بررسی چالش‌های امنیتی و چگونگی مقابله با آنها در شبکه‌های نرم‌افزارمحور

محمود دی‌پیر[1]* و مژگان قصابی[2]

[1] استادیار، دانشکده رایانه و فناوری اطلاعات، دانشگاه هوایی شهید ستاری، تهران، ایران
mdeypir@ssau.ac.ir
[2] دانشجوی کارشناسی ارشد، دانشگاه آزاد اسلامی واحد علوم و تحقیقات ، گروه کامپیوتر، تهران، ایران
mozhgan.ghasabi@srbiau.ac.ir

چکیده

در سال‌های اخیر، شبکه‌های نرم‌افزارمحور به‌منظور انعطاف و برنامه‌پذیری بیشتر در شبکه‌های رایانه‌ای مطرح شده و به‌سرعت در شبکه‌های زیرساختی و مراکز داده به کار گرفته شده‌اند. استفاده از این‌گونه شبکه‌ها مزایایی چون مقیاس‌پذیری، کاهش ترافیک کنترلی، استفاده بهینه از پهنای باند، مهندسی ترافیک بهتر و غیره را دارد؛ که همگی این مزایا ریشه در قابلیت برنامه‌پذیری این شبکه‌ها دارند. در کنار این مزایا، چالش های امنیتی نیز وجود دارند که اغلب از همین قابلیت ها سرچشمه می گیرند. وجود این چالش‌ها، اطمینان‌پذیری شبکه‌های نرم‌افزارمحور را در مقابل شبکه‌های سنتی کاهش می‌دهند؛ بنابراین شبکه‌های نرم‌افزارمحور درصورتی‌که بر اساس یک معماری امن طراحی نشوند در مقابل حملات سایبری شناخته‌شده‌ای چون حمله منع خدمت توزیع‌شده، جعل و سرقت داده ها بسیار آسیب‌پذیر خواهند بود؛ اما به‌منظور مقابله با این چالش‌ها، راه‌کارهایی ارایه شده است. در این مقاله چالش‌های امنیتی شبکه‌های نرم‌افزارمحور را به‌صورت ساختاری بررسی کرده و راه‌کارهای ارایه‌شده برای آنها را توصیف خواهیم کرد. علاوه‌براین، کاربردهای امنیتی این نوع شبکه‌ها مانند ایجاد و جداسازی ترافیک شبکه‌های مجازی، کنترل دسترسی بر روی جریان های داده و مسیریابی امن را بررسی کرده و درنهایت چگونگی شبیه‌سازی حملات به‌منظور انجام آزمایش‌های امنیتی را به‌کمک شبیه‌ساز این‌گونه شبکه‌ها، بررسی می‌کنیم.

واژگان کلیدی: شبکه‌های نرم‌افزارمحور، امنیت، حملات سایبری، حملات منع خدمت.

۱- مقدّمه

استفاده از شبکه‌های نرم‌افزار[1]محور سبب انعطاف‌پذیری و برنامه‌پذیر شدن[2] شبکه می‌شود. با استفاده از این‌گونه شبکه‌ها می‌توان همهٔ جوانب شبکه را از تنظیمات سطح پایین گرفته تا ایجاد تسهیلات مدیریت شبکه، مدیریت ترافیک و خطایابی را به‌صورت نرم‌افزاری و بهینه انجام داد. این نوع شبکه‌ها روش‌های جدیدی برای حل مسایل قدیمی شبکه مانند مسیریابی[۱] و کنترل ازدحام را ارایه می‌دهند. همچنین اجازه استفاده از روش‌هایی چون کنترل دسترسی و مقابله با حملات سایبری را فراهم می‌سازند. در این مقاله، ابتدا تهدیدهایی را که می‌توانند از آسیب‌پذیری های شبکه‌های نرم‌افزارمحور استفاده کنند، برشمرده، سپس به چگونگی مقابله با این حملات به‌وسیله ارایه راه‌کارهای مختلف می پردازیم. همچنین در مورد چگونگی طراحی یک شبکهٔ نرم‌افزارمحور امن که در مقابل حملات و آسیب‌پذیری‌ها مقاوم باشد، بحث می کنیم. درنهایت برخی از کاربردهای امنیتی شبکه‌های نرم‌افزارمحور را برمی‌شماریم و چگونگی شبیه‌سازی حملات و بررسی تأثیرات حملات امنیتی را توصیف خواهیم کرد.

