیچ‌ها و کنترل خارجی داده‌ها از طریق یک‌نهاد منطقی نرم‌افزاری به نام کنترلر است. این ایده در سال ۲۰۰۴ مطرح شد و در سال ۲۰۱۰ شتاب گرفت. در سال ۲۰۱۱ با تشکیل بنیاد شبکه‌های باز[۱] و عضویت بیش از هشتاد شرکت بزرگ صنعت شبکه در آن و تدوین استاندارد «اوپن فلو» وارد فاز جدیدی شد. اولین محصولات شبکه نرم‌افزاری تعریف‌شده در سال ۲۰۱۲ وارد بازار شدند. پیش‌بینی می‌شود این نوع شبکه‌ها کم‌کم جایگزین شبکه‌های سنتی مبتنی بر اترنت وTCP/IP شوند.

۱-۱-۲ شبکه‌های SDN و شبکه‌های OSI

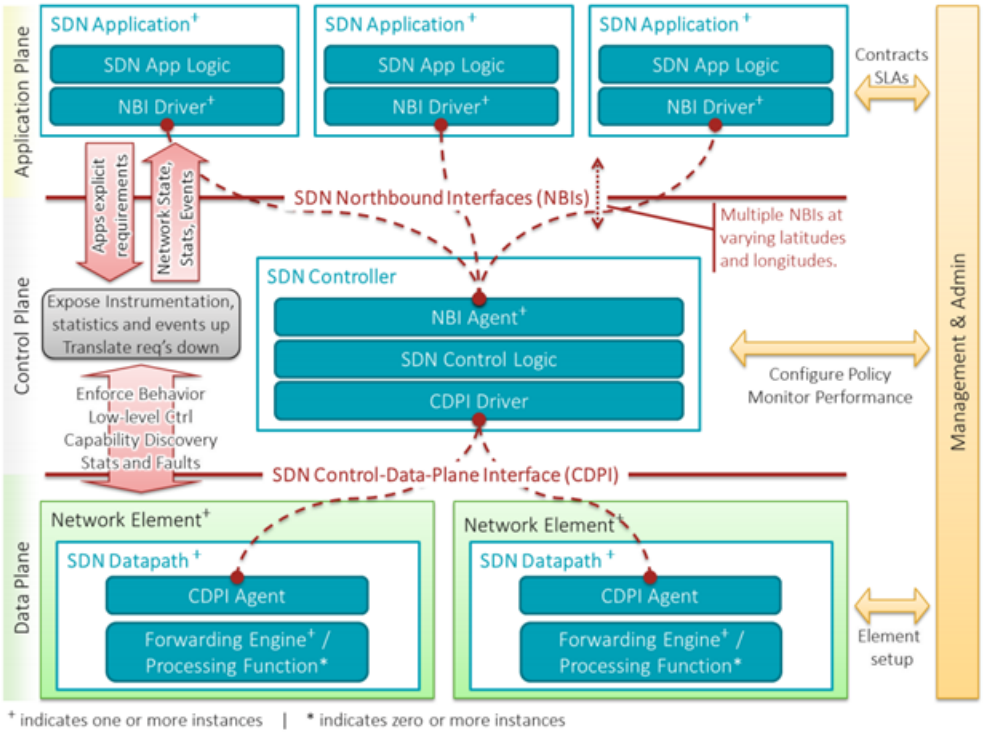
در سال ۱۹۹۵ شبکه نرم‌افزاری تعریف‌شده پس از انتشار جاوا توسط سان مایکروسیستمز شروع به کار کرد. شبکه‌های قدیمی مبتنی بر سوئیچ هستند. سوئیچ یک دستگاه سخت‌افزاری شبکه است که ارتباط بین دستگاه‌های مختلف در یک شبکه خانگی یا شرکتی را تأمین می‌کند. عملکرد سوئیچ به این صورت است که پیام را از یک دستگاه متصل به شبکه دریافت می‌کند و با شناسایی دستگاه گیرنده پیام را فقط برای همان دستگاه مقصد ارسال می‌کند.

در شبکه OSI هفت لایه داریم که در لایه ۲ (دیتا لینک) و لایه ۳ (نتورک) از سوئیچ استفاده می‌شود. سوییچ در شبکه ابزاری است برای ایجاد اتصال میان سیستم‌ها و دیوایس‌های مختلفی که می‌خواهید بین آن‌ها در شبکه‌های محلی مثل شبکه‌های خانگی و شرکتی تعامل وجود داشته باشد. این اتصالات معمولاً با کابل برقرار می‌شود. کارکرد سوییچ به این صورت است که اگر یک دستگاه قصد ارسال اطلاعات برای یک دستگاه دیگر را دارد بتواند مقصد آن اطلاعات را تشخیص دهد. سوییچ‌های لایه ۲ از مک آدرس دستگاه (موجود در فریم‌های پیام دریافتی) برای مشخص کردن دستگاه هدف استفاده می‌کنند و از پورت مناسب اطلاعات را برای دستگاه مقصد ارسال می‌کنند. سوییچ‌های لایه ۳ نیز همین عملکرد را ارائه می‌دهند اما در کنار این قابلیت‌ها، سوییچ‌های لایه ۳ توانایی مسیریابی اطلاعات میان vlan و subnet های موجود در آن شبکه را دارند. یکی از مشکلات استفاده از سوییچ‌ها این است که در سوییچ‌ها با تمام بسته‌ها به یک صورت رفتار می‌شود.

[۱] ONF: Open Networking Foundation

۲-۱ مدل مرجع شبکه نرم‌افزاری تعریف‌شده

شبکه نرم‌افزاری تعریف‌شده بسیار ساده، مؤثر و قابل‌فهم است؛ اما برای داشتن جزییات بیشتر نیاز به داشتن یک مدل با جزییات کامل‌تر به‌عنوان مدل مرجع است. شکل زیر مدل ارائه‌شده در سند بنیاد شبکه‌های باز است.



شکل ۱-۱: مدل مرجع شبکه نرم‌افزاری تعریف‌شده

در نگاه اول، در این مدل سه لایه اصلی مدل شبکه نرم‌افزاری تعریف‌شده دیده می‌شود. این سه لایه عبارت‌اند از:

- در لایه پایین یا ارسال داده، المان‌های ارسال داده در شبکه نرم‌افزاری تعریف‌شده دیده می‌شوند. به این ابزارها سوییچ شبکه نرم‌افزاری تعریف‌شده نیز گفته می‌شود. در هر یک از این المان‌های شبکه یک بخش به نام «واسط لایه ارسال داده با لایه کنترل» یا به‌اختصار «واسط لایه کنترل» وجود دارد. وظیفه این بخش ارتباط با لایه کنترل شامل ارسال بسته‌های نمایه هر جریان، ارسال داده‌های مدیریتی لازم و دریافت وضعیت و داده‌های مدیریتی از کنترلر است. همچنین مسیر کنترلی بخش‌های دیگر المان شبکه یا سوییچ شبکه نرم‌افزاری تعریف‌شده است. در سوییچ شبکه نرم‌افزاری تعریف‌شده بخشی وجود دارد که یک جدول ارسال و پردازش داده را ایجاد می‌کند تا مشخص شود برای هر جریان داده چه عملیاتی باید انجام شود. قسمت دیگر سوییچ شبکه نرم‌افزاری تعریف‌شده بخش ارسال و پردازش داده است. این قسمت وظیفه ارسال و یا

پردازش داده بر اساس سیاست‌های کنترلر و یا لایه‌های بالاتر را دارد. این سیاست‌ها از طریق جداول ارسال و پردازش داده در اختیار این بخش قرار داده می‌شوند.

- در لایه وسط یا لایه کنترل، المان کنترلر (کنترلرها) دیده می‌شود. به کنترلر، سیستم‌عامل شبکه نیز گفته می‌شود. در هر کنترلر چند بخش اصلی و کلی وجود دارد. طبیعتاً یک کنترلر بر اساس معماری می‌تواند بخش‌های جزئی بسیار بیشتری داشته باشد. بخش راه‌انداز «واسط لایه ارسال داده با لایه کنترل» وظیفه ارسال و دریافت بسته‌ها باواسط ارسال داده در سوییچ شبکه نرم‌افزاری تعریف‌شده مطابق آنچه قبلاً گفته شد را دارد. در طرف دیگر بخش واسط شمالی وجود دارد. این واسط با برنامه‌های کاربردی که در لایه کاربر قرار دارند از طریق واسطه‌ای استاندارد در ارتباط است. بخش اصلی کنترلر که عملیات منطقی کنترلر را انجام می‌دهد با عملیات منطقی کنترل‌کننده[1] نشان داده شده است.
- در بالاترین لایه، لایه کاربر قرار دارد. این لایه شامل برنامه‌های کاربری مختلف است. در هر برنامه کاربری دو بخش اصلی وجود دارد. بخش برنامه اصلی و بخش «راه‌انداز واسط شمالی». بخش اصلی درواقع بخش قابل‌برنامه‌ریزی کنترلر است. این بخش، رفتار موردنظر را از طریق برنامه با استفاده از بخش «راه‌انداز واسط شمالی» به کنترلر دیکته کرده و همچنین اطلاعات موردنیاز را از طریق این واسط از شبکه دریافت می‌کند. کنترلر دیدگاه خلاصه‌شده شبکه نرم‌افزاری تعریف‌شده را از طریق همین واسط در اختیار برنامه‌های کاربری قرار می‌دهد. ازآنجاکه هر برنامه ممکن است معماری خاص خود را داشته باشد، معماری این لایه می‌تواند بسته به نرم‌افزارهای کاربری متفاوت باشد. به زبان دیگر این بخش «قسمت قابل‌تعریف با نرم‌افزار» یا لایه قابل‌برنامه‌ریزی است؛ بنابراین هیچ ساختار از پیش تعریف‌شده‌ای برای نوشتن برنامه‌ها در این لایه تعیین نشده است. مسئله اصلی در این لایه پروتکل، ارتباط استاندارد بین لایه کنترل و لایه کاربر است. بعضی از شرکت‌هایی که کنترلر اختصاصی خود را ارائه می‌کنند، پروتکل اختصاصی خود را برای این لایه ارائه کرده‌اند؛ اما پروتکل‌های استاندارد ارتباطی بین لایه کنترل و لایه کاربر، پروتکل‌های REST و JAVA هستند.

لایه دیگری که در شکل به‌موازات سه لایه مذکور مشاهده می‌شود، لایه مدیریت است. هرچند بسیاری از توابعِ مدیریتِ شبکهِ سنتی را می‌توان از طریق لایه کنترل و کاربردهای خاص مدیریت شبکه فراهم نمود، برخی توابع مدیریتی خاص وجود دارد که باید به‌صورت مجزا فراهم شود. در لایه ارسال داده لازم است که راه‌اندازی اولیه عناصر شبکه انجام‌شده، بخش‌هایی که توسط شبکه نرم‌افزاری تعریف‌شده کنترل خواهد شد مشخص‌شده و کنترلر شبکه نرم‌افزاری تعریف‌شده برای آن‌ها پیکربندی می‌شود. در لایه

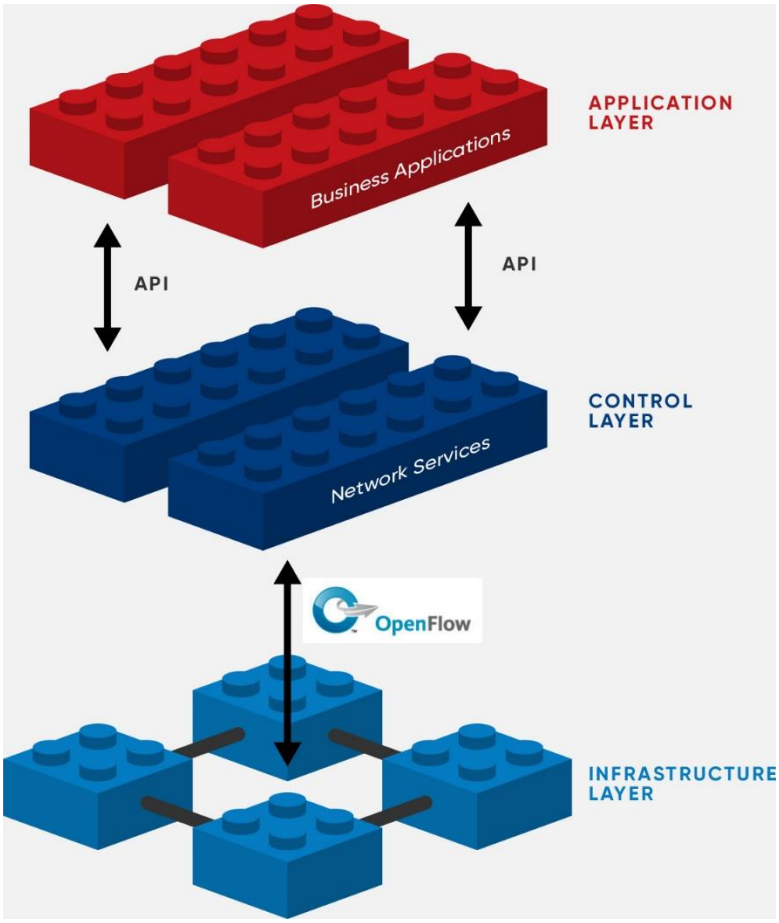
[1] SDN Control Logic

کنترل لازم است که سیاست‌هایی که مشخص‌کننده قلمرو تحت کنترل هر کاربرد شبکه نرم‌افزاری تعریف‌شده است، پیکربندی‌شده و کارایی سیستم پایش شود. در لایه کاربر لازم است که قراردادها و SLA ها پیکربندی شوند. در تمامی سطوح سیستم مدیریت باید موارد امنیتی که امکان ارتباط امن بین توابع توزیع‌شده را فراهم می‌نمایند پیکربندی شوند.

۳-۱ معماری شبکه‌های شبکه نرم‌افزاری تعریف‌شده

مطابق تعریف بنیادِ شبکه‌هایِ باز شبکه‌های نرم‌افزار محور یا شبکه نرم‌افزاری تعریف‌شده یک نوع از معماری شبکه‌ای است که سطوح داده و کنترل را از یکدیگر جدا می‌کند. این امر با قرار دادن سطح کنترل (فراهم‌کننده سیاست‌ها و هوشمندی شبکه) در سرور یا برنامه‌ای که کنترلر نام دارد امکان‌پذیر است. برنامه‌های کاربردی هم در بالای سطح کنترل در سطح کاربرد (Application Layer)قرار می‌گیرند.

در معماری شبکه‌های شبکه نرم‌افزاری تعریف‌شده سعی شده است که توانایی و هوشمندی شبکه‌های کامپیوتری بیشتر شود. این کار با جداسازی بخش کنترل داده‌ها از سوییچ و روتر سخت‌افزاری و تمرکز آن‌ها روی لایه‌های نرم‌افزاری شبکه به‌صورت مجازی انجام می‌شود. یک کنترلرِ نرم‌افزاریِ متمرکز که دیدی از کل شبکه را در اختیار داشته باشد، بسیار مؤثرتر و هوشمندتر از بخش‌های کنترلی روترها و سوییچ‌ها که پراکنده هستند عمل خواهد کرد. با یک کنترلر متمرکز، قابلیت‌هایی مانند برنامه‌ریزی، مقیاس‌پذیری، انعطاف‌پذیری، خودکارسازی و توسعه نرم‌افزاری به شبکه اضافه می‌شود.



شکل ۲-۱: معماری شبکه‌های نرم‌افزاری تعریف‌شده

در معماری شبکه نرم‌افزاری تعریف‌شده ۳ لایه داریم:

- اپلیکیشن[۱]: شامل برنامه‌ها و سرویس‌های در حال اجرا در شبکه
- کنترل[۲]: کنترل شبکه نرم‌افزاری تعریف‌شده یا به عبارتی «مغز» شبکه
- زیرساخت[۳]: زیرساخت شبکه؛ به عبارتی سوییچ‌ها و روترها و سخت‌افزارهای پشتیبانی

[۱] Application
[۲] Control
[۳] Infrastructure

برای ارتباط بین این ۳ لایه از واسط شمالی[۱] و واسط جنوبی[۲] (واسط‌های[۳] سطح بالا و سطح پایین) استفاده می‌کنند که واسط شمالی وظیفه برقراری ارتباط بین لایه‌های زیرساخت و کنترل را بر عهده دارد و واسط جنوبی ارتباط بین لایه‌های اپلیکیشن و کنترل را ایجاد می‌کند.

