به نام حق

دانشکده فنی و مهندسی
کارشناسی ارشد مهندسی کامپیوتر نرم‌افزار
گروه مهندسی کامپیوتر و فناوری اطلاعات

گزارش درس سمینار

موضوع:

درخت عملکرد سرویس امنیتی مبتنی بر SDN / NFV برای CLOUD

نگارش:

سیما محسنی

استاد راهنما:

جناب آقای دکتر آرش قربان نیا دلاور

آذر 1399

چکیده:

امنیت شبکه برای رایانش ابری بسیار مهم است. زنجیره عملکرد خدمات (SFC) که با شبکه مبتنی بر نرم افزار (SDN) و مجازی سازی عملکرد شبکه (NFV) یکپارچه شده باشد می تواند رویکردی جدید برای حل مسائل امنیتی شبکه برای رایانش ابری ارائه نماید در این مقاله ، ما چندین SFC را در یک درخت عملکرد سرویس امنیتی (یا به طور خلاصه SecSFT) برای کاهش نیاز به منابع در اختصاص توابع امنیتی مجازی ترکیب می کنیم.با توجه به ایده استفاده از درخت تصمیم برای طبقه بندی ، ما قوانین تصمیم و قوانین تشخیص را در گره های SecSFT تعیین می کنیم تا بتوانند آنها را شناسایی و روند های مشکوک را از ترافیک مختلط جدا نموده و از موارد مشکوک شناسایی و جلوگیری می کند. گره های SecSFTتوابع مجازی مختلف از جمله توابع شبکه مربوط به امنیت را پیاده سازی کنید (به عنوان مثال ، load balancing ،وtraffic shaping) ، توابع امنیتی شبکه (به عنوان مثال ، تشخیص نفوذ ، فایروال) و سخت افزار امنیت شبکه مجازی سازی. سرانجام ، ما یک SecSFT را در یک ابر آزمایشی ساخته وسرویس های امنیت آن را در شناسایی و کاهش حملات شبکه آزمایش و اعتبار سنجی می کنیم.

عبارات کلیدی:

ابر ، درخت تصمیم ، تشخیص نفوذ ، مجازی سازی عملکرد شبکه ، درخت عملکرد سرویس امنیتی، شبکه مبتنی بر نرم افزار.

مقدمه:

استفاده گسترده از خدمات رایانش ابری یک سری مشکلات امنیتی را به ارمغان آورده است. با توجه به انتزاع و جدا کردن محاسبات ، ذخیره سازی و منابع شبکه از دستگاه های سخت افزاری اختصاصی ، امنیت شبکه سنتی بین رفته است.مسائل امنیتی که رایانه ابری با آن روبرو است از هر دو نوع داخلی و خارجی ناشی می شود. تهدیدهای امنیتی داخلی ناشی از کمبود ایزوله سازی در میزبان های ابر است و تهدایات خارجی از تمامی مسیرها به محیط ابر آمده است. از این رو، امنیت رایانش ابری به یک نگرانی عمومی برای دولت ، شرکت ها و دانشگاه ها تبدیل شده است. شبکه مبتنی بر نرم افزار (SDN) یک معماری نوظهور قابل کنترل ، مقرون به صرفه و سازگار است. با قابلیت جداسازی برنام کنترل از برنامه داده این قابلیت را دارد که این دارد که یک کنترل کننده خارجی بتواند ترافیک شبکه را به صورت یکپارچه مدیریت کند [1].عملکرد شبکه مجازی (VNF) [2] برای اولین بار در یک مقاله سفید NFV (مجازی سازی عملکرد شبکه) منتشر شده توسط ETSI (موسسه استاندارد مخابرات اروپا) دراکتبر 2010 پیشنهاد گردید.. ماهیت NFV جدا کردن عملکردهای شبکه از دستگاه های شبکه اختصاصی از طریق مجازی سازی است. دستگاه های شبکه اختصاصی سنتی مانند فایروال ، بازرسی بسته عمیق ، تشخیص نفوذ و مترجم آدرس شبکه، بصورت VNF مجازی شده ، که می توانن در سراسر یک ابر برای پیاده سازی شبکه مربوطه خدمات امنیتی مستقر شوند.

ادغام SDN و NFV باعث قدرتمند شدن مجازی سازی و بهبود خدمات شبکه می شود. زنجیره عملکرد خدمات (SFC) مکانیزمی است که قابلیت تشخیص یک لیست از عملکردهای سرویس را فراهم کرده و به طور پویا ترافیک شبکه از طریق مسیرهای مختلف عملکرد سرویس راهبری می کند [3]. بنابراین ، یک SFCمی تواند برای پیاده سازی توالی های امنیتی مجازی به منظور ارائه خدمات امنیتی برای اجاره کنندگان ابر ایجاد شود. با این حال ، برای جلب رضایت انواع اجاره کنندگان و جلوگیری از آن انواع حملات ، بسیاری از سرویس های امنیتی زنجیره ای مورد نیاز هستند. فعالیت های موجود که از SFC برای تهیه سرویس های امنیتی استفاده می شوند بیش از ابر همچنان به روشی یک SFC در هر سرویس امنیتی محدود هستند [4] - [15]. . در این مقاله ، ما برای غلبه بر این محدودیت ، معماری درختی تابع سرویس امنیتی جدید (یا به اختصار SecSFT) را پیشنهاد می دهیم. در معماری thenovel ، گره های VNF در چندین SFC در صورت اجرای همان عملکرد امنیتی شبکه ، در یک گره VNF ادغام می شوند.

چند مورد مرتبط وجود دارد که بر کشف و جلوگیری از چند نوع حمله با استفاده از SFC متمرکز هستند [4] - [7]. از آنجا که یک عملکرد امنیتی مجازی با عملکرد کامل ممکن است به یک محاسبه ظرفیت بالا احتیاج داشته باشد و ترافیک منجر به آن شود که مقدار زیادی از منابع شبکه را مصرف کند ، بهینه سازی تخصیص منابع و افزایش عملکرد یک SFC مهمترین نگرانی است [8] -

[15]. در مقابل ، ما با ادغام گره های مشابه VNF چندین SFC و اختصاص عملکردهای امنیتی مجازی نزدیک به اهداف ، نیاز به منابع را کاهش می دهیم. SFC های ادغام شده یک درخت تصمیم توزیع شده تشکیل می دهند و عملکرد سرویس های امنیتی به دلیل روش پردازش توزیع شده افزایش می یابد. از آنجا که وضعیت امنیتی در فضای ابری پیچیده است و حملات شبکه می تواند چندین مورد را هدف قرار دهد ، بنابراین درخت تصمیم توزیع شده از یک زنجیره مناسب تر است.