۲- شبکه‌های نرم‌افزارمحور

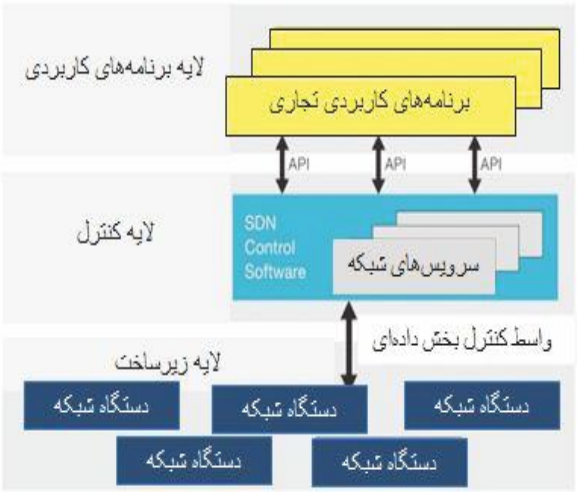
معماری بسیاری از شبکه‌های سنتی، سلسله‌مراتبی است که با استفاده از گره‌هایی از سوییچ‌های اترنت در یک ساختار

[1] Software defined networks
[2] Programmability

* نویسندهٔ عهده‌دار مکاتبات

درختی شـکل می‌گیرد. این معماری بر پایه ارتباطات مدل کارخواه/کارگزار[1] شـکل گرفته اسـت؛ اما چنین معماری ایسـتایی، برای ارتباطات پویا و نیازهای شـرکت‌ها در زمینه مراکز داده و رسـانه های کارگزار، کافی نیسـت[۲]. مواجهه با نیازهای کنونی بازار با اسـتفاده از معماری‌های متداول شـبکه تقریبا غیرممکن اسـت. شـرکت‌های فناوری اطلاعات، برای رویارویی با مسـایلی نظیر رکود یا کاهش بودجه از ابزارهای مدیریتی در سـطح ماشـین و پردازش‌های دسـتی بهره می‌گیرند. شـرکت های کارگزار مخابراتی نیز با چالش‌های مشـابهی روبه‌رو هسـتند؛ چرا که تقاضـا برای دسـترسـی به پهنای باند شـبکه‌های پویا رو به افزایش چشم‌گیری است؛ در عین حال، با افزایش هزینه‌های مربوط به تجهیزات مرکزی و کاهش درآمد، سـود این شـرکت‌ها به خطر می‌افتد. معماری شـبکه های موجود، به‌گونه‌ای طراحی نشـده‌اند که نیازهای کنونی شـرکت‌ها، کارگزارهای مخابراتی و کاربران را برطرف کنند؛ به عبارت دیگر، طراحان شبکه با محدودیت‌هایی مانند: پیچیدگی، سـیاسـت های متناقض، فقدان مقیاس‌پذیری و وابسـتگی به سخت‌افزار، وجودنداشتن هماهنگی بین نیازهای بـازار و قـابلیت های شـبکه مواجه هسـتند که این عوامل صـنعت IT را به سـوی نقطه انحرافی می‌کشـاند. برای جلوگیری از چنین رخدادی، صنعت شبکه مجبور به بازنگری این سـاختارهای سـنتی شـده اسـت. بر همین اسـاس و با پیشـگامی بنیاد شـبکه‌های باز[2]، سـعی شـده اسـت که با مطرح‌کردن مفهوم شبکه‌های نرم‌افزارمحور، معماری جدیدی برای شـبکه‌ها ارایه شـود که بتواند حد قابل قبولی از این نیازها را در نسـل جدید سـامانه‌های مبتنی بر شـبکه پوشش دهد [۳].