واسط‌های شمالی: برای برقراری ارتباط میان کنترلر، سرویس‌ها و برنامه‌های کاربردی در حال اجرای در شبکه استفاده می‌شود. با استفاده از این واسط وضعیت زیرساخت شبکه مشخص می‌شود تا بدانند چه منابعی در دسترس هستند.

علاوه بر این، به‌صورت خودکار می‌تواند اطمینان حاصل کند که ترافیک برنامه بر اساس محدودیت‌ها و قوانینی که ادمین ایجاد کرده مسیریابی شده است یا خیر؟

این واسط با لایه کنترل صحبت می‌کند و می‌گوید که برنامه به چه منابعی نیاز دارد و مقصد آن کجاست.

لایه کنترل نحوه دسترسی به منابع را در شبکه تنظیم می‌کند و از هوش خود برای پیدا کردن مسیر بهینه‌ای که کمترین تأخیر و بیشترین امنیت را داشته باشد استفاده می‌کند.

واسط‌های جنوبی: توسط بنیاد شبکه‌های باز و بانام اوپن فلو استاندارد شده است. اوپن فلو امکان دسترسی مستقیم و قابلیت تغییر در برنامه ارسال داده که توسط تجهیزات شبکه، نظیر سوئیچ‌ها به‌صورت فیزیکی و مجازی انجام می‌شود را فراهم می‌کند.

این واسط ارتباط بین دولایه دیتا (زیرساخت) و کنترل را ایجاد می‌کند؛ به عبارتی بین لایه کنترل و زیرساخت‌های شبکه مانند روتر و سوییچ ارتباط برقرار می‌کند.

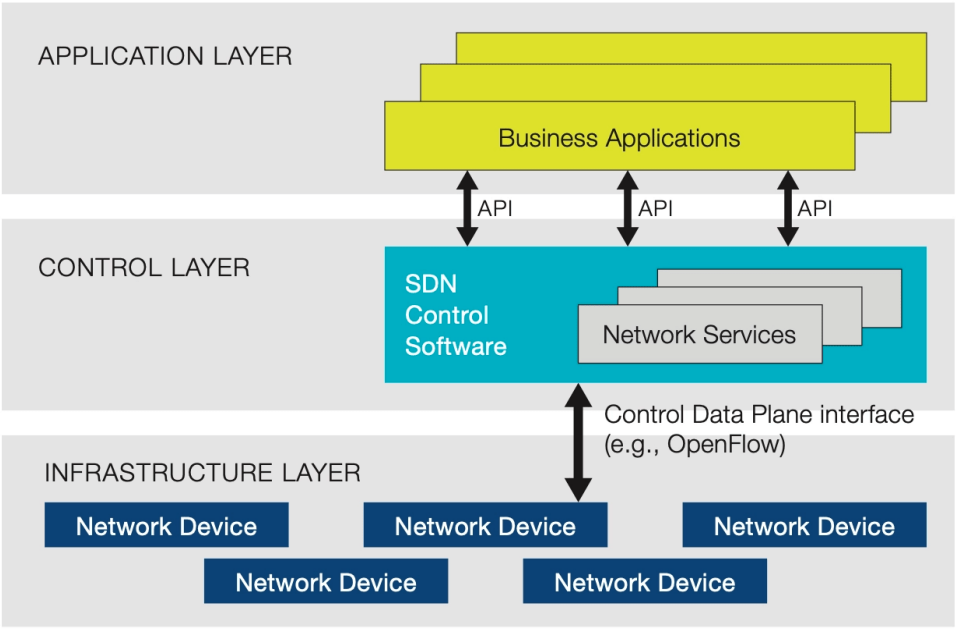
به لایه زیرساخت، زیرساخت شبکه گفته می‌شود، داده‌های برنامه طبق تصمیم لایه کنترل می‌داند که چه مسیری باید طی کند. کنترلر می‌تواند نحوه انتقال داده‌ها و روترها را تغییر دهد. داده‌ها دیگر به دستگاه‌ها و جدول مسیریابی وابسته نیستند تا مسیر داده‌ها مشخص شود، در مقابل هوش مصنوعی تصمیماتی آگاهانه برای انتخاب مسیر بهینه می‌گیرد.

[۱] API Northbound
[۲] API Southbound
[۳] API

۴-۱ کنترلر شبکه نرم‌افزاری تعریف‌شده

کنترل‌کننده شبکه نرم‌افزاری تعریف‌شده[۱]: یک نرم‌افزار کنترلی است که یک نمایش کامل و متمرکز از شبکه را در اختیار شما قرار می‌دهد. ادمین شبکه از کنترلر برای بررسی زیرساخت صفحه انتقال و هدایت ترافیک و همچنین پیاده‌سازی قوانین رفتار شبکه استفاده می‌کند.

ادمین شبکه قوانینی را ایجاد می‌کند تا به‌طور یکنواخت روی چندین نود[۲] شبکه اعمال شود. تعریف قوانین، مشخص می‌کند که در زمان ترافیک شبکه، سطح دسترسی به شبکه، چه میزان از منابع استفاده شود و همچنین اولویت‌ها را مشخص می‌کند. برخورداری از دید متمرکز در شبکه و سیاست‌های موجود، مدیریت شبکه را یکدست‌تر و یکنواخت‌تر می‌کند.



شکل ۳-۱: ارتباط لایه‌ها از طریق واسط

لایه‌های برنامه، کنترل و زیرساخت‌ها در شبکه نرم‌افزاری تعریف‌شده جداگانه نگه‌داشته شده و از طریق واسط‌ها ارتباط برقرار می‌کنند.

[۱] SDN controllers
[۲] node

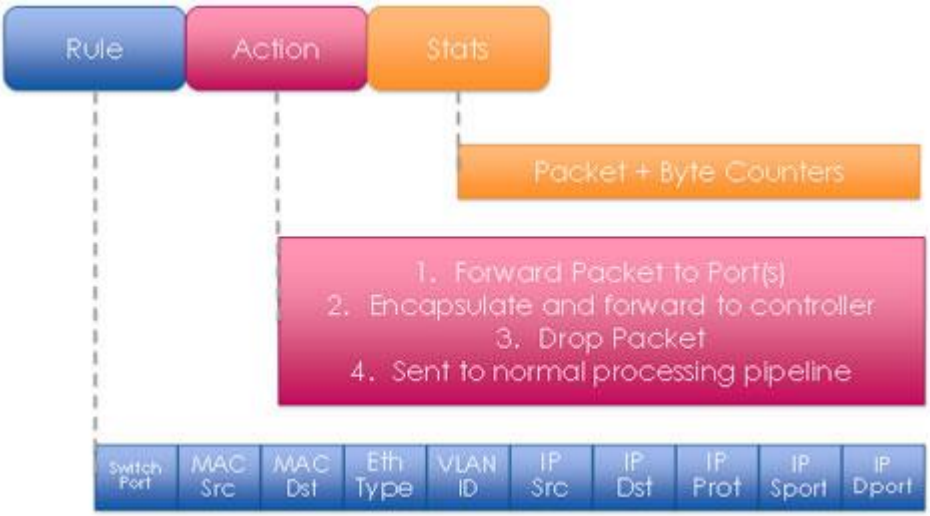
۵-۱ پروتکل اوپن فلو

پروتکل اوپن فلو یکی از اولین استانداردهای شبکه نرم‌افزاری تعریف‌شده محسوب می‌شود. این پروتکل ارتباطی را در محیط‌های شبکه نرم‌افزاری تعریف‌شده تعریف می‌کند که به کنترلر شبکه نرم‌افزاری تعریف‌شده اجازه می‌دهد تا مستقیماً با بخش انتقال دستگاه‌های شبکه، مانند سوییچ و روتر ازنظر فیزیکی و مجازی ارتباط برقرار کند تا راحت‌تر بتوانند با تغییرات سازگار شوند.

همان‌طور که در قبل هم اشاره شد کنترل‌کننده شبکه نرم‌افزاری تعریف‌شده در جایگاه مغز شبکه قرار دارد و انتقال دیتا به سوییچ و روترها از طریق واسطهای جنوبی انجام می‌شود. لازم به ذکر است واسطهای جنوبی بین لایه‌های اپلیکیشن و کنترل ارتباط برقرار می‌کند.

برای کار در محیط **اوپن فلو**، هر د ستگاهی که می‌خواهد با یک کنترلر شبکه نرم‌افزاری تعریف شده ارتباط بگیرد باید از پروتکل اوپن فلو پشتیبانی کند. با استفاده از رابط[۱]، کنترلر شبکه نرم‌افزاری تعریف‌شده تغییراتی در جدول جریان عبوری از سوییچ/روتر ایجاد می‌کند و به ادمین شبکه اجازه می‌دهد تا ترافیک شبکه را تقسیم کند، جریان‌های عبوری را بهینه کند و تنظیمات جدیدی را ایجاد کند.

جریان‌های ورودی در محیط اوپن فلو:



[۱] Interface

شکل ۴-۱: جریان‌های ورودی در محیط اوپن فلو

۱-۵-۱ مزایای پروتکل اوپن فلو

- قابلیت برنامه‌نویسی[۱]
- نوآوری دارد و متفاوت است.
- سرعت بالایی در ارائه خدمات و ویژگی‌های جدید دارد.
- هوش مرکزی[۲]
- سادگی
- عملکرد بهینه
- مدیریت ساده قوانین و سیاست‌ها policy

مهم‌ترین مزیت این پروتکل:

- جداسازی سخت‌افزار و نرم‌افزار
- جداسازی بخش کنترل و انتقال
- تنظیمات فیزیکی و منطقی

۱-۵-۲ تاریخچه پروتکل اوپن فلو

مفهوم اصلی اوپن فلو برای اولین بار در دانشگاه استنفورد در سال ۲۰۰۸ بیان شد. در دسامبر ۲۰۰۹ ورژن ۱,۰ از پروتکل اوپن فلو منتشر شد.

از زمان انتشار چندین شرکت مانند: OpenDaylight Project از این پروتکل پشتیبانی می‌شود. حتی کنترلرهای OpenDaylight ارائه می‌شوند. شرکت‌های دیگری مانند: Cisco و Brocade کنترلرهایی باقابلیت **اوپن فلو** را با Cisco XNC و Brocade Vyatta Controller ارائه می‌دهند.

[۱] Programmability

[۲] Centralized Intelligence

۱-۶ استانداردسازی شبکه‌های SDN و NFV

شبکه‌های نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه به‌صورت فزاینده‌ای در شبکه‌های ارتباطی ازجمله شبکه‌های تلفن، شبکه‌های تلفن همراه و شبکه‌های رایانه‌ای استفاده می‌شود. از ویژگی‌های اصلی شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه می‌توان به ماژولار بودن سخت‌افزار و نرم‌افزار، مجازی‌سازی در کلیه سطوح شبکه، تعیین سطح کاربری برای کاربران، فراهم کردن امکان ایزوله برای کنترل بیشتر، کنترل مرکزیت و قابلیت برنامه‌نویسی اشاره کرد. چندین جنبه فنی وجود دارد که شبکه‌های شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه را تحت تأثیر قرار می‌دهد. برای پذیرش سریع‌تر شبکه‌های در حال تکامل با شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه باید استانداردهایی تعریف شود که سازمان‌هایی مثل 3GPP، ONF، IETF، ETSI وIEEE در استانداردسازی بخش‌های مختلف این استانداردها نقش دارند.

۱-۶-۱ استاندارد برای شبکه‌های هم‌پوشان[۱]

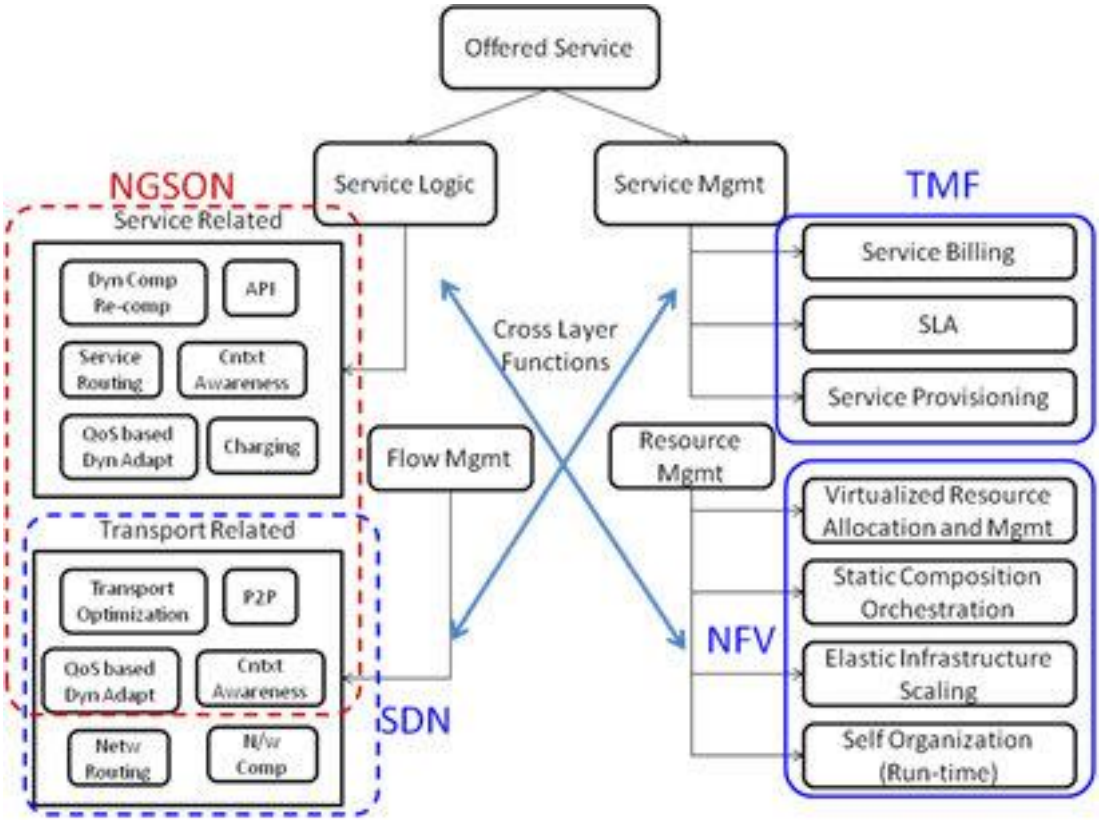
NGSON[۲] این شبکه در حال تلاش برای استانداردسازی یک اکوسیستم خدماتی به نفع اپراتورهای شبکه، ارائه‌دهندگان خدمات، ارائه‌دهندگان محتوا و کاربران نهایی است. همچنین پروتکل‌هایی برای ترکیب خدمات، تحویل محتوا و مدیریت خودکار سازمان تهیه می‌کند. این استاندارد یک چارچوب مرجع در شبکه‌های پوشش[۳] با ارائه خدمات مبتنی برip ، برای تحویل خدمات مشترک و مشتری محور استانداردسازی کرده است. این خدمات شامل قابلیت‌هایی چون همگام‌سازی، مدیریت خودکار شبکه، مسیریابی پیشرفته و نقشه است.

شکل زیر نمونه‌ای از خدمات ارائه‌شده را نشان می‌دهد:

[۱] Service Overlay Networks

[۲] Next Generation Service Overlay Networks

[۳] Overlay



شکل ۵-۱: خدمات NGSON

NGSON قصد گسترش و تسریع خدمات و برنامه‌های شبکه نرم‌افزاری تعریف‌شده را دارد. همچنین به دنبال پیدا کردن یک روش کارآمدتر برای ارائه آن‌ها از طریق طراحی معماری یک اکوسیستم است.

۲-۶-۱ استانداردسازی توابع مجازی شبکه/شبکه نرم‌افزاری تعریف‌شده[۱]

درصورتی‌که مدل‌سازی انجام نشود، کیفیت، امنیت، عملکرد، قابلیت اطمینان، هزینه‌های عملیاتی و نگهداری شبکه افزایش می‌یابد. تعریف‌های متفاوتی از کیفیت وجود دارد که نیازمند توسعه استانداردهایی از قبیل امنیت، افزایش عملکرد برای خدمات ارائه‌شده در شبکه نرم‌افزاری تعریف‌شده/مجازی‌سازی توابع شبکه، ایجاد دیتابیس، شناسایی سطح قابلیت اطمینان، کیفیت خدمات هستند.