مشارکت اصلی مقاله ما می تواند به شرح زیر باشد:

(1) با توجه به اینکه بسیاری از زنجیره های عملکرد سرویس های امنیتی برای جلب رضایت اجاره کنندگان مختلف و جلوگیری از انواع حملات به ابر مورد نیاز است ، و بیشتر گره های VNF در میان زنجیره ها همان کارها را با نرخ استفاده مجدد بسیار کم انجام می دهند ، ما یک معماری جدید SecSFT را پیشنهاد می دهیم. همان گره های VNF عملکردی تا آنجا که ممکن است ادغام می شوند ، و درخت / درختان به معنای مصرف منابع و تحویل کارآمد و فیلتر کردن ترافیک شبکه ، بهینه سازی می شوند.

(2) ما یک مدل توزیع شده برای SecSFT با توجه به الگوریتم طبقه بندی درخت تصمیم ، که قوانین تصمیم را به گره های VNF مربوط به SecSFT مرتبط می کند ، پیشنهاد می دهیم. مقادیر ویژگی جریان شبکه در هر یک از گره های VNF جمع آوری و تحلیل می شود. جریان های شبکه با تطبیق قوانین تصمیم گیری شناسایی و تقسیم می شوند و جریان های مشکوک شبکه در گره فعلی شناسایی و فیلتر می شوند یا برای تقسیم و ریزبینی بیشتر به گره های بعدی ارسال می شوند.

I. کارهای مرتبط

چند مورد مرتبط با استفاده از SFC برای ارائه خدمات امنیتی برای شبکه ها وجود دارد. به عنوان مثال ، زینگ و همکاران [4] یک چارچوب سیستم تشخیص نفوذ ، SnortFlow ، با استفاده از ترکیبی از OpenFlow و Snort در یک محیط ابر ارائه داده اند. Snort برای شناسایی رفتارهای نفوذی در پیام های شبکه استفاده می شود. کنترل کننده SDN مسئول توزیع جداول جریان برای پیکربندی مجدد شبکه است. Phan و Park [5] یک راه حل برای مقابله با حملات DDoS در محیط ابر مبتنی بر SDN ارائه دادند. طبقه بندی ترافیک بر اساس ماشین بردار پشتیبان و الگوریتم های نقشه خود سازماندهی شده و شناسایی حمله توسط یک طرح فیلتر شده IP مبتنی بر سابقه انجام می شود.

SFC در ابر مبتنی بر SDN برای دفاع از حمله سطح مختلف DDoS شکل گرفت. نگوین و همکاران [6] از DES، AES و الگوریتم های رمزگذاری دیگر برای رمزگذاری سرصفحه های پیام که حاوی اطلاعات مسیر حمل و نقل در پیام های SFC است ، استفاده می

کند ، در نتیجه از شنود اطلاعات یا حمله مرد میانی جلوگیری می کند. لی و همکاران [7] یک طرح انتخاب خودکار برای زنجیره عملکرد سرویس های امنیتی ارائه داده است که با استفاده از الگوریتم یادگیری تقویت کننده یادگیری Q برای تجزیه و تحلیل و وزن دادن به حالت های شبکه ، به منظور دستیابی به یک راه حل متنوع برای دفاع از شبکه ، استفاده می کند. بر خلاف کارهای موجود که از یک SFC برای ارائه سرویس امنیتی استفاده می کنند ، ما از درختی استفاده می کنیم که چندین SFC را با هم ترکیب می کند تا مجموعه ای از خدمات امنیتی ابر را فراهم کند.

تفاوت بین کارهای کمتر متمرکز بر موضوعات امنیتی در زنجیره گذاری سرویس های امنیتی ، و کراهای اصلی تر بهینه سازی تخصیص منابع برای زنجیره های عملکرد سرویس های امنیتی و ارتقا performance عملکرد آنها است. به عنوان مثال ، Ye [8] طرحی را پیشنهاد کرد که به شما اجازه می دهد هر دو VNF یک سرور را به اشتراک بگذارند تا مصرف منابع پهنای باند را به حداقل برسانند. لین و همکاران [9] برای هماهنگی در استقرار سرویس شبکه ، یک مدل بازی ساخت. لیو و همکاران [10] یک راه حل معماری سیستم ارائه داده است که از SDN و NFV برای کنترل منابع امنیتی مجازی برای ساخت سرویس های عملکرد امنیتی استفاده می کند. دویاردیکا و تاچیبانا [11] جایگزینی بهینه توابع شبکه مجازی امنیتی را برای زنجیره های عملکرد خدمات امنیتی بر اساس سطح امنیت ارائه دادند. Shameli-Sendi و همکاران [12] یک رویکرد امنیتی برای زیرساخت های ابری ارائه داد که شامل بهترین روش ، دانش فنی کارشناسان امنیتی و محدودیت های مختلف امنیتی در یک الگوی امنیت شبکه است. اعتقاد بر این است که قرارگیری مطلوب توابع امنیتی سازگار در ابر یک مشکل NP-Hard است و بنابراین یک چارچوب بهینه سازی آگاه از شبکه و محاسبات مقیاس پذیر در کار پیشنهاد شده است. راه حل ابتکاری مبتنی بر الگوریتم جستجو برای اولین بار توسط لیو و همکاران ارائه شد. [13] برای بهینه سازی تخصیص منابع با محدودیت هایی که امنیت و منابع مورد نیاز را نقض نمی کند. پی و همکاران [14] محل VNF پویا را در سیستم ابر جغرافیایی توزیع شده مطالعه کرد. این به عنوان یک مدل برنامه نویسی باینری یکپارچه برای به حداقل رساندن هزینه جاسازی در درخواست های جاسازی SFC و بهینه سازی تعداد نمونه های VNF قرار گرفته ، فرموله شده است. لی و همکاران [15] SFC مبتنی بر زمینه و پویا را از طریق شبکه های چند دامنه ای با اختصاص فراداده برای به اشتراک گذاشتن اطلاعات زمینه بسته ها در میان آن شبکه ها ، تحقق بخشید. بر خلاف کارهای موجود ، ما از معماری درخت برای کاهش مصرف منابع در استفاده مجدد از توابع امنیتی مجازی یکسان استفاده می کنیم.