اگر بخواهیم به‌طور دقیق‌تر دلیل اصـلی احسـاس نیاز به SDNها را بررسـی کنیم، بایستی گفت که انگیزهٔ اصلی در این تغییرات، تحقق ویژگی مجازی‌سـازی شبکه بوده است. بر اسـاس این ویژگی، تعـدادی شـبکـه مجـازی متمایز با سـازوکارهای نشـانی‌دهی و هدایت بسـتهٔ متفاوت، می‌توانند یک بسـتر ارتباط فیزیکی یکسـان را به اشـتراک گذاشته و از آن اسـتفاده کنند. یکی از کارآمدترین روش‌های نیل به این هدف، اسـتفاده از قابلیت‌های نرم‌افزار اسـت؛ به‌گونه‌ای که نرم‌افزارهای مختلف با منطق‌های خاص خود، هر یک هدایت بسـته‌های یک جریان متناظر را در کنار سـایر جریان‌های ارتباطی در بسـتر فیزیکی بر عهده می‌گیرند؛ درواقـع این بسـتر بین جریان‌های مختلف، به اشـتراک گذاشـته می‌شود. این ویژگی سـبب افزایش بهره‌وری از منابع شـبکه برای مدیریت ارتباطات متعدد به‌ویژه در مراکز دادهٔ کنونی می‌شود [۴]. بـا درنظرگرفتن نیـازمنـدی‌های دیگر همچون قابلیت برنـامه‌ریزی شـبکه مطابق خواسـته‌های کاربران، امکان دسترسی به داده‌های توزیع‌شده، مدیریت جامع و یکپارچه و افزایش میزان گسـترش‌پذیری شـبکه، لزوم اصـلاح سـاختار شـبکه‌های سـنتی و ارایهٔ یک معماری جدید، به‌طورکامل مشهود است.



(شکل-۱): معماری شبکه SDN [۵]

با توجه به این مقدمات، معماری SDN بر اسـاس ایده جـداسـازی منطق نرم‌افزاری بسـتر کنترلی از بسـتر سـخت‌افزاری انتقال داده‌ها شـکل گرفته است. در شبکه‌های نرم‌افزارمحور سـوییچ‌ها بسته‌های ورودی را مورد پردازش قرار نمی دهنـد؛ بلکه برای تطبیق بسـته های ورودی به جداول جریـان مراجعه می کنند و اگر تطبیقی بین رکوردهای جدول و بسته ورودی نیافتند، بسته را برای پردازش به بررسی‌کننده ارسـال خواهند کرد. درواقع بررسـی‌کننده، یک سیستم عامل در شـبکه‌های SDN اسـت که بسته‌های دریافتی را پردازش کرده و در مورد بسـته‌ها طبق قوانین موجود تصـمیم‌گیری می‌کند؛ به‌عبارت دیگر در این شـبکه‌ها ارسال و پردازش‌ها از یکدیگر جدا شـده‌اند. همان‌گونه که در شـکل (۱) نیز قابل مشـاهده است؛ کنترل‌کنندهٔ اصلی سامانه به‌صورت یک واحد متمرکز از بسـتر انتقال داده، جدا شـده و نرم‌افزارها در لایهٔ بالایی و بسـتر ارتباطی شـبکه در لایهٔ زیرین قرار گرفته‌اند و

[1] Client/server

[2] Open Networking Foundation

لایهٔ کنترل ارتباط مابین برنامه های کاربردی بخش بالا و دستگاه‌های قسمت پایین معماری را فراهم می‌کند. درواقع لایهٔ کنترل، بررسی‌کننده شبکه است. بررسی‌کننده نمای متمرکز شبکه را حفظ می‌کند و از طریق رابط های باز به برنامه های کاربردی اجازه کنترل اصول زیربنایی شبکه را می‌دهد. در این معماری جزییات لایهٔ پایینی از دید نرم‌افزارهای لایهٔ بالاتر، پنهان شده است [۵]. این ویژگی سبب می‌شود که سازمان های بزرگ و توسعه‌دهندگان زیرساخت شبکه ضمن افزایش قابلیت گسترش‌پذیری و انعطاف‌پذیری شبکه، بتوانند با برنامه‌ریزی، ساختار شبکه را مطابق با نیازمندی‌ها اصلاح و به‌روزرسانی کنند.