۱-۲-۶-۱ تعریفی از امنیت در محیط‌های مجازی‌سازی

تلاش برای استانداردسازی مدل‌های امنیتی، اصطلاحات، تجزیه‌وتحلیل اجزای اصلی شبکه نرم‌افزاری تعریف‌شده/مجازی‌سازی توابع شبکه، یکپارچگی و در دسترس بودن است.

[۱] sdn/nfv

۱-۶-۲-۲ تعریفی از عملکرد محیط‌های مجازی‌سازی

تلاش برای استانداردسازی مدل‌های عملکرد بهتر، تجزیه‌وتحلیل برای فعال کردن عملیات سیستم بهینه و تحویل خدمات در زیرساخت‌های شبکه نرم‌افزاری تعریف‌شده-مجازی سازی توابع شبکه

۱-۶-۲-۳ تعریفی از قابلیت اطمینان برای محیط‌های مجازی‌سازی

تلاش برای استانداردسازی مدل‌های قابلیت اطمینان، تجزیه‌وتحلیل برای فعال کردن عملیات بهینه سیستم و تحویل خدمات در زیرساخت‌های شبکه نرم‌افزاری تعریف‌شده/مجازی‌سازی توابع شبکه

۱-۶-۳ ارتباطات کوانتومی شبکه نرم‌افزاری تعریف‌شده

هدف از این پروژه تعریف یک رابط کلاسیک با دستگاه‌های ارتباطی کوانتومی است که به این دستگاه‌ها اجازه پیکربندی مجدد پروتکل‌ها و اندازه‌گیری را می‌دهد. این رابط پیکربندی نقاط نهایی کوانتومی را در یک شبکه ارتباطی به‌منظور ایجاد پویایی، اصلاح یا حذف پروتکل‌ها و یا کاربردهای کوانتومی فراهم خواهد کرد. طراحی پروتکل یکپارچگی آینده را با شبکه‌سازی تعریف‌شده توسط نرم‌افزار و بنیاد شبکه‌سازی باز تسهیل خواهید کرد.

۱-۷ مزایای شبکه نرم‌افزاری تعریف‌شده

شبکه نرم‌افزاری تعریف‌شده شبکه‌ای متمرکز و قابل‌برنامه‌ریزی را ارائه می‌دهد که می‌تواند منابع شبکه را به‌طور پویا تأمین کند تا نیازهای متغیر را برطرف کند. همچنین مزایای فنی و تجاری زیر را ارائه می‌دهد:

۱-۷-۱ قابلیت برنامه‌نویسی مستقیم

شبکه‌های نرم‌افزاری تعریف‌شده قابلیت برنامه‌نویسی مستقیم را دارد زیرا توابع کنترلی جدا از توابع انتقال هستند به همین دلیل می‌توان با استفاده از منبع‌های اختصاصی مانند OpenStack, Puppet, Salt, Ansible, Chef پیکربندی یا تنظیم شود.

۱-۷-۲ مدیریت متمرکز

بخش منطقی هوش شبکه، روی نرم‌افزار کنترلی شبکه نرم‌افزاری تعریف‌شده متمرکزشده است. وظیفه این بخش حفظ نمای کلی از شبکه برای برنامه‌ها است. همچنین موتور قوانین شبکه نرم‌افزاری تعریف‌شده به‌عنوان یک سوییچ مستقل منطقی به نظر می‌رسد.

۱-۷-۳ کاهش هزینه

شبکه نرم‌افزاری تعریف‌شده به صورتی ذاتی هزینه‌ها را کاهش می‌دهد و نیازی به خرید سخت‌افزارهایی مثل ASIC_BASED نیست.

بیشتر سوییچ‌های داخل بازار از قابلیت‌های شبکه نرم‌افزاری تعریف‌شده و نرم‌افزارهایی مانند اوپن فلو پشتیبانی می‌کنند. اگر در زیرساخت شبکه از سوییچ‌هایی باقابلیت شبکه نرم‌افزاری تعریف‌شده استفاده‌شده باشد، چه این شبکه یک مرکز داده[۱] باشد یا یک شبکه خانگی تفاوتی ندارد، فقط نیاز به فعال‌سازی دارد.

۱-۷-۴ بروز رسانی

امکان به‌روزرسانی نرم‌افزارهای شبکه به این معنی است که نیازی به جداسازی و جایگزین کردن کل زیرساخت شبکه برای تغییر کسب‌وکار نیست. می‌توان قوانین را به‌صورت یکنواخت در شبکه گسترش داد تا احتمال بروز خطای انسانی هنگام به‌روزرسانی شبکه کاهش یابد.

۱-۷-۵ انعطاف‌پذیری

شبکه نرم‌افزاری تعریف‌شده می‌تواند به سازمان‌ها کمک کند تا به‌سرعت برنامه‌های جدید، خدمات و زیرساخت‌های جدید خود را مستقر کرده و به‌راحتی اهداف و نیازهای تجاری خود را تغییر دهند، زیرا هر زمان که چیزی جدید ایجاد شود، یک به‌روزرسانی ساده آن را در سراسر شبکه مستقر می‌کند.

۱-۸ چالش‌های شبکه نرم‌افزاری تعریف‌شده

شبکه نرم‌افزاری تعریف‌شده بدون چالش و مسئله نیست مانند همه حوزه‌های it این بخش هم مشکلاتی دارد:

۱-۸-۱ خطرات امنیتی مدیریت متمرکز

مدیریت متمرکز جز مزیت شبکه‌های نرم‌افزاری تعریف‌شده است ولی یک خطر امنیتی نیز محسوب می‌شود. بخش مدیریت متمرکز[۲] یک مرکز حساس هست که اگر موردحمله قرار گیرد کل شبکه تحت تأثیر قرار خواهد گرفت.

[۱] Data Center

[۲] centralized management

۱-۸-۲ گلوگاه کنترل‌کننده

درصورتی‌که تنها یک نمونه کنترلر شبکه نرم‌افزاری تعریف‌شده وجود داشته باشد، می‌تواند به یک تنگنا برای یک شبکه، با حجم زیاد ترافیک، روتر و سوئیچ تبدیل شود.

۱-۸-۳ هیچ استانداردی برای واسطها پذیرفته‌نشده

افراد مختلفی در حال ایجاد واسطهای متفاوتی برای کنترل‌کننده‌های شبکه نرم‌افزاری تعریف‌شده هستند. این امر باعث می‌شود توسعه نرم‌افزاری سخت شود زیرا به‌منظور تعامل با کنترل‌های مختلف، توسعه‌دهندگان باید چند ورژن را آماده کنند.

۱-۸-۴ رابط کاربری

نرم‌افزار شبکه نرم‌افزاری تعریف‌شده امکان devOps دارد که شامل: تست خودکار، تغییر توسعه اپلیکیشن و مجازی‌سازی بخش‌های از شبکه است.

در ساختمان‌های هوشمند می‌توانند از شبکه نرم‌افزاری تعریف‌شده برای مدیریت شبکه بی‌سیم که تمام دستگاه‌های داخل ساختمان را به هم متصل می‌کند استفاده کنند.

از اصول مجازی‌سازی که شبکه نرم‌افزاری تعریف‌شده معرفی کرده است می‌توانیم در v2x (ارتباط وسیله نقلیه) استفاده کرد.

نرم‌افزار شبکه نرم‌افزاری تعریف‌شده معمولاً فقط یک مرکز داده دارد ولی می‌توانیم آن را به‌اندازه کل کسب‌وکار گسترش دهید. با استفاده از فناوری شبکه نرم‌افزاری تعریف‌شده می‌توانیم ارتباطات بین شبکه‌های بی‌سیم، باسیم و اترنت را ساده کرده و به‌صورت مرکزی مدیریت کرد و خدمات‌رسانی را به‌صورت خودکار انجام داد.

۱-۹ تأثیر شبکه‌های تعریف‌شده نرم‌افزاری بر فناوری‌های شبکه‌ای

مجازی‌سازی که شبکه نرم‌افزاری تعریف‌شده مطرح شد حرکتی روبه‌جلو در فناوری شبکه است. افرادی که شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه آموخته‌اند پیشرفت‌های زیادی کردند. Sd-wan و sd-branch و secure access service edge نمونه‌های از این شبکه‌ها هستند. در شبکه‌های 5G کنترل شبکه از طریق یک هسته مرکزی شبیه به شبکه نرم‌افزاری تعریف‌شده صورت می‌گیرد و از قوانین شبکه نرم‌افزاری تعریف‌شده یاد می‌گیرد.

۱-۱۰ شبکه نرم‌افزاری تعریف‌شده در مقایسه با مجازی‌سازی توابع شبکه

مجازی‌سازی توابع شبکه یکی از اجزای مجازی‌سازی شبکه است. این توابع می‌توانند میان بار شبکه، تعادل ایجاد کرده و قابلیت تنظیمات متفاوت از فایروال تا مسیریابی هستند.

شبکه نرم‌افزاری تعریف‌شده صفحه‌های کنترل و دیتا را از هم جدا کرده و مرکز کنترل دارد. مرکز کنترل این اجازه را می‌دهد تا به‌صورت خودکار فعالیت کند و باعث ایجاد شبکه باقابلیت برنامه‌ریزی می‌شود.

در اصل مجازی‌سازی توابع شبکه اجزای شبکه را مجازی می‌کند و شبکه نرم‌افزاری تعریف‌شده روی کنترل مؤلفه متمرکز می‌شود. شبکه نرم‌افزاری تعریف‌شده و مجازی‌سازی توابع شبکه به یکدیگر وابسته نیستند.

۱-۱۱ شبکه نرم‌افزاری تعریف‌شده و sd-wan

شبکه گسترده تعریف‌شده توسط نرم‌افزار (SD-WAN)[۱] توسعه‌یافته شبکه نرم‌افزاری تعریف‌شده است. Sd-wan یک منطقه جغرافیایی گسترده را پوشش می‌دهد و طبق گفته IEEE این فناوری به شرکت‌ها اجازه می‌دهد تا کلیه شبکه‌های خود را در یک منطقه جغرافیایی به هم متصل کنند بدون اینکه فاصله بین گره‌ها مهم باشد.

برای مثال: تمام شبکه، تمام شعب و مرکز داده می‌تواند از طریق sd-wan به هم متصل شوند و از طریق یک مرکز کنترل شوند. به دلیل این ویژگی سازمان‌ها و شرکت‌های زیادی از sd-wan استفاده می‌کنند. یکی دیگر از مزیت‌های sd-wan این است که رمزنگاری انتها به انتها[۲] است که باعث افزایش امنیت شبکه می‌شود.

Sd-wan برای ایجاد شبکه محلی[۳] نیز کاربرد دارد.

۱-۱۲ بررسی شبکه نرم‌افزاری تعریف‌شده در گوگل

ساختار شبکه wan در گوگل به دو بخش تقسیم می‌شود:

۱- شبکه‌ای که اطلاعات کاربران را منتقل می‌کند (شبکه خارجی)

[۱] Software-defined wide-area network (SD-WAN)

[2] end to end

[۳] lan

۲- شبکه‌ای که دیتا را بین دیتاسنترها جابه‌جا می‌کند (شبکه داخلی)

این دو شبکه زیرساختی کاملاً متفاوت دارند که در یک شبکه به نام g-scale قرار داده‌شده‌اند و گوگل در آن از شبکه نرم‌افزاری تعریف‌شده استفاده کرده است.

زمانی که گوگل شروع به کاربر روی این شبکه کرد، هیچ دستگاهی وجود نداشت که از پروتکل اوپن فلو پشتیبانی کند و توانایی پشتیبانی از نیاز شبکه را داشته باشد، به همین دلیل گوگل شروع به ساخت سخت‌افزارهای کرد که از اوپن فلو پشتیبانی کند. گوگل چند بخش آماده کرد که در هر یک از این بخش‌ها از چندین سوییچ استفاده می‌کرد و چندین ترابایت پهنای باند داشت.

این بخش‌ها به یکدیگر مرتبط هستند و چندین کنترلر اوپن فلو از طریق پروتکل اوپن فلو با سوییچ‌ها در ارتباط هستند تا اطمینان حاصل کنند که مشکلی در شبکه نیست.

در این شبکه wan، یک مرکز مهندسی ترافیک ساخته شد. این بخش به‌صورت لحظه‌ای اطلاعات توپولوژی و اندازه‌گیری‌ها را از پهنای باند و شبکه اپلیکیشن‌ها و خدمات بررسی می‌کند. با استفاده از این اطلاعات مسیرهای جریان اطلاعات را محاسبه و اعلام می‌کند. سپس با استفاده از اوپن فلو این مسیرها را در سوییچ‌ها اعمال می‌کند. در صورت تغییر مشخصات شبکه یا بروز مشکل، سیستم به‌صورت خودکار مسیر جدید را محاسبه و اعمال می‌کند.

فصل دوم:
امنیت در شبکه‌های نرم‌افزاری تعریف‌شده

۲-۱ امنیت در شبکه‌های نرم‌افزاری تعریف‌شده

در محیط‌های شبکه نرم‌افزاری تعریف‌شده، امنیت شبکه نرم‌افزاری تعریف‌شده باید در همه بخش‌های یک شبکه تعریف‌شده نرم‌افزاری اعمال شود. امنیت شبکه نرم‌افزاری تعریف‌شده باید در معماری ساخته می‌شود. همچنین به‌عنوان یک سرویس برای محافظت از دسترسی، حفظ یکپارچگی و حفظ حریم خصوصی کلیه منابع و اطلاعات متصل ارائه شود.

۲-۱-۱ روند تکامل امنیت در شبکه نرم‌افزاری تعریف‌شده

شبکه نرم‌افزاری تعریف‌شده با مجازی‌سازی توابع امنیتی شروع به کارکرد. این توابع امنیتی روی سخت‌افزارهای عمومی vnf انجام شد. با مجازی‌سازی می‌توان سیستم را از راه دور مدیریت کرد. همچنین پیاده‌سازی و به‌روزرسانی سیستم را آسان‌تر کرد.

با مجازی‌سازی توابع امنیتی، vnf به‌عنوان نرم‌افزار روی دستگاه‌ها قرار گرفت. این نوع دستگاه‌ها که vnf روی آن‌ها اجرا می‌شود می‌توانند عملکردهای سخت‌افزاری که به خاطر آن‌ها جایگزین شده‌اند را انجام دهند.

اخیراً امنیت به‌عنوان کد در گرایش امنیت فناوری اطلاعات[۱] عنوان می‌شود. در این روش به‌جای نوشتن مستقیم توابع امنیتی روی برنامه‌ها یا نرم‌افزارهای شبکه، در vnfبه‌صورت جدا قرار می‌گیرند. امنیت ازلحاظ کد، قسمتی از devops است که اغلب به آن DevSecOps گفته می‌شود؛ که باعث ایجاد رویکردی‌های امنیتی از فرایند توسعه می‌شود و بعداً نمی‌تواند افزوده شود.

۲-۲ تقسیم‌بندی[۲]

در یک شبکه مجازی یا مرکز داده تقسیم‌بندی به محدود کردن محدوده امنیتی کمک می‌کند. اپراتورها می‌توانند از تقسیم‌بندی شبکه برای ایجاد شبکه‌های مجازی برای برنامه‌های مختلف ترافیک استفاده کنند. تقسیم‌بندی شبکه مانع از همگرایی ترافیک یک برنامه با برنامه دیگر می‌شود.

در یک دیتاسنتر، میکروسگمنت بخش‌های مختلف را از هم جدا می‌کند. با استفاده از این میکروسگمنت‌ها می‌توان ترافیک را به‌طور مؤثری کنترل کرد. این بدان معناست که می‌توان آسیب‌پذیری ارتباط بین سرورها و ارتباط بین حجم کاری را در زمان حمله کاهش داد.

[۱] IT Security

[۲] segmentation

۲-۳ لایه‌های امنیتی در معماری شبکه نرم‌افزاری تعریف‌شده

۲-۳-۱ امنیت کنترل‌کننده

به‌عنوان نقطه تصمیم‌گیرنده مرکزی دسترسی به کنترلر شبکه نرم‌افزاری تعریف‌شده باید کنترل شود.

۲-۳-۲ حفاظت از کنترل‌کننده

اگر کنترلر شبکه نرم‌افزاری تعریف‌شده پایین بیاید (برای مثال در اثر حمله DDOS)، شبکه پیش می‌رود؛ این بدان معنا است که در دسترس بودن کنترلر شبکه نرم‌افزاری تعریف‌شده نیاز به حفاظت دارد.