معماری درخت خدمات امنیتی

در این بخش ، ما معماری درخت عملکرد سرویس امنیتی (SecSFT) را پیشنهاد می دهیم که چندین SFC را برای ارائه انواع خدمات امنیتی و استفاده مجدد از منابع امنیتی ترکیب می کند. این امکان را می دهد تا به طور انعطاف پذیر در نزدیکی چندین هدف مستقر شود تا از چندین جهت حملات شبکه را شناسایی و جلوگیری کند.

- درخت تصمیم

ما درخت عملکرد را بر اساس درخت تصمیم برای طبقه بندی ترافیک حمله شبکه طراحی می کنیم. به همین ترتیب قوانین درخت تصمیم را به گره های VNF SecSFT اختصاص دهید. به طور خاص ، الگوریتم درخت تصمیم C4.5 [16] برای ساخت SecSFT استفاده می شود. الگوریتم C4.5 مقدار ویژگی هر گره داخلی را با توجه به نسبت کسب اطلاعات تعیین می کند. برای جلوگیری از تناسب بیش از حد ، الگوریتم Pessimistic Error Pruning (PEP) را اتخاذ می کنیم [17]. PEP به مجموعه داده های اضافی آزمون ، هرس از بالا به پایین نیاز ندارد. برای یک گره برگ به شماره i با نمونه های ni و خطاهای ei ، میزان خطا (ei + 0.5) / ni بود ، جایی که 0,5 عامل مجازات است. سپس برای یک زیر درخت با گره های برگ L ، میزان قضاوت غلط آن است:

$$\text{ErrorRatio} = \left(\sum_{i=1}^{L} e_i + 0.5L\right) / \sum_{i=1}^{L} n_i$$

وقتی ErrorRatio از یک نمونه برابر با ۱ باشد ، به این معنی است که subtree کلاس را به اشتباه طبقه بندی کرده است. وقتی مقدار برابر با ۰ باشد ، نمونه به درستی طبقه بندی می شود. تعداد داوری های نادرست از زیرشاخه را می توان با توزیع برنولی بیان کرد. میانگین و انحراف معیار قضاوت نادرست برای زیرشاخص عبارتند از:

$$\text{ErrorMean} = \text{ErrorRatio} \times \sum_{i=1}^{L} n_i$$

$$\text{ErrorSTD} = \sqrt{\text{ErrorRatio} \times \sum_{i=1}^{L} n_i \times (1 - \text{ErrorRatio})}$$

درخت فرعی را با گره برگ جایگزین کنید ، سپس میزان خطای گره برگ:

$$ErrorRatio' = \left(e' + 0.5\right)/n', \qquad (1)$$

و میانگین تعداد داوری های اشتباه در گره برگ است:

$$ErrorMean' = ErrorRatio' \times n'.$$

بنابراین ، هنگامی که زیر درخت اصلی شرایط زیر را داشته باشد ، شاخه فرعی هرس می شود:

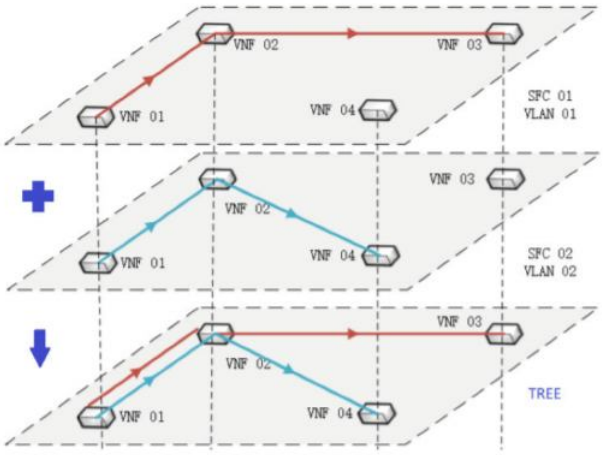
$$ErrorMean + ErrorSTD \geq ErrorMean'.$$

- **ساخت SecSFT**

معماری SecSFT دارای یک صفحه کنترل و مدیریت است که از ارکستراسیون و ماژول های نظارت جهانی تشکیل شده است. ماژول نظارت جهانی عمدتا چشم انداز جهانی را برای کنترل و مدیریت فراهم می کند ، کل توپولوژی شبکه را جمع آوری می کند ، منابع محاسباتی موجود در شبکه را پیدا می کند و کل وضعیت امنیت شبکه را بدست می آورد. ماژول ارکستراسیون به طور عمده با توجه به وضعیت امنیتی موجود ، برنامه های ارکستراسیون SecSFT ها را فرموله می کند ، نقشه برداری و برنامه ریزی منابع VNF را با توجه به طرح های ارکستراسیون و منابع موجود انجام می دهد و جداول جریان را برای ساخت توپولوژی های درخت تعریف می کند تا جریان داده ها از طریق SecSFT منتقل شود. کنترل کننده SDN نقش مهمی در صفحه کنترل و مدیریت دارد. کنترل کننده SDN از پروتکل شبکه OpenFlow [18] برای برقراری ارتباط با دستگاه های شبکه استفاده می کند. چندین کنترل کننده محبوب SDN مانند POX، Floodlight، OpenDayLight، ONOS و غیره وجود دارد. با در نظر گرفتن عملکرد در بین این کنترل کننده ها ، ONOS در این مقاله انتخاب می شود. ONOS از مزایای اکوسیستم منبع باز ، پشتیبانی از محیط خوشه کنترل کننده و قابلیت برنامه نویسی کامل برخوردار است. کنترل کننده SDN می تواند جداول جریان مختلف را با توجه به انواع سرویس های امنیتی و توپولوژی درختان به سوئیچ های OpenFlow ارسال کند. وقتی ترافیک شبکه به ورودی SecSFT رسید ، طبق قوانین تصمیم گیری به شاخه های مختلف درخت هدایت می شود. معماری SecSFT از JSON (JavaScript Object Notation) به عنوان قالب استاندارد برای ذخیره سازی و تبادل داده استفاده می کند. حالت ضبط متن از مزایای ساختار واضح و لایه های مختصر برخوردار است. این می تواند برای سوابق ساختار توپولوژیکی شبکه ، به روزرسانی منابع زیرساختی و الگوهای جداول جریان OpenFlow و غیره استفاده شود. ما برای هر SFC یک شبکه مجازی ایجاد می کنیم تا اطمینان حاصل شود که با شناسایی شناسه های شبکه منطقی مجازی ، می توان ترافیک شبکه عبوری از زنجیره های مختلف را جدا کرد. برای شناسایی و جلوگیری از حملات شبکه ، منابع امنیتی ابر در نزدیکی اهداف بالقوه حمله مستقر شده و در SecSFT ادغام می شوند. همانطور که