راهبری توسعه SDNها و تدوین استانداردهای مربوطه بر عهده بنیاد شبکه‌های باز[1] است که یک تشکل غیرانتفاعی است. از جمله مشهورترین استانداردهای تدوین‌شده توسط این بنیاد، پروتکل Open Flow است که چگونگی برقراری ارتباط بین بستر کنترلی و بستر ارتباطی تجهیزات مورد نیاز در SDN را تبیین می‌کند. این استاندارد درحقیقت نخستین استانداردی است که خاص این‌گونه شبکه‌ها تعریف شده و ضمن ارایه کارایی مطلوب، توانسته سازندگان متفاوت را قادر به ارایه محصولات هماهنگ کند. مزایای این پروتکل عبارتند از: الف) مدیریت متمرکز و همگون ادوات با نشان‌های تجاری متفاوت ب) مخفی‌سازی جزییات بستر فیزیکی ارتباط از دید مدیریت شبکه ج) افزایش سرعت توسعه سرویس‌های شبکه بدون نیاز به پیکربندی جداگانه ادوات شبکه و یا تأمین تجهیزات جدید با قابلیت‌های به‌روزشده د) امکان برنامه‌ریزی کل مجموعه شبکه با استفاده از محیط‌های توسعه نرم‌افزار شناخته‌شده به‌منظور تأمین نیازهای خاص هر شبکه هـ) قابلیت اطمینان و امنیت بیشتر شبکه به دلیل مدیریت متمرکز و هماهنگ که احتمال بروز رخنه‌های امنیتی و خطاهای پیکربندی را کاهش می‌دهد. و) امکان کنترل شبکه در سطوح مختلف راهبری، کاربری و.. ی) افزایش کیفیت سرویس‌دهی از دید کاربران نهایی به واسطه تمرکز و یکپارچگی اطلاعات مربوط به هر ارتباط [۵].

۳- رویکرد شرکت‌های بزرگ درخصوص شبکه‌های نرم‌افزارمحور

معماری SDN در عرصه شبکه، تغییرات بنیادی ایجاد کرده که در جامعه حوزه شبکه جنب و جوش زیادی را به راه انداخته است. ظهور شبکه‌های نرم‌افزارمحور، کسب و کار و حاشیهٔ سود فروش سخت‌افزاری بسیاری از شرکت‌های بزرگ را تحت تأثیر قرار داده است. در سال‌های ۲۰۰۷ تا ۲۰۱۰ مدیران شرکت‌های بزرگ، مزایا و قابلیت‌های شبکه‌های نرم‌افزارمحور و پروتکل Open Flow را انکار می‌کردند و حذف مسیریاب‌ها و سوییچ‌ها را غیرممکن می‌دانستند. با تشکیل بنیاد ONF و عضویت شش شرکت بزرگ صنعت شبکه در این بنیاد و فعالیت شرکت‌های نوپا در عرصه SDN، کم‌کم شرکت‌های بزرگ، شبکه‌های نرم‌افزارمحور را به‌عنوان یک تهدید جدی شناختند و برای پیشگامی در این معماری نوظهور اقداماتی را انجام دادند. در ادامه رویکردهای شرکت سیسکو و شرکت ماکروسافت در زمینهٔ شبکه‌های نرم‌افزارمحور بیان می‌شود.