۲-۳-۳ ایجاد اعتماد

محافظت از ارتباطات در سراسر شبکه بسیار مهم است؛ یعنی محافظت از کنترل‌کننده شبکه نرم‌افزاری تعریف‌شده، برنامه‌های بارگذاری شده بر روی آن و دستگاه‌هایی که مدیریت می‌کند. البته اپراتورهای انجام دهنده این کار باید قابل‌اعتماد باشند.

۲-۴ محیط امنیتی

امنیت تعریف‌شده نرم‌افزار با مجازی‌سازی توابع امنیتی به‌منظور جایگزینی سخت‌افزار با نرم‌افزار آغاز شد. آغاز انتقال به نرم‌افزارهای امنیتی به‌عنوان کدی است که در برنامه‌های شبکه نوشته‌شده است. محیط امنیتی مجازی، به مدیران اجازه می‌دهد تا از راه دور نگهداری را از طریق کنترل متمرکز انجام دهند.

۲-۴-۱ تدوین قوانین و سیاست‌ها

مواردی که لازم است سیستم را چک کنید تا مطمئن شوید کنترلر های شبکه نرم‌افزاری تعریف‌شده آنچه شما می‌خواهید را درست انجام می‌دهند.

۲-۴-۲ بررسی حملات و محافظت

زمانی که یک حمله اتفاق می‌افتد شما باید بتوانید اطلاعات را بازیابی کنید و یک گزارش تهیه کنید، سپس در برابر حملات آینده محافظت‌های لازم را انجام دهید.

فراتر از معماری، چگونگی استقرار، مدیریت و کنترل امنیت شبکه نرم‌افزاری تعریف‌شده در یک محیط شبکه نرم‌افزاری تعریف‌شده موردتوجه بسیاری است. برخی معتقدند امنیت به بهترین وجه در شبکه نرم‌افزاری تعریف‌شده قرار دارد و برخی دیگر

احساس می‌کنند که بهترین راه‌حل در سرورها، ذخیره‌سازی و دیگر کامپونت‌های دستگاه‌ها است. صرف‌نظر از این نظرات برای ایجاد محیطی مقیاس‌پذیر، کارآمدتر و ایمن‌تر راه‌حل‌هایی باید طراحی شود:

سادگی: مدیریت، نگهداری و استقرار در محیط شبکه نرم‌افزاری تعریف‌شده

مقرون‌به‌صرفه: باید مطمئن شوید که می‌توانید امنیت را در همه بخش‌ها پیاده کنید.

ایمن: برای محافظت در برابر آخرین تهدیدات پیشرفته و هدفمند نسبت به سازمان شما.

دسته جدیدی برای امنیت در محیط‌های نسل بعدی به نام امنیت تعریف‌شده از نرم‌افزار در حال ظهور است. در این نوع از امنیت با جدا کردن کنترلر امنیتی از پردازش و ارسال سیستم، اجرای امنیت شبکه را تحویل می‌دهد، مشابه روشی که شبکه نرم‌افزاری تعریف‌شده، کنترل شبکه را از انتقال انتزاع می‌کند. درنتیجه یک سیستم توزیع پویا است که عملکرد مجوز امنیت شبکه را مجازی می‌کند، مقیاس‌هایی مانند ماشین‌های مجازی را نشان می‌دهد و به‌عنوان یک سیستم واحد منطقی اداره می‌شود.

SDSec نمونه‌ای از مجازی‌سازی توابع شبکه است. در این نوع از مجازی‌سازی با جداسازی عملکرد شبکه مانند فایروال و کشف نفوذ از وسایل سخت‌افزاری اختصاصی، روش جدیدی برای طراحی، استقرار و مدیریت امنیت شبکه نرم‌افزاری تعریف‌شده ارائه می‌شود؛ بنابراین می‌توان توسط آن کاری کرد که نرم‌افزار این برنامه، برای ادغام و ارائه مؤلفه‌های شبکه موردنیاز جهت پشتیبانی از یک زیرساخت کاملاً مجازی – ازجمله سرورهای مجازی، ذخیره‌سازی و حتی سایر شبکه‌ها– طراحی شود.

۵-۲ ارتقاء امنیت

معماری شبکه نرم‌افزار محور شبکه نرم‌افزاری تعریف‌شده، پتانسیل نوآوری در استفاده از شبکه را مطرح می‌کند. در این نوع معماری ترکیبی از دیدگاه جهانی در سطح شبکه، قابلیت برنامه‌نویسی شبکه، فرایند برداشت هوش از سیستم‌های تشخیص نفوذ و سیستم‌های جلوگیری از نفوذ پشتیبانی می‌شود، برای نمونه با تحلیل و برنامه‌نویسی متمرکز شبکه، دنبال می‌شود. این رویکرد می‌تواند شبکه نرم‌افزار محور شبکه نرم‌افزاری تعریف‌شده را برای حمله مخرب نسبت به شبکه‌های سنتی، قوی‌تر نماید.

دستیابی غیرمجاز به‌عنوان‌مثال	لایه شبکه نرم‌افزاری تعریف‌شده تحت تأثیر یا مورد هدف				
	لایه برنامه کاربردی	واسط App-Ctl	لایه کنترل	واسط داده Ctl	لایه داده
دستیابی کنترل‌کننده غیرمجاز			✓	✓	✓
برنامه کاربردی تائید نشده	✓	✓	✓		
جابجایی غیرقانونی داده از امکانات کامپیوتری					✓
کشف قاعده جریان (حمله کانال جانبی روی بافر ورودی)					✓
کشف سیاست ارسال (تحلیل زمان‌بندی پردازش بسته)			✓	✓	✓
برنامه‌های کاربردی بد اندیشانه مثلاً					
درج قاعده کلاه‌برداری	✓	✓	✓		
تسخیر کنترل‌کننده			✓	✓	✓
رد سرویس مثلاً					
طغیان ارتباطات راه گزین- کنترل‌کننده			✓	✓	✓
طغیان جدول جریان راه گزینی					✓
مسائل پیکربندی مثلاً					
پذیرش کمبود TLS (تکنیک تائید سندیت)			✓	✓	✓
اعمال سیاست	✓	✓	✓		

جدول ۱-۲: دسته‌بندی مسائل امنیتی وابسته با چارچوب شبکه نرم‌افزاری تعریف‌شده از طریق لایه/ واسط تحت تأثیر قرار می‌گیرد.

۱-۵-۲ کادر میانی شبکه نرم‌افزاری تعریف‌شده

شبکه‌های سنتی از کادرهای میانی برای ارائه کارکردهای امنیت شبکه استفاده می‌کنند. اخیراً، در مورد تلفیق کادرهای میانی امنیتی، در شبکه نرم‌افزاری تعریف‌شده؛ برای به‌کارگیری مزیت قابلیت برنامه‌نویسی جهت هدایت مجدد ترافیک شبکه انتخاب‌شده از طریق کادر میانی، نظریاتی مطرح شده است. به‌عنوان‌مثال، معماری Slick، یک کنترل‌کننده متمرکز را ارائه می‌دهد که مسئول نصب و مهاجرت کارکردها روی کادرهای میانی برحسب عادت است. سپس برنامه‌های کاربردی می‌توانند کنترل‌کننده Slick را برای نصب کارکردهای ضروری، جهت مسیریابی جریان‌های خاص بر مبنای شرایط لازم امنیتی، هدایت کنند.

معماری FlowTags، استفاده از کادرهای میانی اصلاح‌شده حداقل را ارائه می‌دهد. این معماری توسط یک کنترل‌کننده شبکه نرم‌افزاری تعریف‌شده از طریق یک واسط برنامه‌نویسی برنامه کاربردی FlowTags تعامل برقرار می‌کند. FlowTags از

اطلاعات جریان ترافیکی تشکیل می‌شود که در عناوین بسته برای ارائه پیگیری جریان درج شده است و مسیریابی کنترل‌شده بسته‌های برچسب دار را میسر می‌سازد. یکی از مشکلات این معماری، این است که فقط با سیاست‌های از قبل تعریف‌شده کار می‌کند و در حال حاضر اقدامات دینامیکی را ساماندهی نمی‌کند.

لایه اعمال سیاست SIMPLE یک رویکرد برای استفاده از شبکه نرم‌افزاری تعریف‌شده جهت مدیریت گسترش‌های کادر میانی است. به هیچ اصلاحی در قابلیت‌های شبکه نرم‌افزاری تعریف‌شده یا عاملیت کادر میانی نیاز نیست که آن را برای سیستم‌های موروثی، مناسب می‌سازد.

بر مبنای این ارائه‌ها، به نظر می‌رسد یک رویکرد ساده برای تأمین امنیت شبکه شامل معرفی یک کادر میانی مناسب و برنامه‌نویسی شبکه برای هدایت ترافیک انتخاب‌شده از طریق کادر میانی است. یکپارچگی مناسب کادرهای میانی شبکه نرم‌افزاری تعریف‌شده باید در طول جریمه عملکردی تعیین گردد.

درعین‌حال، همان‌طور که در جدول ۱-۱ نشان داده شده است، طیف حملاتی که برای شبکه تهدید ایجاد می‌کنند، کاملاً شناخته‌شده هستند. به‌این‌ترتیب، فراتر از کادرهای میانی، یک سری از راه‌حل‌ها مطرح‌شده‌اند که به‌طور ویژه‌ای چارچوب شبکه نرم‌افزاری تعریف‌شده را برای تأمین راه‌حل‌های امنیت شبکه، به کار می‌گیرند.

۲-۵-۲ شبکه نرم‌افزاری تعریف‌شده = «شبکه‌سازی امنیت محور»؟

حمله‌کنندگان از روش‌های گوناگون اسکن برای کشف اهداف آسیب‌پذیر در شبکه استفاده می‌کنند. یک دفاع ارائه‌شده برای بی‌اثر کردن این حملات، استفاده از آدرس‌های پروتکل اینترنتی مجازی تصادفی با استفاده از شبکه نرم‌افزاری تعریف‌شده است. این تکنیک از کنترل‌کننده اوپن فلو برای مدیریت یک مخزن از آدرس‌های IP مجازی استفاده می‌کند که به هاست‌ها در شبکه تخصیص می‌یابند، IP آدرس‌های حقیقی را از دنیای خارجی پنهان می‌کنند. این کار دفاع هدف متحرک را ارائه می‌دهد که شکلی از امنیت مجازی سازگار است.

ایجاد سیستم‌های نظارتی برای حفاظت از شبکه در برابر حمله، ضروری است. یک روش تشخیص حمله منع سرویس توزیع‌شده (DDoS)، بر مبنای مشخصات جریان ترافیکی متعدد، است. این سیستم NOX را تحت نظارت قرار می‌دهد و در فواصل زمانی منظم مسیریابی می‌کند و از نقشه‌های خودسازمان‌دهنده برای شناسایی جریان‌های نابهنجار استفاده می‌کند. در رویکرد دیگری، OpenSAFE مبتنی بر سیاست ALARMS برای مدیریت مسیریابی ترافیک، از طریق دستگاه‌های نظارت شبکه استفاده می‌کند. در شبکه‌های ابری یک ایده مشابه با شبکه نرم‌افزاری تعریف‌شده تمرکز دارد که جریان‌های شبکه را برای تضمین این مسئله کنترل

می‌کند و کل بسته‌های شبکه ضروری با برخی دستگاه‌های امنیتی بازرسی می‌شوند. این چارچوب، ازنظر خودکار بودن خط سیر بسته‌های شبکه که باید از طریق دستگاه‌های امنیت شبکه‌ی از قبل نصب‌شده وارسی شوند، منحرف می‌کند.

این راه‌حل‌ها بر مبنای طرح مدیریت شبکه متمرکز است؛ بااین‌حال، سایر مشوق‌های کاری، نمایندگی برخی کنترل‌ها را به دستگاه‌ها و هاست‌های شبکه برمی‌گردانند. به‌عنوان‌مثال رزوناس، کنترل دستیابی دینامیکی اعمال‌شده توسط خود دستگاه‌ها را بر مبنای سیاست‌های امنیت در سطح بالاتر، ارائه می‌دهد. نائوس و همکارانش، پروتکل ident++ را برای پرس‌وجوی هاست‌های نهایی و برای اطلاعات بیشتر کاربران مطرح می‌کنند تا تصمیم ارسال بگیرند؛ بحث آن‌ها این است که کنترل‌کننده مرکزی می‌تواند به یک تنگنا تبدیل شود. درحالی‌که با حفظ ویژگی‌های قابلیت برنامه‌نویسی شبکه نرم‌افزاری تعریف‌شده، این روش‌ها، شامل بودن دستگاه‌های شبکه تحت کنترل شبکه را مطرح می‌کنند تا اینکه میزان وابستگی به یک کنترل‌کننده متمرکز واحد را مشخص کند.

شکل خاص سیستم نظارت IDS، مورد تمرکز شماری از راه‌حل‌های شبکه نرم‌افزاری تعریف‌شده قرار داشته است. اسکورا و همکارانش، یک IDS یادگیری را ارائه می‌دهند، از معماری شبکه نرم‌افزاری تعریف‌شده هم برای کشف و هم پاسخ‌دهی به حملات شبکه که در دستگاه‌های موبایل تعبیه‌شده، استفاده می‌کند. یک NIDS تسریع شده سخت‌افزاری (IDS شبکه) یا طرح NIPS (IPS شبکه) که اجازه می‌دهد مدیر شبکه، الگوهای رشته‌ای را برای استفاده از طریق یک ماژول بازرسی بسته عمیق (DPI) پیکربندی نماید.

امکان تقویت و ساده‌سازی امنیت شبکه، از طریق معماری شبکه نرم‌افزاری تعریف‌شده، از این تحقیقات برداشت می‌شود. این پتانسیل ازنظر بازرگانی، با طیفی از محصولات امنیت شبکه نرم‌افزاری تعریف‌شده در مراحل مختلف توسعه، به رسمیت شناخته‌شده است.

۶-۲ چالش‌های امنیتی

درحالی‌که امنیت به‌عنوان مزیت چارچوب شبکه نرم‌افزاری تعریف‌شده به رسمیت شناخته‌شده است، راه‌حل‌های معدودی برای حل تأمین چالش‌های شبکه نرم‌افزاری تعریف‌شده، وجود دارند.

شبکه‌های نرم‌افزاری تعریف‌شده، به ما توانایی برنامه‌نویسی راحت شبکه و همچنین بررسی خلاقیت سیاست‌های جریان دینامیکی را ارائه می‌دهند. اعمال سیاست امنیت شبکه در این محیط، حیاتی است. در سیاست‌های حاصل از برنامه‌های کاربردی یا نصب‌شده در کل دستگاه‌های متعدد، بررسی مدل به یک مرحله مهم در کشف ناسازگاری‌ها تبدیل می‌شود. بررسی مدل ترکیب‌شده با اجرای نمادین ممکن است برای بررسی درستی برنامه‌های کاربردی اوپن فلو به کار می‌رود. هم‌چنین می‌توان برای محک زدن

سوء پیکربندی مسیریابی داخلی از نمودارهای تصمیم دودوئی در یک جدول جریان واحد، استفاده کرد. FlowChecker، FlowVisor را بکار می‌گیرد که جداسازی را با پارتیشن‌بندی منابع شبکه به‌صورت بخش‌ها، میسر می‌سازد. سان و همکارانش، فلاور را مطرح می‌کنند که از مجموعه اثبات‌ها و نظریه‌ها برای اثبات سیاست‌های جریان استفاده می‌کند؛ درحالی‌که VeriFlow اثبات نامتغیرها را در زمان حقیقی، مطالعه می‌کند. یک‌لایه اضافی که بین کنترل‌کننده شبکه نرم‌افزاری تعریف‌شده و دستگاه‌های شبکه قرار دارد، مانع قواعد جریان قبل از رسیدن به شبکه می‌شود. اگرچه VeriFlow، تأخیر کم (سرعت) فرایند بررسی را بهتر می‌کند، اما نمی‌تواند کنترل‌کننده‌های متعدد را ساماندهی نماید. اعمال سیاست مبتنی بر جریان را در کنار جداسازی شبکه، میسر می‌سازد. این راه‌حل، به‌صورت یک برنامه کاربردی NOX اجرا می‌شود و اجازه یکپارچه‌سازی منابع تائید هویت خارجی برای تأمین کنترل دستیابی را می‌دهد. اخیراً، جداسازی عالی (Splendid Isolation) به‌عنوان وسیله‌ای برای اثبات جداسازی ترافیک برنامه، مطرح‌شده است

فرسکو، یک نقش برجسته دارد که یک چارچوب توسعه برنامه کاربردی امنیت اوپن فلو را ارائه می‌دهند. ایده پشت FRESCO، اجازه طراحی سریع و توسعه ماژول‌های ویژه امنیتی است که می‌تواند به‌عنوان یک برنامه کاربردی اوپن فلو همراه شود. پوراس و همکارانش یک کتابخانه از ماژول‌های قابل‌استفاده مجدد را ارائه می‌دهند. این کتابخانه می‌تواند برای کشف و کاهش تهدیدهای شبکه به کار رود. این سیستم، موتور اعمال FortNox را به همراه دارد که تعارض‌های ممکن را با درج قاعده، ساماندهی می‌کند. اگر یک قاعده متعارض به‌عنوان نتیجه یک قاعده اوپن فلو پدید آید، به صورتی که یک قاعده موجود ممنوع یا مجاز را فعال یا غیرفعال کند، آنگاه قاعده جدید، بسته به میزان تائید هویت امنیت مؤلف، برای ارائه قاعده متعارض موجود، پذیرفته یا رد می‌شود. گرچه FortNox، مؤلفه‌های بی‌شماری را ارائه می‌دهد که برای اعمال امنیت ضروری هستند، مؤلفین احساس می‌کنند بازهم کار زیادی برای ارائه یک مجموعه جامع از برنامه‌های کاربردی نیاز است.