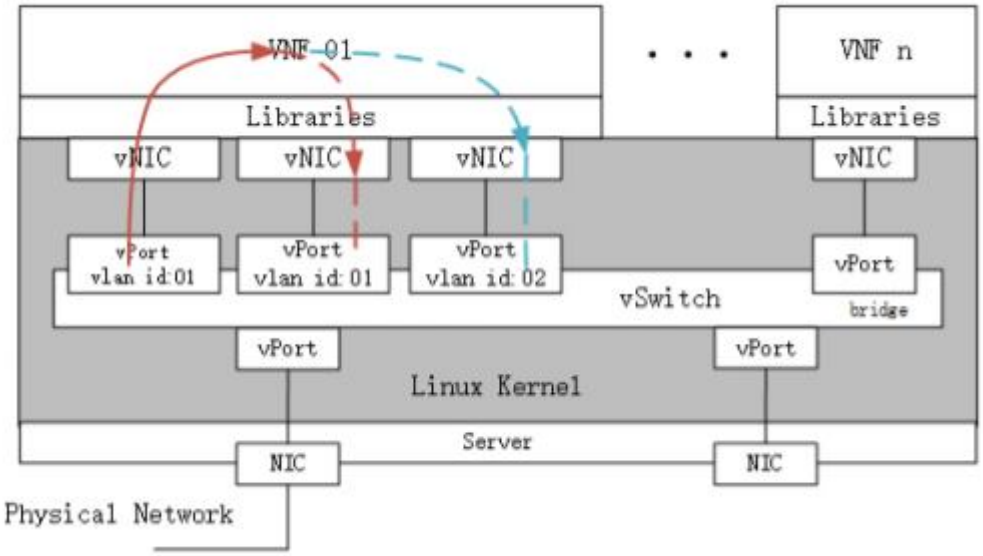
شکل 1 نشان می دهد ، دو SFC در یک ابر در چندین دامنه شبکه مستقر شده اند. دو VNF اول در دو SFC نشان داده شده در شکل 1 دارای عملکردهای امنیتی شبکه یکسانی هستند ، بنابراین آنها در همان VNF ادغام می شوند.

شکل1- دو زنجیره ادغام شده در یک درخت



رویکرد ادغام در شکل 2 نشان داده شده است. ما چندین کارت شبکه مجازی (vNIC) برای VNF ادغام شده ایجاد می کنیم. پورت های شبکه مجازی (vPorts) مربوط به این vNIC ها روی پل مجازی سوئیچ مجازی نصب شده اند. شناسه های VLAN برای زیرشبکه های منطقی مجازی که به vPorts متصل هستند تنظیم شده اند. هنگامی که یک فریم از پورت ورودی به VNF می رسد ، شناسه VLAN در هدر آن در vPort سلب می شود. بعد از پردازش فریم در VNF ، یک vPort به عنوان درگاه فوروارد انتخاب می شود و یک شناسه VLAN جدید متعلق به شبکه مجازی بعدی قبل از ارسال به هدر فریم اضافه می شود.

شکل2- شبکه های مختلف منطقی مجازی در یکVNF پل می شوند.



- **جمع آوری و طبقه بندی ترافیک**

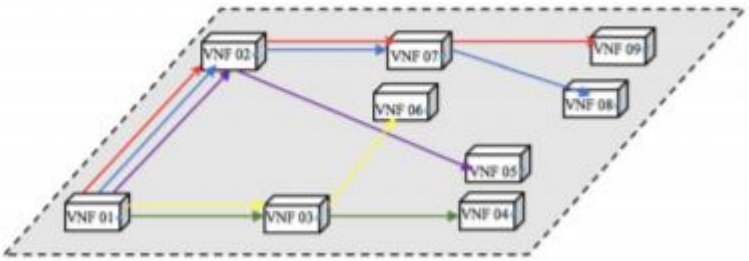
انواع حملات معمول شبکه شامل DoS (انکار سرویس) ، اسکن پورت ، دسترسی از راه دور غیر مجاز و دسترسی فوق العاده کاربر غیر مجاز و غیره است. در این مقاله ، ما چهار حمله شبکه خاص ، SYN Flood، UDP Flood، IPsweep و Portscan را برای آزمایش معماری SecSFT انتخاب می کنیم. وقتی حمله شبکه جدیدی رخ می دهد ، مقادیر ویژگی ترافیک شبکه با ترافیک عادی شبکه در برخی از گره های شبکه تفاوت زیادی خواهد داشت. با اشاره به نفوذ شبکه ارائه شده توسط KDDCUP 99 [19] ، تا 41 ویژگی از رفتارهای نفوذ شبکه وجود دارد. برای چهار حمله شبکه خاص که انتخاب می کنیم ، یازده ویژگی از 41 ویژگی برای SecSFT برای ساخت درخت تصمیم مناسب تر است ، همانطور که در جدول 1 ذکر شده است. این ویژگی ها در طول زمان مقادیر مختلفی دارند. با استفاده از مقادیر ویژگی آماری مبتنی بر زمان ، می توان رفتار غیر عادی حملات شبکه را به صورت پویا گرفت. در این مقاله ، از یک پنجره کشویی 1 ثانیه ای برای جمع آوری مقادیر ویژگی جریان استفاده می شود. مقادیر ویژگی جریان شبکه در گره های VNF مربوط به درخت تصمیم گیری جمع آوری می شود. با جمع آوری مقادیر مشخصه ترافیک شبکه که از یک گره VNF در پنجره زمان عبور می کند و آنها را با قوانین تصمیم گره VNF تطبیق می دهد ، می توان تصمیمی گرفت که تعیین کند آیا جریان به هاپ بعدی ارسال می شود یا برای پردازش متوقف می شود به صورت محلی برای اینکه جریان به صورت محلی پردازش شود ، قوانین تشخیص / پیشگیری از نفوذ مربوط به توابع امنیتی مجازی VNF فعلی اعمال می شود. اگرچه ما می توانیم قوانین تصمیم گیری و قوانین تشخیص نفوذ را برای تعیین هدایت یا کاهش جریان ترکیب کنیم ، اما زمان مطابقت با بسیاری از قوانین ممکن است تأخیر بیشتری برای ترافیک ایجاد کند. ما از libpcap منبع باز (یک کتابخانه تابع ضبط بسته شبکه) برای استنشاق

ترافیک شبکه ارائه شده به گره های خاص VNF ، آماری از مقادیر ویژگی نسبی و انتقال ترافیک به شبکه های مجازی استفاده می کنیم. توابع شبکه مجازی سازی شده در این مقاله از Docker به عنوان محفظه استفاده می کنند.