۳-۱- رویکرد شرکت سیسکو در زمینه SDN

شرکت سیسکو در سال ۲۰۱۱ تغییرات زیادی را در رویکرد خود اعمال و همچنین به عضویت بنیاد ONF پیوست. این شرکت شروع به راه‌اندازهای شبکه نرم افزار محور، مراکز داده مجازی، شبکه‌های مجازی و... کرد. درواقع سیسکو به‌جای ارایه محصولات بیشتر، به فکر ارایه راهکار و معماری برای کل شبکه‌های نرم‌افزارمحور بود و موجب ایجاد یکپارچگی شد. سیسکو در سال ۲۰۱۲ معماری و راهکار مختص شبکه‌های نرم‌افزارمحور را با نام ONE[2] معرفی کرد. در این معماری علاوه‌بر Open Flow از پروتکل Open Stack نیز استفاده می‌شود. شرکت سیسکو در سال ۲۰۱۳ سخت‌افزارهای مختص شبکهٔ نرم‌افزارمحور مانند سوییچ‌های مجازی، سری جدید سوییچ‌های قدرتمند[3] و بسترهای جدید مراکز داده[4] را معرفی کرد که همگی مبتنی بر معماری ONE با قابلیت‌های نرم‌افزاری و برنامه‌پذیری بودند. شکل (۲) نمونه‌ای از سوییچ‌های جدید Catalyst 6800 را نشان می‌دهد.

(شکل-۲): سوییچ Catalyst 6800 سیسکو

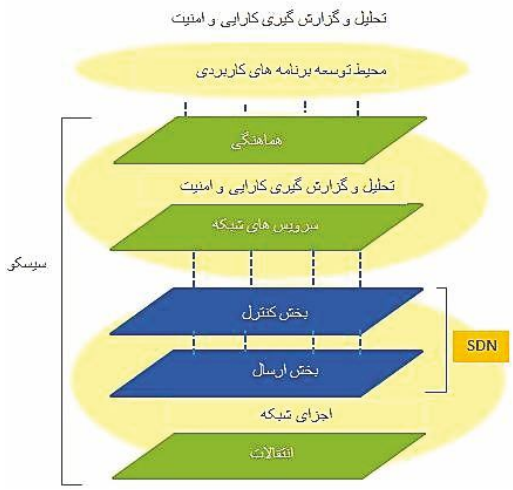
[1] Open Networking Foundation

[2] Open Network Environment

[3] Catalyst 6800

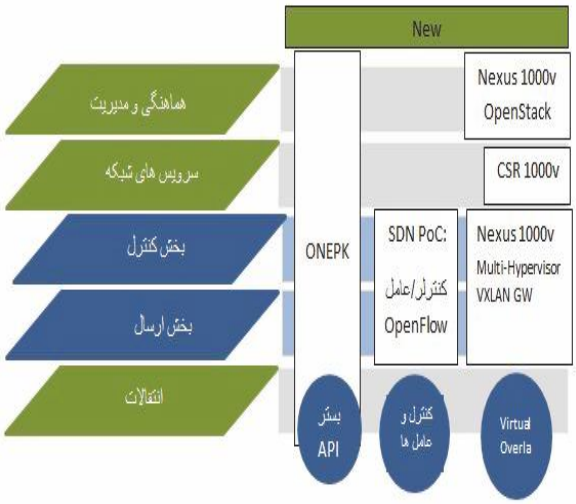
[4] Nexus7700,Insiem

معماری ONE یک چارچوب[1] برای هوشمندسازی شبکه ها است. این چارچوب انواع رویکردهای شبکهٔ نرم‌افزارمحور مانند واسط‌های برنامه‌نویسی و بررسی‌کننده‌ها را در اختیار کاربر قرار می‌دهد تا با استفاده از آن ها امکانات و قابلیت های بیشتری از جمله قابلیت های زیرساختی، سادگی عملیات و داشتن چشم‌انداز وسیع تر نسبت به شبکه را ایجاد کند و همچنین به مدیران شبکه اجازه می دهد استانداردهای صنعتی، پروتکل ها و ملزومات شبکه خود را بر اساس نیاز و مدل سازمان خود انتخاب و پیاده‌سازی کنند و به‌مرور زمان نیز با استفاده از بازخوردها و نظرات مشتریان و کاربران اقدام به سفارشی سازی و بازسازی شبکه کنند [۶]. معماری ONE را در شکل (۳) می‌بینید.