با حرکت از فضای طراحی به‌سوی اجرا، یکی از دغدغه‌های مهم صنعت با امنیت در شبکه نرم‌افزاری تعریف‌شده، تأمین فرایند حسابرسی است. برای اجابت و عملیات شبکه، مقدار موجودی کنترل‌شده از دستگاه‌های شبکه، لازم است. این موجودی، اطلاعاتی ازجمله نوع دستگاه‌هایی که کار می‌کنند و اینکه چطور به شبکه و غیره محدود می‌شوند را دربرمی گیرد. این مسئله به‌طور مستقیم به پتانسیل مجازی‌سازی عناصر شبکه و کارکردهایی که از طریق شبکه نرم‌افزاری تعریف‌شده حمایت می‌شود، مربوط است. گرچه یک چالش حل‌نشده پیرامون عملی بودن وضعیت شبکه نگاشت در کل کارکردهای موبایل و مجازی وجود دارد، اما برخی از آثار مرتبط درزمینه اثبات شبکه، ارزش ذکر کردن را دارند. مسئله مقیاس‌پذیری و امنیت شبکه‌های اوپن فلو و استفاده آن‌ها در فضای فیزیکی – سایبری را بررسی می‌کنند. Verificare مدل‌سازی مشخصات، اثبات صحت شبکه، همگرایی و خواص مربوط به پویایی

را بررسی می‌کند. هادیگل و همکارانش، استفاده از اشکال‌یابی شبکه نمونه اولیه را مطرح می‌کنند؛ که می‌تواند به توسعه‌دهندگان شبکه نرم‌افزاری تعریف‌شده اجازه دهد، زنجیره رویدادهایی را بازسازی کنند که منجر به یک خطا[1] و شناسایی دلیل ریشه‌ای آن می‌شود.

[1] Bug

فصل سوم:
مهندسی ترافیک در شبکه نرم‌افزاری تعریف‌شده: اندازه‌گیری و مدیریت

۳-۱ مهندسی ترافیک مبتنی بر IP

به‌طورکلی، مهندسی ترافیک مبتنی بر IP، مشکل تعادل بار ترافیک چند مسیری برای اجتناب از ازدحام شبکه را با استفاده از بهینه‌سازی الگوریتم مسیریابی IP حل می‌کند. به‌عنوان‌مثال، FORTZ و THORUP یک الگوریتم جستجوی محلی را پیشنهاد داده‌اند که بر مبنای وزن لینک‌های اولین مسیر کوتاه باز (OSPF) برای تنظیم استراتژی محاسبه مسیریابی به کار می‌رود. این الگوریتم درنهایت منجر به محاسبه کوتاه‌ترین مسیرهای معادل چندگانه جهت کسب تعادل بار ترافیک می‌شود. CHEN و همکارانش یک برنامه‌ریزی چندگانه را برای سنجش چندرسانه‌ای اینترنت اشیا طراحی کردند. تکنولوژی مهندسی ترافیک بر مبنای IP دارای دو اشکال واضح است:

اول: هنگامی‌که وزن لینک‌های OSPF برای کنترل مسیریابی یک شبکه مورداستفاده قرار می‌گیرد، ترافیک نمی‌تواند در یک نسبت دلخواه تقسیم گردد و این امر منجر به عدم توانایی استفاده کامل از منابع شبکه خواهد شد

دوم: هنگامی‌که لینک‌ها شکسته می‌شوند، یا وزن لینک‌های توپولوژی شبکه تغییر می‌کند، پروتکل OSPF مدت‌زمانی را صرف پوشش یک توپولوژی جدید شبکه می‌کند که احتمالاً این زمان صرف ازدحام شبکه، تلفات بسته‌ها، تأخیرها و حتی لوپ‌های مسیریابی می‌گردد.

به‌منظور اجتناب از کاستی‌ها و نواقص مهندسی ترافیک مبتنی بر IP، محققان راه‌حل دیگری را مطرح کرده‌اند که در آن بسته‌های شبکه به‌وسیله پروتکل MPLS (سوئیچینگ برچسب چند پروتکلی) به‌جای هدرهای IP ارسال می‌شوند. بااین‌حال، مکانیزم پروتکل MPLS بسیار پیچیده بوده و می‌تواند منجر به یک سربار با تأثیر بالا گردد؛ بنابراین برآوردن نیازهای شبکه‌های مرکز داده، شامل به‌کارگیری پهنای باند لینک بالا، صرفه‌جویی در انرژی سبز و قابلیت اطمینان بالا، مشکل خواهد بود.

به‌طور خلاصه، مدیریت کنترل و ارسال داده در شبکه‌های سنتی به‌شدت کوپل شده هستند، ازاین‌رو تمام شبکه به‌وسیله دستگاه‌های توزیع‌شده کنترل می‌شود که کنترل ریزدانه مدیریت ترافیک نمی‌تواند حاصل شود. در این حالت انعطاف‌پذیری و توسعه‌پذیری برای اصلاح و بهبود سخت است؛ بنابراین، توسعه معماری شبکه و تکنولوژی مربوط به مهندسی ترافیک جهت حل این مشکل ضروری است.

۳-۲ مقایسه معماری شبکه‌های سنتی و شبکه نرم‌افزاری تعریف‌شده

دارای ویژگی‌های متمایز زیر هست:

۳-۲-۱ تمرکز کنترل

کنترلر شبکه نرم‌افزاری تعریف‌شده تمام اطلاعات شبکه شامل توپولوژی شبکه، تغییرات پویا در حالت‌های شبکه و الزامات اپلیکیشن سرا سری مانند QoS (کیفیت سرویس) و الزامات امنیتی را ذخیره می‌کند.

۳-۲-۲ برنامه پذیری

یک کاربر شبکه می‌تواند به‌طور کاملاً پویا به دستگاه‌های مربوط به لایه ارسال داده، جهت بهینه‌سازی تخصیص منابع شبکه، برنامه دهد.

۳-۲-۳ باز بودن

تجهیزات ارسال دارای یک رابط (اینترفیس) واحد و یکپارچه جهت برقراری ارتباط با کنترلر شبکه نرم‌افزاری تعریف‌شده بوده که به تأمین‌کنندگان مختلف تجهیزات وابستگی ندارد. همچنین کنترلر شبکه نرم‌افزاری تعریف‌شده می‌تواند به‌راحتی اطلاعاتی از وضعیت شبکه برای برنامه‌ریزی ترافیک شبکه به دست آورد.

۳-۳ ویژگی‌های شبکه نرم‌افزاری تعریف‌شده برای حل مشکلات فعلی مهندسی ترافیک در شبکه

مفید هستند به‌صورت خلاصه به شرح ذیل هستند:

۳-۳-۱ سنجش ترافیک

ما می‌توانیم وظایف سنجش سرا سری مقیاس‌پذیر را به‌طور منعطف در شبکه نرم‌افزاری تعریف‌شده مستقر نماییم که می‌تواند اطلاعات اوضاع شبکه را در بلادرنگ جمع‌آوری نموده و ترافیک را به‌صورت متمرکز در کنترلر، نظارت و تجزیه تحلیل نماید.

۳-۳-۲ برنامه‌ریزی و مدیریت ترافیک

الزامات اپلیکیشن ترافیک می‌تواند به‌طور سرا سری مطرح شود؛ به‌گونه‌ای که برنامه‌ریزی ترافیک جزءبه‌جزء و انعطاف‌پذیر امکان‌پذیر است.

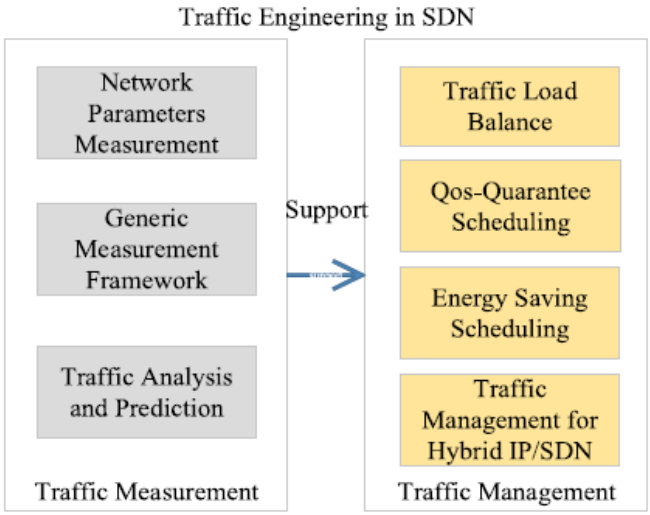
۳-۳-۳ سوئیچ اوپن فلو

دارای خطوط لوله جدول جریان متعدد است که مدیریت جریان را انعطاف‌پذیرتر و مؤثرتر می‌سازد.

بااین‌حال، اگرچه شبکه نرم‌افزاری تعریف‌شده پشتیبانی زیادی را برای مدیریت ترافیک فراهم می‌نماید، اما مدیریت ترافیک دارای مشکلات زیادی است که لازم است برطرف گردند. در حال حاضر، هیچ مطالعه‌ای که بیانگر سازگاری کامل فن مهندسی ترافیک شبکه سنتی با شبکه نرم‌افزاری تعریف‌شده باشد، وجود ندارد. مشکل دیگر این است که شبکه نرم‌افزاری تعریف‌شده و شبکه IP سنتی برای مدتی طولانی در کنار هم خواهند بود. البته تحقیقات کمی با توجه به تکنولوژی مهندسی ترافیک شبکه‌های ترکیبی IP/SDN وجود دارد؛ بنابراین، فن‌آوری‌های مهندسی ترافیک بر اساس شبکه نرم‌افزاری تعریف‌شده برای اپلیکیشن‌های بعدی شبکه نرم‌افزاری تعریف‌شده مهم‌تر است.

۳-۴ چهارچوب مهندسی ترافیک در شبکه نرم‌افزاری تعریف‌شده

همان‌گونه که در شکل ۱-۳ نشان داده‌شده است، چارچوبی را برای مهندسی ترافیک در شبکه نرم‌افزاری تعریف‌شده، با ترکیب ایده‌های مهندسی ترافیک در شبکه‌های سنتی و ویژگی‌های شبکه نرم‌افزاری تعریف‌شده، بررسی می‌کنیم.



شکل ۱-۳: چهارچوبی برای مهندسی ترافیک در شبکه نرم‌افزاری تعریف‌شده

این چارچوب شامل دو بخش است. سنجش ترافیک و مدیریت ترافیک. سنجش ترافیک عمدتاً به مطالعه چگونگی نظارت، اندازه‌گیری و به دست آوردن اطلاعات اوضاع شبکه در پیرامون شبکه نرم‌افزاری تعریف‌شده می‌پردازد. اطلاعات وضعیت شبکه شامل وضعیت پورت‌ها (بالا یا پایین)، انواع مختلف شمارشگر بسته، شمارگرهای بسته رهاشده، نسبت به‌کارگیری پهنای باند لینک، پوشیدگی شبکه end-to-end، ماتریس ترافیک end-to-end و غیره است. بر اساس اطلاعات وضعیت شبکه، می‌توان تنفیذ

کرد که آیا وضعیت فعلی شبکه صحیح است یا خیر. همچنین می‌توان برای جلوگیری از ازدحام شبکه و بهبود راندمان شبکه، روند ترافیک شبکه در آینده را به‌وسیله تجزیه‌وتحلیل آمار و ارقام شمارگرهای بسته پیش‌بینی کرد. سنجش ترافیک را در سه جهت تقسیم کرده‌ایم: اندازه‌گیری پارامترهای شبکه، یک چارچوب اندازه‌گیری عمومی، تجزیه‌وتحلیل و پیش‌بینی ترافیک.

مدیریت ترافیک عمدتاً چگونگی مدیریت را بررسی می‌کند و ترافیک شبکه را بر اساس اطلاعات وضعیت شبکه برنامه‌ریزی می‌کند. اطلاعات وضعیت شبکه به‌وسیله توپولوژی سنجش ترافیک به دست می‌آید تا نیازهای کاربر نهایی اپلیکیشن‌های شبکه، همچون QoS، برآورده گردد. مدیریت ترافیک را در چهار جهت تقسیم کرده‌ایم: تعادل بار ترافیک، برنامه‌ریزی کیفیت سرویس تضمینی، برنامه‌ریزی صرفه‌جویی در انرژی و مدیریت ترافیک برای کنترلر ترکیبی IP/SDN. لازم به ذکر است که به‌کارگیری و استفاده از کنترلر ترکیبی IP/SDN سابقه زیادی دارد؛ بنابراین ضروری است که مهندسی ترافیک در ترکیب IP/SDN موردمطالعه قرار گیرد.

۳-۵ سنجش ترافیک در شبکه نرم‌افزاری تعریف‌شده

سنجش ترافیک بخش مهمی از مهندسی ترافیک است. این بخش فن‌آوری‌های مرتبط با ارسال را از سه منظر زیر مطرح می‌کند: طراحی پارامترهای شبکه و فن‌آوری‌های مانیتورینگ، یک چارچوب عمومی سنجش جریان، تجزیه‌وتحلیل ترافیک و فن‌آوری‌های پیش‌بینی.

۳-۵-۱ سنجش پارامترهای شبکه

پارامترهای شبکه مجموعه‌ای از شاخص‌هاست که وضعیت فعلی شبکه را نشان می‌دهد. طراحی پارامتر معقول و مناسب برای شبکه یک پیش‌شرط برای مدیریت مؤثر شبکه است؛ بنابراین، طراحی پارامترهای شبکه وظیفه اولیه و اصلی سنجش شبکه است. پارامترهای سنجش شبکه نرم‌افزاری تعریف‌شده عمدتاً شامل سه نوع است: پارامترهای توپولوژی شبکه، پارامترهای ترافیک شبکه و پارامترهای کارایی شبکه.

پارامترهای توپولوژی شبکه به تعداد گره‌های شبکه، پهنای باند لینک، ساختارهای اتصال و وضعیت پورت‌ها بستگی دارند. با توجه به مشخصات اوپن فلو، کنترلر شبکه نرم‌افزاری تعریف‌شده توپولوژی شبکه فعلی را به‌صورت فعالانه با استفاده از پروتکل کشف لایه لینک (LLDP)، نمایان ساخته و اطلاعات توپولوژی سرا سری را حفظ می‌کند. در طول فرایند کشف لینک، کنترلر شبکه نرم‌افزاری تعریف‌شده بسته‌های LLDP را به‌عنوان پیام‌های بسته خروجی به تمام سوئیچ‌ها در شبکه ارسال می‌کند. هنگامی‌که سوئیچ شبکه نرم‌افزاری تعریف‌شده یک بسته LLDP را از کنترلر دریافت می‌کند، بسته را به تمام سوئیچ‌های دیگری که مستقیماً به

آن‌ها متصل گردیده ارسال می‌کند. هنگامی‌که یک سوئیچ، یک بسته LLDP را از سوئیچ دیگری دریافت می‌کند، بسته LLDP را به کنترلر به‌عنوان یک پیام بسته ورودی برای کمک ارسال می‌کند، چراکه هیچ قاعده ارسال مرتبط در سوئیچ جدول جریان وجود ندارد. پس‌ازآنکه کنترلر شبکه نرم‌افزاری تعریف‌شده پیام‌های بسته ورودی را با LLDP دریافت کرد، می‌تواند تجزیه‌وتحلیل کند که سوئیچ‌ها مستقیماً به یکدیگر متصل بوده و توپولوژی سرا سری را ایجاد کند. روند کار مربوطه در شکل ۳–۲ نشان داده‌شده است.