جدول1- ویژگی هایی که برای شناسایی حملات برای secsft انتخاب شده اند.

	Attribute	Attribute description	Attack
1.	pkt_all_count	Total number of packets arriving at the VNF in the time window	Flooding DDoS
2.	pkt_all_bytes	Total bytes of packets arriving at the VNF in the time window	
3.	src_port_diff_rate	Relative rates of packets from different source ports arriving at the VNF in the time window	
4.	src_host_diff_rate	Relative rates of packets from different source hosts arriving at the VNF in the time window	
5.	dst_port_diff_rate	Relative rates of packets to different destination ports arriving at the VNF in the time window	Flooding DDoS, Probing
6.	dst_host_diff_rate	Relative rates of packets from different source hosts arriving at the VNF in the time window	Probing
7.	tcp_rate	Rate of TCP packets arriving at the VNF in the time window	SYN Flooding
8.	syn_rate	Rate of SYN bits in TCP packets arriving at the VNF in time window	
9.	udp_rate	Rate of UDP packets arriving at the VNF in the time window	UDP Flooding
10.	icmp_rate	Rate of ICMP packets arriving at the VNF in the time window	Probing, ICMP Flooding
11.	avg_pkt_rate	Average rate of incoming packets arriving at the VNF in the time window	Flooding

شکل3- توپولوژی درخت عملکرد سرویس



یک مثال از SecSFT در شکل 3 نشان داده شده است. خطوط مختلف رنگ نشان دهنده SFC های مختلف هستند و هر زنجیره ای مجموعه ای از سرویس های امنیتی خاص را ارائه می دهد. VNF 01 ریشه درخت است. وقتی ترافیک شبکه به ورودی درخت می رسد ، هنگام عبور از درخت به جریان های کوچکتر و کوچکتر تقسیم می شود.

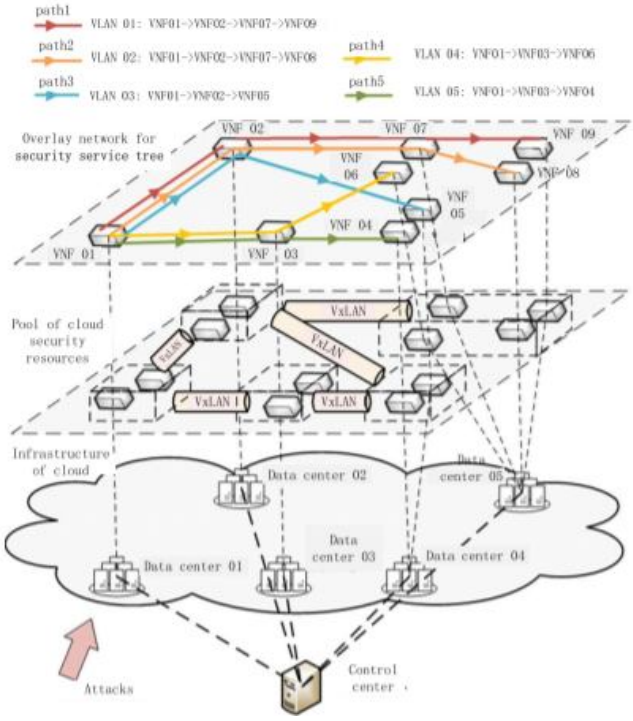
- **ارتباط بین VNF ها**

برای ارتباط بین VNF ها در محیط مجازی ، ما از OVS به عنوان سوئیچ چند لایه مجازی استفاده می کنیم. برای اطمینان از در دسترس بودن و مقیاس پذیری بالا ، یک طرح معماری دو پل OVS طراحی شده است. ما دو پل مجازی OVS را روی یک میزبان می سازیم. پل مجازی br-int به طور عمده افزودن و سلب شناسه های VLAN از شبکه های مجازی و عملکرد عادی انتقال بسته های داده را تکمیل می کند. با این حال ، در پل مجازی br-tun ، جدول جریان چند سطحی OpenFlow [20] برای گروه بندی و پردازش بسته های داده از منابع مختلف استفاده شده است. عنوان هر بسته داده با قسمت تطبیق جدول جریان پروتکل OpenFlow مطابقت دارد. در صورت موفقیت آمیز بودن ، عملیات مربوطه مانند حمل و نقل ، دور انداختن و اصلاح ، اجرا می شود. OVS یک جفت پورت پچ بین br-int و br-tun ایجاد می کند تا تحویل بسته ها را درک کند. برای پاسخگویی به نیاز ارتباط بین VNF های توزیع شده در مراکز داده چندگانه ، از شبکه overlay برای نگهداری مراکز داده متعدد در یک دامنه پخش استفاده می کنیم. بسته های پخش شده از VNF ها می توانند به تمام مراکز داده دسترسی پیدا کنند و یک شبکه لایه 2 را تحقق بخشند. این امکان را برای همه سرورها ، کانتینرها ، ماشین های مجازی و غیره فراهم می کند تا در محدوده شبکه لایه 2 ارتباط برقرار کنند. VxLAN (شبکه قابل گسترش مجازی) [21] برای انتزاع شبکه فیزیکی زیرین ، ایجاد تونل مجازی و یک شبکه مجازی لایه بزرگ 2 و تحقق انتقال پیام لایه 2 در شبکه های لایه 3 استفاده می شود.

- **تجربه SecSFT**

ما از شش سرور و چندین دستگاه سوئیچینگ استفاده کردیم. یکی از سرورها به عنوان مرکز کنترل SecSFT استفاده می شود. ما کانتینر Docker را برای پیاده سازی VNF و مدیریت منابع امنیتی مجازی انتخاب کردیم. طرح سوئیچ مجازی چند لایه OpenvSwitch برای ساخت شبکه مجازی کانتینرها به کار گرفته شده است. سیستم عامل SDN منبع باز ONOS در مرکز کنترل مستقر شد. ابر آزمایش و SecSFT در شکل 4 نشان داده شده است.