(شکل-۳): معماری ONE سیسکو مبتنی بر شبکه‌های نرم‌افزارمحور[۷]

ONEPK[2] جعبهٔ ابزاری ارایه‌شده توسط شرکت سیسکو برای به‌کارگیری چارچوب ONE و SDN است که می‌تواند حرکت به سوی شبکه‌های نرم‌افزارمحور را سرعت بخشد. با استفاده از این جعبه‌ابزار می توان نوآوری، برنامه پذیری مسیریاب‌ها و سوییچ ها، و خودکارسازی را روی بستر شبکه به‌وجود آورد. این ابزار درواقع بستر اختصاصی سیسکو برای برنامه‌نویسی شبکه های نرم‌افزارمحور است که این محیط توسعه از زبان های برنامه نویسی جاوا، پایتون و C پشتیبانی می کند. زیرساخت این بستر روی انواع سیستم عامل های سیسکو قرار گرفته که در واقع رابط مابین سیستم عامل و تجهیزات شبکه است. شکل (۴) نمایی از این ابزار را در راهکار ONE نشان می‌دهد. بسیاری از شرکت‌هایی که محصولاتی را برای شبکه های نرم‌افزارمحور توسعه دادند، فقط به ارایه یک بررسی‌کنندهٔ مرکزی یا پیاده‌سازی یک شبکه هم پوشان[3] اکتفا کردند؛ ولی شرکت سیسکو معماری کاملی براساس شبکه‌های نرم‌افزارمحور برای مشتریان خود ارایه کرده است تا با استفاده از آن و ابزارهای سخت‌افزاری و نرم‌افزاری ارایه‌شده توسط خود سیسکو از جمله سوییچ 1000v بتوانند شبکه SDN خود را بسازند[۷].



(شکل-۴): ONEPK چارچوبی برای برنامه‌نویسی شبکه‌های SDN [۷]

۳-۲- رویکرد شرکت مایکروسافت در زمینه SDN

سال‌های پیش، نقش شرکت مایکروسافت در مراکز داده و شبکه های گسترده، تولید ویندوز بود که سود بسیار ناچیزی نسبت به شرکت های تولیدکنندهٔ سخت‌افزار داشت. با ظهور شبکه های نرم‌افزارمحور فرصت مناسبی برای شرکت مایکروسافت ایجاد شد. مایکروسافت یکی از شش شرکت مؤسس بنیاد ONF است که نتایج همراهی با این بنیاد در ویندوز ۲۰۱۲ و بستر مجازی سازی اختصاصی مایکروسافت و ابزارهای مدیریت ماشین مجازی(VMM) نمود پیدا کرد. طبق اظهارات مایکروسافت، راهکارها و ابزارهایی که این شرکت فراهم آورده، یک بستر باز و عمومی قابل توسعه و منعطف برای ساخت هر نوع شبکه مبتنی بر شبکه های

[1] Framework

[2] ONE Platform Kit

[3] Overlay

نرم‌افزارمحور را فراهم می کند که با استفاده از این ابزارها می‌توان یک شبکهٔ نرم‌افزارمحور کامل را ایجاد کرد.

Hyper-V مهم‌ترین ابزار مایکروسافت برای شبکه‌های نرم‌افزارمحور است. این بستر مجازی‌سازی، اجازه می‌دهد انواع شبکه‌های مجازی، ماشین‌های مجازی را روی یک شبکه فیزیکی تعریف، پیکربندی و سیاست های شبکه‌های نرم‌افزارمحور را روی آن‌ها اعمال کرد. این ابزار با ساختن شبکه های مجازی چندمستأجره[1] روی یک شبکهٔ فیزیکی مشترک، امکان انعطاف پذیری بیشتر شبکه را فراهم می کند. Hyper-V روی هر میزبان می‌تواند سیاست های شبکه های نرم‌افزارمحور را به صورت پویا به هر شبکه مستأجر اختصاص دهد تا ترافیک شبکه براساس این سیاست‌ها به سوی مقصد هدایت شود.[۸]