شکل ۲–۳: نحوه عملکرد LLDP برای توپولوژی شبکه نرم‌افزاری تعریف‌شده

پارامترهای ترافیک شبکه به تعداد یا سرعت بسته‌های شبکه که از طریق تجهیزات شبکه یا یک پورت شبکه عبور می‌کنند،

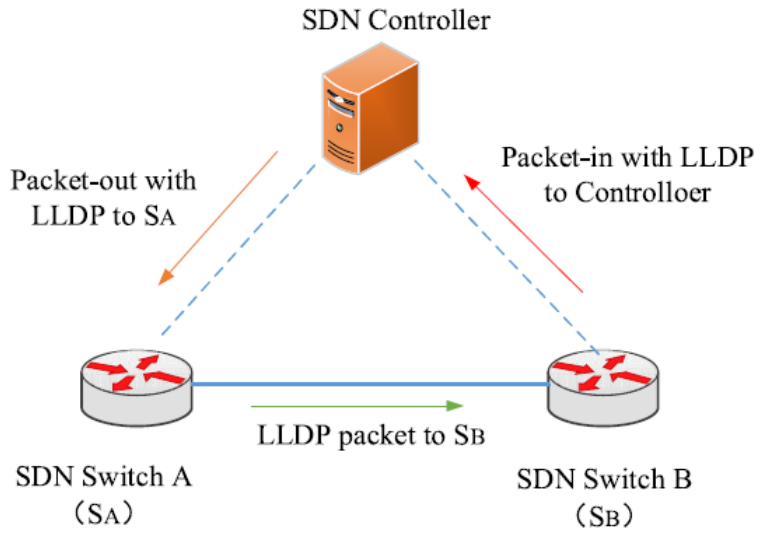


FIGURE 2. Detection process using LLDP for SDN topology.

مانند تعداد کل بسته‌های IP، تعداد بایت‌ها در ثانیه و غیره، بستگی دارند. پارامترهای ترافیک شبکه به‌عنوان یک بیس مهم برای تشخیص وضعیت شبکه‌های فعلی و تجزیه‌وتحلیل رفتار کاربران شبکه مطرح است. در شبکه نرم‌افزاری تعریف‌شده، دو نوع ترافیک شبکه وجود دارد، ترافیک کنترل و ترافیک داده.

ترافیک کنترل جریان داده ایست که بین کنترلر شبکه نرم‌افزاری تعریف‌شده و سوئیچ‌ها منتقل می‌شود.

ترافیک داده جریان داده‌ای است که بین سوئیچ‌ها منتقل می‌گردد.

به‌منظور تعیین ویژگی‌های جریان، بایستی اطلاعات آماری از هر پورت سوئیچ جمع‌آوری کنیم. این اطلاعات شامل تعداد بسته‌ها، اندازه بسته‌ها و همچنین ماتریس ترافیک end-to-end کل شبکه است. این ماتریس بیانگر جریان شبکه بین هر دو گره شبکه است. Yuan و همکارانش، یک معماری اندازه‌گیری باقابلیت برنامه‌ریزی را تحت عنوان ProgME پیشنهاد کردند که می‌تواند آمار ترافیک کاربر خاص را جمع‌آوری کند. آن‌ها یک سیستم برآورد ماتریس ترافیک را برای شبکه نرم‌افزاری تعریف‌شده

پیشنهاد کردند که OpenTM نام دارد. OpenTM می‌تواند تمام جریان‌های شبکه فعال را با توجه اطلاعات مسیریابی کنترلر و اطلاعات مسیر ارسال جریان، شناسایی و کشف نماید. OpenTM شامل انواع روش‌های پرس‌جوی انتخابی برای مسیریابی گره‌ها، جهت به دست آوردن اطلاعات دقیق در مورد تعداد بسته‌ها و ارزیابی جریان است. به‌طورکلی، تعداد جریان‌های شبکه خیلی بزرگ است؛ اما منابع اندازه‌گیری در دسترس، مانند CPU و جدول حافظه آدرس پذیر محتوی سه‌تایی (TCAM)، محدود هستند؛ بنابراین به دست آوردن ماتریس ترافیک به‌وسیله اندازه‌گیری مقدار هر جریان به‌صورت مستقیم، غیرممکن است. به‌منظور حل این مشکل، در روش iSTAMP ابتدا مقدار بزرگ‌ترین جریان k یا جریان همگرایی را، اندازه‌گیری کرده سپس ماتریس ترافیک یکپارچه را با استفاده از فن‌آوری‌های مربوطه تخمین می‌زند. به نظر می‌رسد که iSTAMP تعادل بین محدودیت‌های منابع شبکه و دقت اندازه‌گیری را بیابد؛ اما هنوز هم دارای اشکالاتی ازجمله اشکالات زیر است:

اول: هنگامی‌که سوئیچ‌های شبکه نرم‌افزاری تعریف‌شده بسته‌ها را ارسال می‌کنند، قاعده تطبیق جهت یافتن یک ارسال بر مبنای اولویت و نشانه‌های عام است و فقط جریان‌های دارای گره مقصد مشابه، با اشاره به محدودیت‌های ممکن تراکم جریان، می‌تواند در یک مورد جمع شود؛ اما در روش iSTAMP، محدودیت‌ها در تراکم جریان را نادیده می‌گیرد.

دوم: به‌منظور پیدا کردن بزرگ‌ترین جریان k، iSTAMP جدول TCAM را برای اندازه‌گیری جریان‌های تک در دوره‌های زمانی متعدد بکار می‌برد. در این صورت هزینه‌های اندازه‌گیری به میزان چشم‌گیری افزایش می‌یابد. Hu و Luo برآورد ماتریس ترافیک را در شبکه مرکز داده بر اساس شبکه نرم‌افزاری تعریف‌شده، موردمطالعه قرار داده‌اند. آن‌ها برای جلوگیری از مشکل تجمع جریان مذکور در بالا، فرض می‌کنند که تجمع جریان‌ها، تنها در سوئیچ‌های ToR SDN رخ می‌دهد. بااین‌حال، پیاده‌سازی این متد در یک شبکه واقعی دشوار است.

پارامترهای کارایی شبکه عمدتاً شامل تأخیر شبکه، به‌کارگیری پهنای باند، میزان توان عملیاتی، نرخ از دست دادن بسته و غیره است. Van Andrichem و همکارانش یک ماژول مانیتورینگ شبکه (OpenNetMon) را ارائه کردند که روش‌های اندازه‌گیری مختلفی برای پارامترهای کارایی شبکه طراحی می‌کند. OpenNetMon می‌تواند توان عملیاتی شبکه، نرخ از دست دادن بسته و تأخیرهای شبکه را به‌طور پیوسته نظارت کند. اول: هنگامی‌که نرخ توان عملیاتی را اندازه‌گیری می‌کند، تنها از آخرین سوئیچ در مسیر ارسال با استفاده از یک فرکانس تطبیقی پرس‌وجو می‌کند. شمارنده، تعداد بسته‌های (S) هر جریان را در فاصله نمونه‌برداری (T) برمی‌گرداند. به این صورت توان عملیاتی مسیر ارسال، می‌تواند به‌عنوان S/T به دست آید. دوم: هنگامی‌که نرخ از دست دادن بسته را محاسبه می‌کند، سوئیچ‌های اول و آخر در مسیر ارسال را جهت به دست آوردن آمار شمارنده بررسی می‌کند، سپس افزایش شمارنده اول منهای افزایش شمارنده آخر در یک سیکل اندازه‌گیری شده و درنهایت نتیجه تقسیم‌بر زمان شده و نرخ

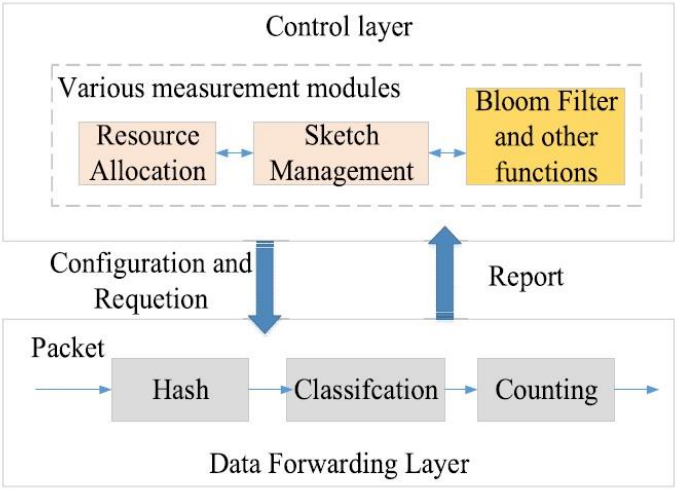
از دست دادن بسته در این مسیر ارسال، به دست می‌آید. سوم: برای محاسبه تأخیر، OpenNetMon پیام‌های کاوشگر را به لایه‌های داده در مسیر ارسال می‌فرستد. این پیام‌ها از میان تمام گره‌های این مسیر رسیده و درنهایت به کنترلر بازمی‌گردند. جایی که تأخیرهای ارتباطات می‌تواند با محاسبه اختلافات زمانی به دست آید. حسگر جریان (FlowSense) یک ماژول نظارتی را برای کنترلر شبکه نرم‌افزاری تعریف‌شده ایجاد می‌کند که می‌تواند تغییرات پویا در جریان‌های شبکه را با توجه به پیام‌های دریافتی توسط کنترلر، آنالیز نماید. به‌عنوان‌مثال، پس‌ازاینکه کنترلر یک پیام FlowRemoved از یک جریان خاص را دریافت می‌کند، نرخ توان عملیاتی این جریان با استفاده از مقادیر آماری جریان تقسیم‌شده به‌وسیله اندازه‌ی جریان مربوطه، به دست می‌آید.

۲-۵-۳ چارچوب اندازه‌گیری عمومی

اندازه‌گیری ترافیک شبکه دارای دو نوع مد است. مد فعال و غیرفعال. مد اندازه‌گیری فعال، ترافیک تشخیص اضافی را تولید می‌کند؛ درحالی‌که مد اندازه‌گیری غیرفعال این کار را نمی‌کند؛ چراکه این مد تنها ترافیکی را که از میان سوئیچ‌های شبکه نرم‌افزاری تعریف‌شده عبور می‌کنند را نظارت می‌کند. بسیاری از سیستم‌های اندازه‌گیری شبکه نرم‌افزاری تعریف‌شده روش‌های نظارتی ترافیک شبکه‌های IP سنتی مبتنی بر نمونه‌گیری بسته را اتخاذ کرده‌اند. در این نوع از روش‌های نظارتی سوئیچ‌های شبکه نرم‌افزاری تعریف‌شده به‌طور تصادفی بسته‌های محلی را برای به دست آوردن آمار ترافیک شناور می‌سازد. یک سیستم معروف برای نمونه‌برداری و آنالیز بسته تحت عنوان NetFlow وجود دارد که توسط سیسکو توسعه‌یافته است. NetFlow با استفاده از پنج روش جریان‌ها را شناسایی می‌کند که در آن‌یک سوئیچ برای هر جریان اطلاعات رکورد حافظه کش را نگهداری می‌کند. هنگامی‌که یک جریان می‌رسد، NetFlow اطلاعات هد این جریان را با اطلاعات ذخیره‌شده در یک رکورد حافظه کش محلی مقایسه می‌کند. اگر این دو با یکدیگر مطابقت داشته باشند، شماره بسته به‌روزرسانی می‌گردد، در غیر این صورت یک ورودی جدید به رکورد حافظه کش برای این جریان جدید می‌افزاید. از دیگر راه‌حل‌های نمونه‌برداری جریان شامل sFlow و jFlow است. این دو روش در ورودی هر روتر به یک نسبت خاص نمونه‌برداری کرده سپس اطلاعات سر جریان و برچسب زمانی مربوط به سرور گردآورنده مرکزی را ارسال می‌کنند. تمام روش‌های اندازه‌گیری بالا به زیاد بودن بار مسئولیت در کنترلر مرکزی شبکه نرم‌افزاری تعریف‌شده در شبکه‌های مقیاس بزرگ اشاره دارند؛ بنابراین آن‌ها برای اپلیکیشن‌های شبکه نرم‌افزاری تعریف‌شده مناسب نیستند.

به‌منظور پشتیبانی مؤثر از انواع task های اندازه‌گیری و اپلیکیشن‌ها برای شبکه نرم‌افزاری تعریف‌شده، طراحی یک چارچوب اندازه‌گیری شبکه نرم‌افزاری تعریف‌شده عمومی ضروری است. PayLess یک چارچوب اندازه‌گیری جریان با هزینه کم و مبتنی بر رأی‌گیری است. چارچوب PayLess به‌عنوان یک جزء از کنترلر اوپن فلو طراحی‌شده است و یک رابط کاربری RESTful برای اپلیکیشن‌ها فراهم می‌کند. چهارچوب PayLess به‌صورت خاص مسئول تجزیه دستورات درخواستی از سطح اپلیکیشن وظایف و

انتقال این دستورات در مسیر طراحی‌شده به برخی از سوئیچ‌ها است؛ بنابراین PayLess نمایی خلاصه از اطلاعات شبکه را برای اپلیکیشن‌ها نگهداری کرده و رابط‌های برنامه‌نویسی یکپارچه را برای انواع اپلیکیشن‌های شبکه فراهم می‌کند. PayLess یک اپلیکیشن کاری را در فرمت JSON تعریف می‌کند که در آن مؤلفه‌های مرسوم کاربران می‌تواند به چارچوب اندازه‌گیری اضافه‌شده باشد. Yu و همکارانش چارچوب اندازه‌گیری انتزاعی OpenSketch را با استفاده از یک روش تعریف‌شده نرم‌افزاری برای اندازه‌گیری ترافیک به‌صورت کلی ارائه نموده‌اند. اندازه‌گیری مبتنی بر جریان دارای نقص‌هایی همچون سربار شمارنده و عدم دقت اندازه‌گیری است؛ بنابراین قادر به مواجه با نیازهای مختلف کاربران نیست. از سوی دیگر، OpenSketch از سفارشی‌سازی برای کاربران مختلف پشتیبانی می‌کند و وظایف مختلف دارای نیازمندی‌های مختلفی برای سخت‌افزار است؛ بنابراین گسترش وظایف متعدد بر روی تجهیزات سخت‌افزاری مشابه امری است مشکل. به‌منظور در نظر گرفتن کامل تطبیق‌پذیری و بهره‌وری، OpenSketch پیشنهاد می‌کند که کنترل اندازه‌گیری و لایه‌های داده باید از هم جدا شوند، در آن لایه‌های داده به‌عنوان اقدامات سه مرحله‌ای که می‌تواند به‌صورت پویا پیکربندی‌شده باشد طراحی‌شده‌اند. در مرحله اول، OpenSketch انواع الگوریتم‌های hash را ارائه می‌کند که بسته‌ها را در مقدار کمی از داده اندازه‌گیری ترسیم می‌کند. در مرحله دوم، قوانین با استفاده از TCAM و تطبیق جستجوی عمومی، ذخیره‌سازی و جستجو می‌گردد. در مرحله سوم، شمارش منعطف با استفاده از SRAM فراهم می‌گردد. با استفاده از این طراحی خط لوله، OpenSketch یک نوع از الگوریتم مقداری را در چند مرحله عمومی خلاصه می‌کند. در همان زمان، ماژول‌های اندازه‌گیری مختلفی را در لایه کنترل فراهم می‌کند که به کاربران اجازه می‌دهد تا با هر مرحله از لایه داده ذکرشده در بالا سازگار شوند؛ اما باید توجه داشت که OpenSketch نیازمند طراحی مجدد بخش‌هایی از سخت‌افزار بوده که احتمالاً کاربرد گسترده آن را محدود می‌کند. شکل ۳-۳ معماری OpenSketch را نشان می‌دهد.