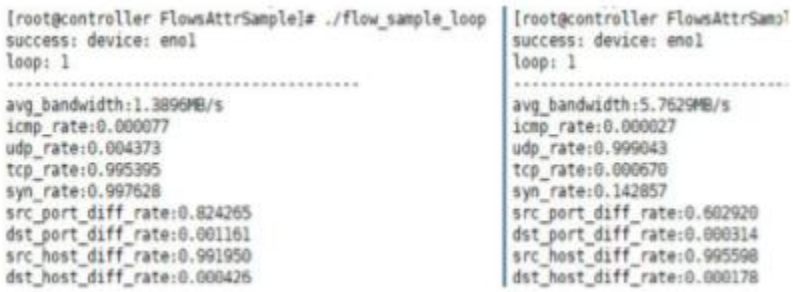
شکل4- آزمایش ابر و SecSFT.



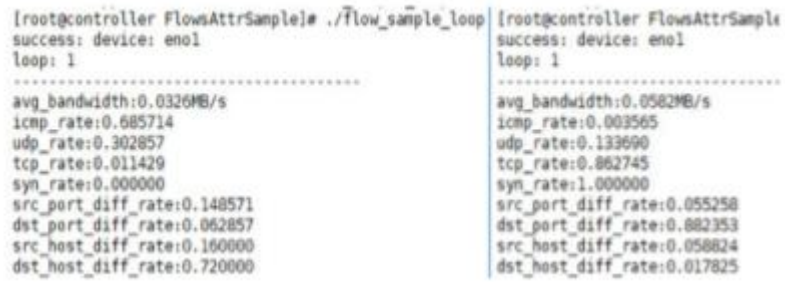
نمونه آزمایشی از تونل های VxLAN برای انتزاع شبکه فیزیکی زیرین و ساخت یک شبکه مجازی بزرگ لایه 2 استفاده می کند. جداسازی در بین انواع مختلف ترافیک شبکه با تقسیم VLAN حاصل می شود. کانتینرها از طریق شبکه مجازی با یکدیگر ارتباط برقرار می کنند.

ساخت و آموزش SecSFT به مجموعه ای از ترافیک حملات شبکه مربوطه نیاز دارد. ما مقادیر مختلفی از ویژگی های جریان های حمله به شبکه را جمع آوری کرده و در پایگاه داده ذخیره کردیم. در این آزمایش ، ما 500 نمونه آموزش را برای هر نوع حمله انتخاب کردیم. برای تأیید SecSFT از چهار نوع حمله استفاده شد. SYN Flood و UDP Flood از انواع حمله DDoS و IPsweep و Portscan از بین انواع حمله اسکن انتخاب می شوند. ما حملات SYN Flood، UDP Flood، IPsweep و Portscan را از طریق سوکت خام لینوکس ایجاد کردیم. ترافیک عادی با جمع آوری ترافیک شبکه در شبکه واقعی و راه اندازی مجدد آن به فضای آزمایش بدست می آید. مقادیر مشخصه ترافیک شبکه در گره های VNF SecSFT استخراج و تحلیل می شود. مقادیر مشخصه ترافیک شبکه در مقابل چهار حمله شبکه در کنترل کننده جمع آوری شده است ، همانطور که در شکل 5 و شکل 6 نشان داده شده است.

شکل 5- مقادیر ویژگی ترافیک در مقابل SYN Flood و UDP Flood.



شکل 6- مقادیر ویژگی ترافیک در مقابل IPsweep و Portscan



برای بررسی اینکه آیا SecSFT می تواند ترافیک حمله شبکه را به درستی شناسایی و پردازش کند ، ما به ترتیب ترافیک شبکه عادی ، ترافیک حمله SYN Flood ، ترافیک حمله UDP Flood ، ترافیک حمله IPsweep و ترافیک حمله Portscan را به گره اصلی SecSFT ارسال می کنیم. در ماژول نظارت جهانی منابع مجازی در مرکز کنترل ، ما نرخ بسته را که به هر گره VNF در SecSFT ارسال می شود بررسی می کنیم. این نرخ ها برای نشان دادن مسیرهای مختلف ترافیک شبکه و بررسی اینکه آیا ترافیک شبکه بر اساس قوانین تصمیم گیری به شاخه های ریزتر ارسال می شود ، استفاده می شود. به عنوان نمونه ای از نتایج ، جریان حمله SYN Flood در شکل 7 و 8 نشان داده شده است. ما می توانیم دریابیم که یک زنجیره از VNF ها در درخت عملکرد سرویس VNF06 <- VNF03 <- VNF01 است. هشدار تشخیص نفوذ برای حمله SYN Flood توسط VNF06 صادر می شود ، همانطور که در شکل 8 نشان داده شده است. به همین ترتیب ، همانطور که در شکل 9-14 نشان داده شده است ، حملات مختلف منجر به زنجیره های مختلف VNF از SecSFT می شود و آلارم برای حملات صادر می شود ، به ترتیب ، توسط VNF های امنیتی مربوطه. در مقابل ، ترافیک عادی بدون زنگ هشدار از طریق درخت عبور می کند ، همانطور که در شکل 15 نشان داده شده است. ترافیک مخلوط با ترافیک عادی شبکه ، ترافیک حمله SYN Flood ، ترافیک حمله UDP Flood ، ترافیک حمله IPsweep و ترافیک حمله Portscan به طور مشابه ترسیم شده است. پس از عبور از SecSFT به پنج نوع جریان مختلف شبکه. در حین پردازش

SecSFT ، مقادیر مشخصه جریان های شبکه ای که از گره های VNF عبور می کنند ، جمع آوری شده و با قوانین تصمیم سازگار می شوند و موارد نامشخص برای بررسی دقیق تر به دایره بعدی ارسال می شوند. هر زنجیره VNF در SecSFT نوعی تشخیص و جلوگیری از نفوذ را انجام می دهد. سرانجام ، ما برای ارزیابی نتایج تشخیص SecSFT از دقت ، زنگ هشدار کاذب و زنگ خطر استفاده می کنیم. دقت نشان دهنده تناسب در نمونه های جریان های مخلوط است که به درستی حمله و ترافیک عادی را مشخص می کند ، زنگ هشدار کاذب بازتاب دهنده نسبت موجود در نمونه های ترافیک عادی است که به دروغ به عنوان حمله شناخته می شود و زنگ هشدار بازتاب دهنده نسبت در نمونه هایی از حمله که به دروغ به عنوان ترافیک عادی شناخته شده است.