VMM[2] ابزار دیگر مایکروسافت برای به کارگیری شبکه های نرم‌افزارمحور است. این ابزار کلید اصلی پیکربندی خودکار شبکه‌های نرم‌افزارمحور بر روی Hyper-V است. با استفاده از VMM می‌توان شبکه‌های مجازی چندمستأجره را بدون نیاز به پیکربندی شبکه فیزیکی، تعریف کرد. کاربران با استفاده از VMM می توانند سوییچ های مجازی مختلفی مطابق نیازها و مشخصات شبکه مجازی روی هر یک از میزبان‌های Hyper-V تعریف کنند و برای هر سوییچ مجازی چندین ماشین مجازی مرتبط با آن را بسازند. این ابزار با نظارت بارکاری[3] ماشین‌های مجازی و تغییر سیاست‌های مبتنی بر شبکه‌های نرم‌افزارمحور بر روی میزبان ها، بالاترین کارایی و مقیاس‌پذیری را در شبکه‌های مجازی و مراکز داده به‌وجود می‌آورد [۹].

ویندوز سرور ۲۰۱۲ جزء دیگری از راهکار SDN مایکروسافت است که این شرکت با ارایه ویندوز سرور ۲۰۱۲ بستر لازم برای ایجاد شبکه های نرم‌افزارمحور را کامل کرد. این سیستم عامل ارتباط بین بخش های Hyper-V وVMM را با سیستم مرکزی برقرار می کند و توسعه SDN را توسط سازمان‌ها و شرکت‌ها امکان‌پذیر می‌سازد. در ویندوز ۲۰۱۲ می‌توان سوییچ‌های مجازی قابل توسعه با بالاترین هماهنگی و یکپارچگی با شبکه های فیزیکی را ایجاد کرد. سازمان ها در استفاده از این سیستم عامل محدودیتی ندارند و می‌توانند افزونه‌های برنامه‌نویسی مورد نیاز خود را روی ویندوز سرور ۲۰۱۲ نصب و پیکربندی کنند [۱۰].

۴- چالش های امنیتی در شبکه های نرم‌افزارمحور

شبکه‌های نرم‌افزارمحور برای حل مسایل سنتی شبکه‌ها نظیر کنترل ازدحام، مهندسی ترافیک و مسیریابی و توازن بار راهکارهای نرم‌افزاری و خلاقانه‌ای ارایه می دهند [۱] و در عین حال امکان اعمال سیاست‌های شبکه‌ای پیچیده‌ای مانند کنترل دسترسی، تشخیص و مقابله با نفوذ و قابلیت اطمینان را فراهم می کنند. به‌عنوان مثال در شبکه ایتین[4][۱۱] معماری شبکه های نرم‌افزارمحور به مدیران اجازه می دهد تا سیاست های کنترل دسترسی ریزدانه‌ای را در سطح شبکه به کار گیرند. به‌عنوان مثال در [۱۲] یک هسته‌ی اعمال امنیت در کنترلرهای شبکه های نرم‌افزارمحور به‌منظور اولویت‌بندی مسایل امنیتی ارایه شده است. با توسعهٔ این ایده در [۱۳] چارچوبی به‌منظور توسعه و نصب برنامه‌های امنیتی در شبکه‌های نرم‌افزارمحور معرفی شده است. از آن‌جایی‌که بسیاری از سوییچ های شبکه امروزه از پروتکل Open Flow [۱۴] پشتیبانی می‌کنند؛ این فناوری در بسیاری از شبکه‌های صنعتی به کار گرفته شده است. بنابراین، امنیت این نوع شبکه‌ها به یک نگرانی جدی در صنعت تبدیل شده است [۱۵]،[۱۶]. علت اصلی این نگرانی به مزایای شبکه های نرم‌افزاری مربوط می‌شود. به عبارت دیگر قابلیت برنامه‌ریزی شبکه و تمرکز منطقی کنترل هم مزیت هستند و هم می توانند چالش امنیتی ایجاد کنند. درواقع این قابلیت ها راه را برای تهدیدهای جدید که درقبل وجود نداشت و یا بهره‌برداری از آنها مشکل بود، باز می کنند. شبکه‌های سنتی یک محافظت طبیعی و ذاتی در مقابل آسیب پذیری‌های معمول سیستم های IT دارند. این محافظت طبیعی ریشه در ماهیت بسته ها و دستگاه‌های انحصاری، طراحی ایستا، ناهمگونی نرم‌افزارها و توزیع‌شدگی کنترل دارد که سبب دفاع در مقابل تهدیدها معمول می‌شود. برای مثال یک حمله یا تهدید که از آسیب‌پذیری یک دستگاه خاص مربوط به یک تولیدکننده استفاده می کند، به‌احتمال فقط به بخش خاصی از شبکه که دارای آن دستگاه است، می‌تواند ضرر برساند. تنوع دستگاه‌ها و ادوات شبکه به‌نسبت شبکه‌های معمولی در شبکه‌های نرم‌افزارمحور کمتر است، زیرا دستگاه‌های مورد