شکل ۳-۳: معماری OpenSketch

مجموعه اطلاعات مربوط به ترافیک و تجمع، دو بخش مهم از اندازه‌گیری ترافیک هستند که نیازمند پشتیبانی سخت‌افزاری در لایه داده، مانند TCAM هستند. بااین‌حال، منابع سخت‌افزاری موجود شامل سوئیچ‌های شبکه نرم‌افزاری تعریف‌شده، غالباً محدود هستند. علاوه بر این اپلیکیشن‌های شبکه، به‌جای پشتیبانی از تنها یک task اندازه‌گیری در یک دوره زمانی، دارای چندین نیازمندی task نظارتی جریان هم‌زمان هستند؛ بنابراین مسئله‌ی چگونگی تخصیص منابع سخت‌افزاری به‌صورت منطقی، جهت فراهم نمودن اندازه‌گیری جریان مؤثر و دقیق نیز بسیار مهم است. از این نظر، Moshref و همکارانش یک معماری مدیریت تطبیقی برای منابع سخت‌افزاری تحت عنوان DREAM را پیشنهاد نموده‌اند. این معماری می‌تواند دقت اندازه‌گیری و هزینه منابع را به‌صورت ویژه متعادل نماید. ایده اصلی این معماری به شرح زیر است:

- دقت یک task اندازه‌گیری جریان، به منابع سخت‌افزاری اختصاص داده‌شده به آن task بستگی دارد. با افزایش تعداد منابع اختصاص داده‌شده به یک task، نکته‌ای وجود خواهد داشت و آن اینکه پس‌ازآن توانایی ارتقا دقت task با افزایش بیشتر در منابع کمتر می‌شود.
- منابع موردنیاز توسط task های اندازه‌گیری، بسته به زمان و اندازه‌ی جریان، مختلف هستند؛ بنابراین سازگاری این منابع جهت بالا بردن راندمان آن‌ها امکان‌پذیر است. DREAM سه سطح را شامل می‌شود: سطح بالایی: لایه کاربر است که مسئول تولید task های اندازه‌گیری شامل انواع task، سطح آستانه جریان خاص، دقت موردنیاز و غیره است. سطح میانی: الگوریتم‌های DREAM در حال اجرا در کنترلر شبکه نرم‌افزاری تعریف‌شده است. آن‌ها task های کاری را از کاربران دریافت و سپس اهداف task مربوطه را ایجاد کرده و درنهایت این task های انتزاعی را در سوئیچ‌های متعدد گسترش می‌دهند. درواقع، فرایند گسترش نیز روندی جهت درخواست منابع TCAM در سوئیچ‌ها و ذخیره‌سازی منطق اندازه‌گیری در TCAM است. با توجه به بازخورد زمان واقعی مانیتورینگ ترافیک، الگوریتم‌های DREAM می‌توانند دقت اندازه‌گیری‌ها را ارزیابی کرده و منابع اختصاص داده‌شده به هر task یا استفاده مجدد منابع task های متعدد را تنظیم نماید. از سوی دیگر، این الگوریتم‌ها می‌توانند تصمیم بگیرند که آیا یک task جدید با توجه به وضعیت فعلی بپذیرند یا خیر. سطح پایینی: ابزارهای ارسال شبکه نرم‌افزاری تعریف‌شده است که مسئول اندازه‌گیری منابع ذخیره‌سازی هستند و نتایج شمارش جریان زمان واقعی را برمی‌گردانند.

۳-۶ تجزیه‌وتحلیل و پیش‌بینی ترافیک

تجزیه‌وتحلیل و پیش‌بینی ترافیک بخش مهمی از اندازه‌گیری ترافیک است که یکی از اهدافش تشخیص ترافیک غیرعادی شبکه و تجزیه‌وتحلیل احتمالات ممکن در شبکه، مانند تراکم‌های شبکه است. به‌این‌ترتیب، ما می‌توانیم اطلاعات بهتری برای برنامه‌ریزی و مدیریت ترافیک شبکه فراهم کنیم. از سوی دیگر، یکی دیگر از اجزای مهم این تست است که آیا حالت‌های فعلی شبکه صحیح است یا خیر. ازجمله این حالات شامل صحت پیکربندی تجهیزات شبکه فعلی، مشکلات لوپ مسیریابی و غیره است. در این بخش، فناوری‌های مرتبط در این رابطه را جهت تجزیه‌وتحلیل و پیش‌بینی ترافیک معرفی می‌کنیم.

در اپلیکیشن روش‌های تشخیص غیرطبیعی سنتی مختلف برای شبکه نرم‌افزاری تعریف‌شده، تجدیدنظر شده و تجربه گسترش عملی خوبی را در تشخیص ترافیک غیرعادی شبکه نرم‌افزاری تعریف‌شده فراهم می‌کند. بااین‌حال، این روش‌ها نسبتاً ساده بوده و فقط برای برخی از سناریوهای خاص مناسب است. Zuo و همکارانش یک روش تشخیص غیرعادی ترافیک آنلاین Open TAD را پیشنهاد دادند. ابتدا رقم جدول جریان از کنترلر جریان جمع‌آوری گردیده و ماتریس ترافیک و ماتریس آنتروپی نمونه جمع‌آوری و مونتاژ می‌شود؛ سپس روش PCA جهت تشخیص ترافیک غیرطبیعی مورداستفاده است. OpenTAD ساده و مؤثر است و ناهنجاری‌های ترافیک می‌تواند به‌سرعت تفکیک‌شده باشد. این روش یک شیوه برای تشخیص غیرعادی ترافیک آنلاین خفیف برای شبکه نرم‌افزاری تعریف‌شده است.

ارسال داده در شبکه‌ها بستگی به انواع تجهیزات شبکه‌ای که انواع توابع را ارائه می‌دهند دارد و فرایندی پیچیده و آسیب‌پذیر برای خطاهای پیکربندی احتمالی، اشکالات نرم‌افزاری و مسائل دیگر است که احتمالاً باعث اشتباهات شبکه‌های مختلف همچون لوپ‌های مسیریابی می‌گردد. شبکه نرم‌افزاری تعریف‌شده اپلیکیشن‌های شبکه را ساده می‌کند، اما هنوز هم اجتناب از اشتباهات به دلیل پیچیدگی خود نرم‌افزار دشوار است؛ بنابراین، اعتبار سنجی و تشخیص وضعیت صحیح، کار تحقیقاتی مهمی در حوزه اندازه‌گیری ترافیک برای شبکه نرم‌افزاری تعریف‌شده است.

Mai و همکارانش، یک ابزار تشخیص وضعیت صحیح برای شبکه نرم‌افزاری تعریف‌شده بر مبنای تجزیه‌وتحلیل استاتیک لایه داده بنام Anteater را پیشنهاد دادند. کار اصلی فرایندهای Anteater به شرح زیر است:

اول: اتصال شبکه و سوئیچ‌های FIB را به حالت‌های Boolean تبدیل کرده و سپس به سه نوع از وضعیت‌های صحیح به‌نام‌های: مسیریابی بدون لوپ، اتصال به شبکه و ثبات قوانین در بیان مشکل SAT، تغییر شکل می‌دهد.

دوم: انتخاب یک حالت صحیح در انتظار بررسی است که این حالت و حالت‌های Boolean بالا و ورودی حل‌کننده SAT برای تجزیه‌وتحلیل ترکیب می‌شوند. چنانچه یک نقش پیکربندی وجود دارد که به وضعیت صحیح آسیب رساند، Anteater یک بسته خاص جهت تریگر مشکل مربوطه ایجاد می‌کند. در تست‌های واقعی، Anteater با موفقیت در یک شبکه پردیس تعداد ۲۳ مورد از مشکلات شبکه را کشف کرد. تشخیص بایستی تا جایی که امکان دارد به لایه داده شبکه نزدیک باشد، بنابراین مشاهده مستقیم‌تر رفتار شبکه و یافتن مشکلات قبل از پیکربندی مدت‌زمانی طول می‌کشد. به‌این‌ترتیب، Anteater همچنین تحلیل‌های یکپارچه را برای محیط‌هایی با زبان و پروتکل‌های متعدد پشتیبانی می‌کند.

Canini و همکارانش یک ابزار اعتبارسنجی شبکه مرکب از بررسی مدل تحت عنوان، NICE، پیشنهاد کردند که در زبان برنامه‌نویسی پایتون جهت پشتیبانی یکپارچه برنامه کنترل اوپن فلو توسعه‌یافته است. هدف اصلی آن، یافتن خطاها در اپلیکیشن اوپن فلو است. به‌عنوان‌مثال، پاسخ‌های کنترلر به یک جریان جدید و تولیدات یک قاعده جدید موردبحث است. در این مرحله، اگر جریان‌های بیش‌ازحد نیز به کنترلر برسد، ممکن است خطاهای اپلیکیشن رخ دهد. NICE برنامه اوپن فلو و وضعیت صحت شبکه را به‌عنوان پارامترهای ورودی در نظر می‌گیرد و صحت هر حالت را به‌وسیله جستجو در فضای حالت با استفاده از بررسی مدل تائید می‌کند. خروجی‌های نهایی وضعیت صحت و درستی را گزارش می‌دهند. با NICE، برنامه اوپن فلو می‌تواند لوپ‌های مسیریابی و سایر مشکلات شبکه را رسیدگی و بازرسی نماید.

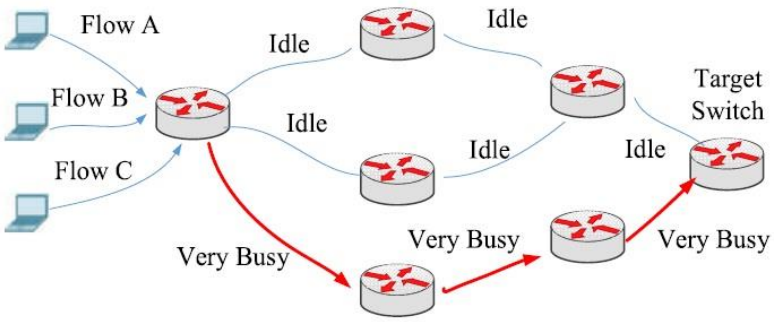
۳-۶-۱ مدیریت ترافیک در شبکه نرم‌افزاری تعریف‌شده

هدف مدیریت شبکه حفظ در دسترس بودن شبکه و بهبود کارایی شبکه است. برنامه‌ریزی مناسب ترافیک یک راه مهم جهت توسعه و بهبود QoS شبکه است. به‌طورکلی، مسیرهای متعددی بین منبع و گره مقصد در شبکه نرم‌افزاری تعریف‌شده وجود دارد که امکان برنامه‌ریزی و زمان‌بندی شبکه را فراهم می‌سازد. کنترلر شبکه نرم‌افزاری تعریف‌شده دیدگاه کلی استفاده از هر مسیر در شبکه را با استفاده از فن‌آوری‌های مختلف اندازه‌گیری شبکه ذکرشده در بالا، حفظ کرده. ما می‌توانیم یک الگوریتم زمان‌بندی ترافیک برای طرحی پویا در مسیرهای ارسال داده در مواجه با نیازهای کاربران، طراحی نماییم. در این بخش، فن‌آوری‌های مدیریت ترافیک در شبکه نرم‌افزاری تعریف‌شده را از جنبه‌های زیر مطرح می‌کنیم: تعادل بار ترافیک، برنامه‌ریزی ضمانت در QoS، برنامه‌ریزی صرفه‌جویی در انرژی و مدیریت ترافیک برای IP/SDN ترکیبی.

۳-۶-۲ تعادل بار ترافیک

فناوری مسیریابی چندگانه هم‌ارز (ECMP) مبتنی بر الگوریتم هش یک راه‌حل تعادل بار مؤثر است. با توجه به فناوری مسیریابی ECMP، ممکن است مسیرهای ارسال متعددی برای یک شبکه هدف وجود داشته باشد. هنگامی‌که بسته به یک سوئیچ

یا روتر می‌رسد، سوئیچ یا روتر فیلدهای هدر بسته را برای انجام محاسبات هش استخراج کرده و سپس با توجه به مقدار هش یکی از مسیرهای ارسال را انتخاب می‌کند که درنتیجه بسته‌های IP دارای هد مشابه در طول مسیر یکسان ارسال خواهند شد. ایراد واضح و روشن فناوری ECMP این است که بسیاری از جریان‌های بزرگ تحت عنوان جریان‌های فیل، در مسیر یکسان ارسال می‌شوند و همان‌طور که در شکل ۳-۴ نشان داده‌شده است، این عمل منجر به عدم تعادل بار و اتلاف پهنای باند می‌گردد.



شکل ۳-۴: سناریوی عدم تعادل بار در ECMP

یک راه‌حل ساده جدا کردن ترافیک از سطح بسته و نه سطح جریان است. این روش تا حد زیادی اثربخشی تعادل بار را بهبود می‌بخشد، اما به‌طورجدی مشکل مرتب کردن مجدد پنجره ارسال TCP را ارسال می‌کند. این مشکل باعث کوچک شدن غیرضروری پنجره ارسال TCP می‌شود. به‌منظور حل مشکل فوق، چندین طرح ECMP بهبودیافته ارائه‌شده است. مکانیسم تعادل بار ترافیک مبتنی بر فناوری زمان‌بندی wildcard است که در آن دو نوع راه‌حل معمولی Devoflow و DIFANE است. DevoFlow مدل اوپن فلو را برای بهبود صرفه‌جویی در منابع و طرح زمان‌بندی ترافیک مقیاس‌پذیر اصلاح می‌کند. DevoFlow به‌صورت ویژه پیشنهاد می‌کند که کنترلر باید مسیر جریان‌های فیل را به‌محض پیدا کردن آن‌ها تغییر دهد و تمام جریان داده را به‌عنوان جریان فیل بالقوه تلقی نکند. در این صورت می‌توان به‌صورت ویژه از تأخیرها جلوگیری کرده و هزینه‌ها را کنترل نماید. DevoFlow یک قاعده تطبیقی چند مسیره مبتنی بر wildcard ها را طراحی می‌کند. به‌طور پیش‌فرض، یک سوئیچ این قاعده را برای جریان‌های ارسال بکار می‌برد. در همان زمان، DevoFlow یک روش آماری ترافیک جهت تشخیص جریان‌های فیل را معرفی می‌کند که در آن جریان‌های فیل برای تغییر مسیرشان نیازمند کمک کنترلر هستند. DIFANE طرح زمان‌بندی ترافیک کارآمد دیگری است که مبتنی بر wildcard ها است. ایده اصلی DIFANE می‌تواند در دو جنبه خلاصه‌شده باشد. اول: کنترلر برخی از قواعد را برای یک مجموعه مجاز از سوئیچ‌ها توزیع نماید. دوم: سوئیچ‌ها به تمام بسته‌های موجود در لایه داده رسیدگی نماید. چنانچه جریان‌ها نتوانند قواعد محلی را تطبیق دهند، آن‌ها کپسوله شده و به یک مجموعه مجاز مناسب از سوئیچ‌ها هدایت می‌شوند.

۳-۶-۳ برنامه‌ریزی ضمانت کیفیت سرویس (QoS-GUARANTEE)

در یک شبکه، بسیاری از ترافیک‌های کسب‌وکار در زمان واقعی، مانند داده‌های صوتی، پیام‌های فوری و غیره، به تأخیر و از بین رفتن بسته در فرایند انتقال حساس هستند؛ بنابراین، زمان‌بندی مناسب و منطقی منابع شبکه جهت فراهم نمودن کیفیت سرویس (QoS) برای محیط کسب‌وکار از مهم‌ترین مشکلات مدیریت ترافیک است. شبکه نرم‌افزاری تعریف‌شده یک رابط کنترل باز برای حمایت از استراتژی زمان‌بندی ترافیک شبکه انعطاف‌پذیر را فراهم می‌کند؛ همچنین می‌تواند نیازهای کیفیت سرویس اپلیکیشن‌های شبکه مختلف را برآورده سازد.