شکل7- سیل SYN منجر به ایجاد زنجیره ای از VNF ها شد.

```
[root@controller SetUpEnv]# python ./VNF_IO_Monitor03.py
            VNF01       VNF02       VNF03       VNF04       VNF05
----------  ----------  ----------  ----------  ----------  ----------
rx MB/s     1.7117      0.0000      1.7128      0.0000      0.0000

            VNF06       VNF07       VNF08       VNF09
----------  ----------  ----------  ----------  ----------
rx MB/s     1.7105      0.0000      0.0000      0.0000
```

شکل8- زنگ خطر برای سیل SYN صادر شده در VNF06.

```
04/18-19:27:34.404129 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 124.209.175.82:25515 -> 192.168.5.10:23333
04/18-19:27:34.692761 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 121.104.242.187:2428 -> 192.168.5.10:23333
04/18-19:27:34.738554 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 123.255.111.92:33949 -> 192.168.5.10:23333
04/18-19:27:34.867645 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 57.222.7.76:62432 -> 192.168.5.10:23333
04/18-19:27:35.054186 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 72.13.172.216:31715 -> 192.168.5.10:23333
04/18-19:27:35.190925 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 126.126.48.41:36816 -> 192.168.5.10:23333
04/18-19:27:35.261180 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 73.133.93.28:12405 -> 192.168.5.10:23333
04/18-19:27:35.662744 [**] [1:29292:0] "SYN FLOOD Attack!!!" [**] [Priority: 0]
{TCP} 89.225.37.164:50350 -> 192.168.5.10:23333
```

شکل9- سیل UDP منجر به ایجاد زنجیره ای از VNF ها شد.

```
[root@controller SetUpEnv]# python ./VNF_IO_Monitor03.py
            VNF01       VNF02       VNF03       VNF04       VNF05
----------  ----------  ----------  ----------  ----------  ----------
rx MB/s     8.6086      0.0000      8.5907      8.5958      0.0000

            VNF06       VNF07       VNF08       VNF09
----------  ----------  ----------  ----------  ----------
rx MB/s     0.0000      0.0000      0.0000      0.0000
```

شکل 10- آلارم برای سیل UDP در VNF04 صادر شده است.

```
04/18-18:58:55.559510 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 236.139.235.8:9290 -> 192.168.5.10:23333
04/18-18:58:55.559515 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 20.90.165.60:57184 -> 192.168.5.10:23333
04/18-18:58:55.559562 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 146.159.66.92:53309 -> 192.168.5.10:23333
04/18-18:58:55.559784 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 113.44.133.8:58979 -> 192.168.5.10:23333
04/18-18:58:55.559833 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 174.96.161.49:52487 -> 192.168.5.10:23333
04/18-18:58:55.559833 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 112.217.189.20:28702 -> 192.168.5.10:23333
04/18-18:58:55.559937 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 75.37.11.7:31669 -> 192.168.5.10:23333
04/18-18:58:55.559977 [**] [1:24323:0] "UDP FLOOD Attack!!!" [**] [Priority: 0]
{UDP} 154.144.51.75:1676 -> 192.168.5.10:23333
```

شکل 11- IPsweep منجر به ایجاد یک زنجیره از VNF ها شد.

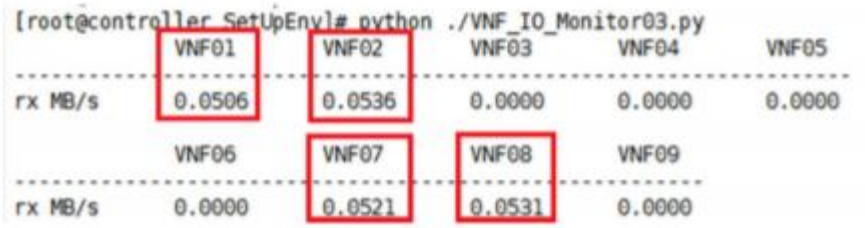
```
[root@controller SetUpEnv]# python ./VNF_IO_Monitor03.py
            VNF01      VNF02      VNF03      VNF04      VNF05
-------------------------------------------------------------
rx MB/s     0.0773     0.0782     0.0000     0.0000     0.0663

            VNF06      VNF07      VNF08      VNF09
-------------------------------------------------------
rx MB/s     0.0000     0.0000     0.0000     0.0000
```

شکل12- آلارم برای IPsweep صادر شده در VNF05

```
04/18-18:57:08.111107 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.55
04/18-18:57:08.119990 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.56
04/18-18:57:08.129107 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.58
04/18-18:57:08.130520 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.59
04/18-18:57:08.137419 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.61
04/18-18:57:08.145577 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.62
04/18-18:57:08.163271 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.57
04/18-18:57:08.171879 [**] [1:234321:0] "IPsweep Attack!!!" [**] [Priority: 0] {
ICMP} 172.18.218.234 -> 192.168.1.60
```

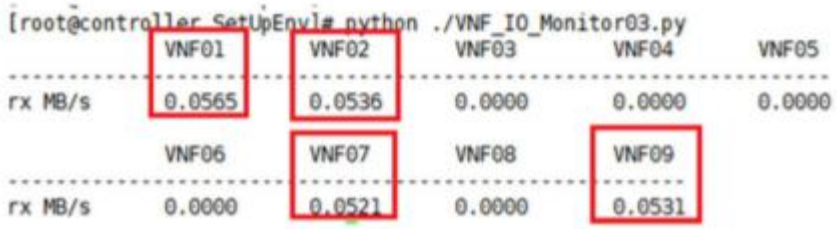
شکل 13- Portscan منجر به ایجاد زنجیره ای از VNF ها شد.



شکل 14- زنگ هشدار Portscan در VNF08 صادر شده است.

```
04/18-18:53:58.122201 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:243
04/18-18:53:58.124321 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:244
04/18-18:53:58.126415 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:245
04/18-18:53:58.128519 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:246
04/18-18:53:58.130477 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:247
04/18-18:53:58.132604 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:248
04/18-18:53:58.134545 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:249
04/18-18:53:58.136799 [**] [1:77622:0] "Portscan Attack!!!" [**] [Priority: 0] {
} 172.18.218.234:20 -> 192.168.5.10:250
```

شکل 15- ترافیک عادی منجر به زنجیره ای از VNF ها شده است.



ماتریس سردرگمی برای ارزیابی نتایج تشخیص SecSFT در برابر حملات در جدول 2 نشان داده شده است ، و صحت ، زنگ هشدار کاذب و زنگ خطر در جدول 3 نشان داده شده است. نتایج ارزیابی نشان می دهد که SecSFT طراحی شده در این مقاله دارای شناسایی بالایی است دقت برای چهار نوع حمله شبکه و سیستم قابلیت تشخیص نفوذ خوبی دارد.