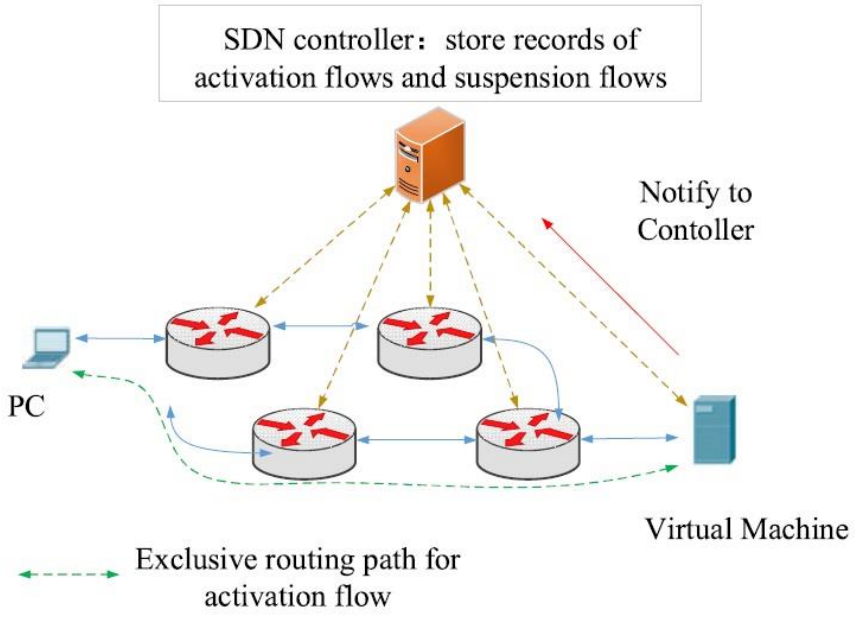
طرح OpenQoS باهدف فراهم آوردن ضمانت کیفیت سرویس برای توزیع جریان‌های کسب‌وکارهای چندرسانه‌ای ارائه‌شده است. ازآنجاکه هدهای بسته در جریان‌های کسب‌وکارهای چندرسانه‌ای با سایر هدهای بسته خیلی متفاوت هستند، OpenQoS تمام ترافیک داده را با استفاده از قواعد تطبیق پیکربندی اوپن فلو، به دو گروه جریان‌های چندرسانه‌ای و داده تقسیم می‌کند. OpenQoS عملکرد مسیرهای ارسال را ازنظر تأخیرها و تلفات بسته مشاهده کرده و بهترین مسیری را که می‌تواند نیازمندی‌های QoS را برآورده سازد، انتخاب می‌کند. سایر جریان‌های داده هنوز هم در مسیر اصلی ارسال می‌شوند؛ اما OpenQoS تنها زمان‌بندی جریان‌های چندرسانه‌ای را بهینه‌سازی نموده و جریان‌های کسب‌وکار دارای الزامات QoS متعدد را در نظر نمی‌گیرد. Ongaro و همکارانش، QoS را یک چارچوب افزایش بر اساس زمان‌بندی ترافیک متمرکز برای شبکه نرم‌افزاری تعریف‌شده ارائه داده‌اند. مسائل خاصی که این چارچوب بر روی آن متمرکز است به شرح زیر است. اول: اینکه در شبکه‌ای که گره‌ها یا لینک‌ها دارای تداخل هستند، مسیرهای مناسب برای جریان‌های کسب‌وکار اپلیکیشن‌های متعدد، چطور انتخاب می‌شود. دوم: تحت الزامات QoS، مجموع پهنای باندهایی که برای هر مسیر اختصاص داده‌شده است نمی‌تواند از ظرفیت پیوند فیزیکی تجاوز نماید. به‌منظور حل این مشکلات، این چارچوب چند مؤلفه کلیدی را با عناوین زیر طراحی می‌کند. یک ماژول ترسیم توپولوژی شبکه، ماژول گردآوری اوضاع شبکه، ماژول انتخاب مسیر و ماژول پیکربندی مسیر پویا. دو ماژول اول مسئول نظارت بر لایه داده، برای گرفتن به‌روزرسانی‌های توپولوژی، جمع‌آوری پارامترهای شبکه به‌صورت پویا و تولید وزن هر مسیر در دیاگرام شبکه هستند. ماژول انتخاب مسیر، مسیرهای مناسب را با توجه به نیازهای QoS کسب‌وکار و وزن دیاگرام شبکه انتخاب می‌کند. ماژول پیکربندی مسیر پویا مسئول به‌روزرسانی قواعد مسیریابی در لایه ارسال داده در زمان مناسب است. ماژول انتخاب مسیر، به‌عنوان ماژول اصلی این چارچوب، تصمیمات برنامه‌ریزی را با استفاده از مدل جریان چند کالا و کوتاه‌ترین مسیر محدود (MCFCSP)، اخذ می‌کند. بر اساس محدودیت‌ها در متغیرهای پارامتری ارائه‌شده توسط این مدل، ما می‌توانیم برای استراتژی زمان‌بندی سرا سری، مشکلات کامل NP را حل کنیم.

مؤلف تائید می‌کند که این راه‌حل بر اساس پلتفرم Mininet قادر است الزامات QoS اپلیکیشن‌های کسب‌وکار در زمان واقعی، شامل ویدئو و انتقال فایل را به‌طور مؤثر برآورده نماید.

۳-۷ برنامه‌ریزی صرفه‌جویی در انرژی

با توجه به اقتصاد و حفاظت از محیط‌زیست، کاهش مصرف انرژی به یک مسئله مهم در سال‌های اخیر تبدیل‌شده است. اطلاعات و تکنولوژی‌های ارتباطی به‌طور گسترده‌ای در زندگی روزمره مورداستفاده است. مصرف انرژی در حوزه شبکه‌های مخابراتی ۵ درصد از تمام مصرف انرژی را در کشورهای توسعه‌یافته در برمی‌گیرد. این مقدار با سرعت ۱۰ درصد در سال در حال رشد است؛ بنابراین این پرسش معنی‌دار مطرح است که چطور مصرف انرژی شبکه را کاهش دهیم. در سال‌های اخیر، راه‌حل‌های زیادی راجع به صرفه‌جویی در مصرف انرژی شبکه ارائه‌شده است. ازآنجاکه سرورها مصرف‌کنندگان اصلی حال حاضر انرژی هستند، تحقیقات اندکی، برای حل مشکل مصرف انرژی در شبکه‌های مرکز داده وجود دارد. بااین‌وجود، درصد مصرف انرژی شبکه ممکن است تا ۵۰ درصد افزایش یابد؛ بنابراین، شبکه نرم‌افزاری تعریف‌شده باید نه‌تنها به‌کارگیری لینک، تعادل بار و الزامات QoS را در نظر بگیرد، بلکه باید ملزومات مصرف انرژی در فرایند مدیریت ترافیک را نیز در نظر داشته باشد. مصرف انرژی شبکه به‌طور عمده به واحدهای شبکه مانند لینک‌ها، سوئیچ‌ها و غیره بستگی دارد. در حال حاضر، دو نوع از روش‌های صرفه‌جویی در مصرف انرژی وجود دارد. نرخ لینک تطبیقی و مدل‌های خواب. روش‌های تطبیقی نرخ لینک به‌طور پویا نرخ لینک را با توجه به تقاضای ترافیک تنظیم می‌کنند. ازآنجایی‌که مصرف انرژی لینک‌ها به مقدار بارهای انتقال داده آن‌ها بستگی دارد، از سوی دیگر، مدل خواب، برخی از اجزای شبکه را خاموش می‌کند و یا برخی از اجزای غیرفعال را جهت صرفه‌جویی در انرژی به حالت خواب درمی‌آورد.

Li و همکارانش یک استراتژی صرفه‌جویی انرژی مبتنی بر مسیریابی انحصاری را پیشنهاد کرده‌اند. این استراتژی نیازی به همگام‌سازی زمانی سرورها ندارد و می‌تواند وضعیت جریان سرا سری را با استفاده از شبکه نرم‌افزاری تعریف‌شده مدیریت کند و صریحاً ماشین‌های مجازی را برای اجازه یا رد یک جریان خاص آگاه نماید. هنگامی‌که یک جریان جدید به یک ماشین مجازی از سرور می‌رسد، آنگاه ماشین مجازی به کنترلر شبکه نرم‌افزاری تعریف‌شده گزارش می‌کند. کنترلر شبکه نرم‌افزاری تعریف‌شده الگوریتم مسیریابی انحصاری را اجرا کرده و جریان‌های فعال و معلق را به‌روزرسانی می‌کند. چنانچه جریان جدیدی فعال‌شده باشد، کنترلر شبکه نرم‌افزاری تعریف‌شده ماشین مجازی را برای اجازه دادن به این جریان آگاه کرده و آیتم‌های مربوطه را در جدول جریان آن سوئیچ درج می‌کند. اگر جریان فعال به حالت معلق تغییر یابد، آنگاه کنترلر شبکه نرم‌افزاری تعریف‌شده ماشین مجازی را برای رد این جریان مطلع می‌کند. اصول مسیریابی انحصاری در شکل ۵ نشان داده‌شده است.



شکل ۵-۳: چارچوب مسیریابی انحصاری برای شبکه نرم‌افزاری تعریف‌شده

با توجه به الگوریتم مسیریابی انحصاری، هر جریان فعال یک لینک مسیریابی را به انحصار خود درمی‌آورد، بنابراین نیازی به رزرو پهنای باند برای یک جریان خاص نیست. صرفه‌جویی انرژی شبکه در اصل به استفاده بالای نرخ لینک‌ها بستگی دارد. برای الگوریتم مسیریابی انحصاری، چنانچه مسیرهای بیکاری وجود داشته باشد، جریان‌ها این مسیرها را انتخاب می‌کنند. از سوی دیگر، اگر جریانی دارای اولویت پایین‌تری باشد، آنگاه این جریان معلق خواهد بود.

Amokrane و همکارانش یک روش گزینش مسیریابی آنلاین را ارائه نموده‌اند که هدف آن به حداقل رساندن مصرف انرژی در یک شبکه MAPs بی‌سیم است. این روش می‌تواند به‌راحتی با شبکه نرم‌افزاری تعریف‌شده ادغام گردد؛ چراکه متکی بر یک کنترلر مرکزی است که بر کل شبکه نظارت و مدیریت کرده و مسیرهای روتینگ را اتخاذ می‌کند. این مقاله این مشکل را به‌عنوان یک مشکل طرح‌ریزی خطی عدد صحیح مطرح می‌کند. به‌عنوان تابع هدف، هزینه حالت‌های خواب و فعال شبکه‌های MAPs را مطرح کرده و همچنین محاسبه مسیریابی مجدد یا ادغام جریان‌های موجود طول می‌کشد. ازآنجایی‌که این مشکل یک مشکل NP سخت است، مؤلف یک الگوریتم کلونی مورچه‌ها (ACOFER) را که ساده اما مؤثر است را ارائه می‌کند. ACOFER در مقایسه با سایر الگوریتم‌ها، منجر به اثربخشی بهتر در ترم‌های صرفه‌جویی انرژی می‌گردد.

۳-۸ مدیریت ترافیک برای IP/SDN ترکیبی

تکنولوژی‌های مهندسی ترافیک مبتنی بر شبکه نرم‌افزاری تعریف‌شده راه‌حل‌های مؤثری برای بهینه‌سازی شبکه فراهم می‌آورند. هرچند روند جابجایی شبکه‌های سنتی به شبکه نرم‌افزاری تعریف‌شده مدت‌زمان زیادی طول می‌کشد، بنابراین شبکه

ترکیبی IP/SDN پدید آمده است. هیچ گره‌ای از یک شبکه ترکیبی IP/SDN دارای ویژگی‌های یک سوئیچ نبوده و ما فقط یک دیدگاه کلی دقیق از شبکه را تا حدی به دست می‌آوریم. مدیریت ترافیک شبکه‌های ترکیبی IP/SDN در مقایسه با شبکه‌های شبکه نرم‌افزاری تعریف‌شده بسیار مشکل‌تر است؛ بنابراین، این مسئله که چطور ترافیک را در شبکه ترکیبی IP/SDN مدیریت کنیم به یک چالش تبدیل می‌شود.

Agarwal و همکارانش، یک معماری مدیریت ترافیک که برای شبکه ترکیبی IP/SDN مفید است را ارائه می‌کنند. این معماری سعی در نفوذ کنترلر متمرکز جهت دستیابی به بهره‌برداری بالا در شبکه و همچنین کاهش تلفات و تأخیرهای بسته را دارد. ابتدا مشکلات بهینه‌سازی شده‌ی کنترلر شبکه نرم‌افزاری تعریف‌شده برای مدیریت ترافیک با گسترش نسبی فرموله شده است؛ سپس طرح‌های تقریبی زمانی چندجمله‌ای کامل (FPTAS) جهت حل این مشکلات طراحی‌شده است. آزمایش‌ها نشان می‌دهد که با استفاده از این الگوریتم‌ها، افزایش کارایی در محیط شبکه‌های ترکیبی IP/SDN دست‌یافتنی است.

Guo و همکارانش موضوعات مهندسی ترافیک در شبکه‌های ترکیبی IP/SDN را کنکاش نموده‌اند. فرض بر این است که وزن‌های OSPF و نرخ‌های تقسیم جریان سوئیچ شبکه نرم‌افزاری تعریف‌شده هر دو قادر به تغییر هستند و کنترلر می‌تواند جریان‌های ورودی به سوئیچ شبکه نرم‌افزاری تعریف‌شده را به‌طور دلخواه تقسیم نماید. گره‌های شبکه سنتی هنوز هم OSPF را اجرا می‌کند. SOTE یک الگوریتم بدیع برای دستیابی به به‌کارگیری لینک ماکزیمم پایین‌تر پیشنهادشده است. نتایج نشان می‌دهد که وقتی تنها ۳۰ درصد از گره‌های شبکه نرم‌افزاری تعریف‌شده گسترش‌یافته‌اند، عملکرد مطلوب مدنظر به‌دست‌آمده است.

Guo و همکارانش برای یک توالی انتقال بهینه از روترهای میراث به روترهای فعال‌شده در شبکه نرم‌افزاری تعریف‌شده تحقیق نموده‌اند، به‌طوری‌که می‌توان تصمیم گرفت که تعداد زیادی از روترها به کجا و چطور انتقال یابند. نویسندگان الگوریتمی ابتکاری برای یافتن یک دنباله انتقال روترها را پیشنهاد کرده‌اند که مزایای بسیاری را از منظر مهندسی ترافیک فراهم می‌کند. این الگوریتم توسط شبیه‌سازها ارزیابی شده بود. آزمایش‌ها نشان می‌دهد که الگوریتم پیشنهادی، سریع‌تر از سایر الگوریتم‌های انتقال در این تحقیق برای یک دنباله انتقال است. اگر گره‌های شبکه بتواند به‌درستی مستقرشده باشد، تقریباً ۴۰ درصد از روترها به بسیاری از مزایا منجر خواهد شد.

نتیجه‌گیری و پیشنهادها

توسعه فن‌آوری مهندسی ترافیک شبکه نرم‌افزاری تعریف‌شده، رویکردی مهم برای ترویج کاربرد متداول شبکه نرم‌افزاری تعریف‌شده است. در این مقاله، چارچوبی را برای مهندسی ترافیک در SD پیشنهاد کردیم و فن‌آوری‌های مرتبط را از دو منظر بحث کردیم: اندازه‌گیری ترافیک و مدیریت ترافیک. در موضوع اندازه‌گیری شبکه، پارامترهای شبکه مرتبط در هر لایه از شبکه وجود دارد که وضعیت شبکه را نشان می‌دهد. به‌طورکلی، پارامترهای لایه کاربرد و لایه شبکه را نظارت می‌کنیم. جدول ماتریس ترافیک، اطلاعات ایده الی برای انعکاس وضعیت فعلی شبکه است، اما فن‌آوری‌های پیش‌بینی‌شده‌ی ماتریس ترافیک به تحقیقات بیشتری نیاز دارد. در موضوع مدیریت شبکه، هدف اصلی، فراهم آوردن استراتژی‌های زمان‌بندی ترافیک منطقی با توجه به آن پارامترهایی از شبکه است که با استفاده از فن‌آوری‌های اندازه‌گیری جریان جهت برآوردن نیازهای خاص به‌دست‌آمده است. ما چهار منظر به نام‌های، تعادل بار ترافیک، ضمانت کیفیت سرویس، برنامه‌ریزی صرفه‌جویی انرژی و مدیریت ترافیک شبکه ترکیبی IP/SDN را موردبحث قراردادیم. در حقیقت، برخی از مسائلی وجود دارد که درگیر موضوع نیستند، مانند مهندسی ترافیک مبتنی بر MPLS در شبکه نرم‌افزاری تعریف‌شده و تحلیل عملکرد مهندسی ترافیک که احتمالاً به موضوعات داغی در آینده تبدیل خواهند شد.

پیوست‌ها

منابع و مآخذ

فهرست منابع فارسی

فهرست منابع لاتین

سایت‌های اطلاع‌رسانی