جدول 2 - ماتریس سردرگمی نتایج تشخیص برای هر نوع حمله

		output attack types				
		Normal	SYN Flood	UDP Flood	IPsweep	Portscan
actual attack type	Normal	500	0	0	0	0
	SYN Flood	9	491	0	0	0
	UDP Flood	14	0	486	0	0
	IPsweep	0	0	0	493	7
	Portscan	49	0	0	0	441

جدول 3- دقت ، میزان زنگ خطر کاذب و میزان زنگ خطر از دست رفته درخت

Network Traffic Type	Accuracy	False Alarm	Miss Alarm
Normal	1.000	0.000	0.000
SYN Flood	0.982	0.151	0.018
UDP Flood	0.972	0.000	0.028
IPsweep	0.986	0.000	0.014
Portscan	0.882	0.030	0.118

نتایج ارزیابی نشان می دهد که SecSFT طراحی شده در این مقاله از دقت شناسایی بالایی برای چهار نوع حمله شبکه برخوردار است و سیستم قابلیت تشخیص نفوذ خوبی دارد.

- **نتیجه گیری**

در این مقاله ، ما معماری درخت تصمیم توزیع شده را بر اساس SDN / NFV و SFC برای خدمات امنیتی در یک ابر طراحی کردیم. ما SecSFT را در یک ابر آزمایشی مستقر کردیم و با موفقیت آن را در طبقه بندی ، شناسایی و فیلتر کردن چهار نوع حمله شبکه اجرا کردیم. سرانجام ، ما آن را با سه شاخص عملکرد ارزیابی کردیم. برای کارهای آینده ، ما SecSFT را با انواع بیشتری از حملات ارزیابی خواهیم کرد و آن را با طرح های تشخیص حمله موجود مقایسه خواهیم کرد.

• منابع

[1] A. Prajapati, A. Sakadasariya, and J. Patel, ‘‘Software defined network: Future of networking,’’ in Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC), Coimbatore, India, Jan. 2018, pp. 1351–1354.

[2] (Jan. 2020). ETSI Network Functions Virtualization (NFV). [Online]. Available: https://www.etsi.org/technologies/689-network-functionsvirtualisation

[3] (Jan. 2020). IETF Service Function Chaining. [Online]. Available: https://tools.ietf.org/wg/sfc/

[4] T. Xing, D. Huang, L. Xu, C.-J. Chung, and P. Khatkar, ‘‘SnortFlow: A OpenFlow-based intrusion prevention system in cloud environment,’’ in Proc. 2nd GENI Res. Educ. Exp. Workshop, Mar. 2013, pp. 89–92.

[5] T. V. Phan and M. Park, ‘‘Efficient distributed denial-of-service attack defense in SDN-based cloud,’’ IEEE Access, vol. 7, pp. 18701–18714, 2019.

[6] V.-C. Nguyen, A.-V. Vu, K. Sun, and Y. Kim, ‘‘An experimental study of security for service function chaining,’’ in Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN), Jul. 2017, pp. 797–799.

[7] G. Li, H. Zhou, B. Feng, G. Li, and S. Yu, ‘‘Automatic selection of security service function chaining using reinforcement learning,’’ in Proc. IEEE Globecom Workshops (GC Wkshps), Dec. 2018, pp. 1–6.

[8] Z. Ye, ‘‘Efficient, scalable and reliable network (function) virtualization in software-defined optical networks,’’ Ph.D. dissertation, California State Univ., Los Angeles, Los Angeles, CA, USA, Jun. 2015. [Online]. Available: https://ubir.buffalo.edu/xmlui/handle/10477/51473

[9] T. Lin, Z. Zhou, M. Tornatore, and B. Mukherjee, ‘‘Demand-aware network function placement,’’ J. Lightw. Technol., vol. 34, no. 11, pp. 2590–2600, Jun. 1, 2016.

[10] Y. Liu, Z. Guo, G. Shou, and Y. Hu, ‘‘To achieve a security service chain by integration of NFV and SDN,’’ in Proc. 6th Int. Conf. Instrum. Meas., Comput., Commun. Control (IMCCC), Harbin, China, Jul. 2016, pp. 21–23.

[11] D. Dwiardhika and T. Tachibana, ‘‘Optimal construction of service function chains based on security level for improving network security,’’ IEEE Access, vol. 7, pp. 145807–145815, 2019.

[12] A. Shameli-Sendi, Y. Jarraya, M. Pourzandi, and M. Cheriet, ‘‘Efficient provisioning of security service function chaining using network security defense patterns,’’ IEEE Trans. Services Comput., vol. 12, no. 4, pp. 534–549, Jul. 2019.

[13] Y. Liu, Y. Lu, W. Qiao, and X. Chen, ‘‘A dynamic composition mechanism of security service chaining oriented to SDN/NFV-enabled networks,’’ IEEE Access, vol. 6, pp. 53918–53929, 2018.

[14] J. Pei, P. Hong, K. Xue, and D. Li, ‘‘Efficiently embedding service function chains with dynamic virtual network function placement in geodistributed cloud system,’’ IEEE Trans. Parallel Distrib. Syst., vol. 30, no. 10, pp. 2179–2192, Oct. 2019.

[15] G. Li, H. Zhou, B. Feng, and G. Li, ‘‘Context-aware service function chaining and its cost-effective orchestration in multi-domain networks,’’ IEEE Access, vol. 6, pp. 34976–34991, 2018.

[16] J. R. Quinlan, C4.5: Programs for Machine Learning. San Mateo, CA, USA: Morgan Kaufmann, 2014.

[17] P. Kapoor and R. Rani, ‘‘Efficient decision tree algorithm using J48 and reduced error pruning,’’ Int. J. Eng. Res. Gen. Sci., vol. 3, no. 3, pp. 1613–1621, 2015.

[18] Y. Li, D. Zhang, J. Taheri, and K. Li, ‘‘SDN components and OpenFlow,’’ in Big Data and Software Defined Networks, J. Taheri, Ed. Edison, NJ, USA: IET, Mar. 2018, ch. 3, pp. 49–67.

[19] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, ‘‘Selecting features for intrusion detection: A feature relevance analysis on KDD 99 benchmark,’’ in Proc. 3rd Annu. Conf. Privacy, Secur. Trust, Oct. 2005, pp. 12–14.

[20] L. Dong, L. Chen, B. He, and W. Wang, ‘‘The research on designs of multiple flow tables in the Openflow protocol,’’ in Proc. 27th Int. Conf. Comput. Commun. Netw. (ICCCN), Hangzhou, China, Jul. 2018, pp. 1–2.

[21] Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks Over Layer 3 Networks, document RFC 7348, 2014. [Online]. Available: https://tools. ietf.org/html/rfc7348