[1] Multi-tenant Virtual Network

[2] Virtual Machine Manager

[3] workload

[4] Ethane

استفاده در شبکه‌های نرم‌افزارمحور هر چند از تولیدکنندگان مختلف باشند، از استاندارد و ساختار یکسانی بر مبنای پروتکل Open Flow استفاده می‌کنند. به عبارت دیگر، میزان ریسک در پیاده‌سازی های پروتکل و نرم‌افزار کنترلی سازگار، افزایش می‌یابد. به‌عنوان مثال حمله‌ای مانند استاکس نت[1][۱۷] به عنوان یک کرم هدفمند که تنها ساختار شبکه‌ای خاص را هدف قرار می دهد، می تواند در شبکه های نرم‌افزارمحور صدها دستگاه را با تغییر برنامه کنترلی از کار بیندازد و نتایج بسیار بدی را در این نوع شبکه قابل تنظیم و قابل برنامه‌ریزی به بار آورد. بسیار محتمل است که چنین حملات هدفمندی که تهدیدهای پایدار پیشرفته [۱۸] نامیده می‌شوند برای مقابله با شبکه‌های نرم‌افزارمحور توسعه داده شوند. بنابراین، اگرچه شبکه‌های نرم‌افزارمحور تحولی شگرف در معماری شبکه ها به وجود آورده است، اما سطح تهدید[2] را به شکل خطرناکی افزایش می‌دهند. ما باید با حفظ مزایای این نوع شبکه‌ها درصدد دفع خطرات سایبری آن برآییم. البته مقابله با این تهدیدها بایستی هم با ارایه راهکارهای امنیتی و تنظیمات مناسب و هم به‌صورت ساختاری در طراحی آنها در نظر گرفته شود که این دو مکمل هم هستند. شبکه‌های نرم‌افزارمحور دو ویژگی زیر را دارند که درصورتی‌که مدیران شبکه آمادگی لازم برای مقابله با آنها را ایجاد نکرده باشند، می‌توانند شبکه را به محلی ناامن برای نفوذ کاربران خرابکار[3] تبدیل کند: ویژگی نخست، توانایی کنترل شبکه به‌وسیلهٔ نرم‌افزار است که نرم‌افزار همیشه منبع اشکالات و آسیب‌پذیری‌هاست. ویژگی دوم، تمرکز هوش و توانایی شبکه در بررسی‌کننده است؛ یعنی هر شخصی با دسترسی و تسلط به کارگزارهایی که محل قرارگیری بررسی‌کننده هستند، ممکن است کل شبکه را به دست گیرد. در ادامه برخی از حملات که ممکن است شبکه‌های نرم‌افزارمحور را تهدید کند، معرفی می‌کنیم.

۴-۱- حملات منع خدمت در شبکه های نرم‌افزارمحور

حمله منع خدمت جزء جدی ترین تهدیدهاست چون بر روی کارایی شبکه، افزایش تأخیر و دور ریختن بسته‌های مشروع تأثیر می‌گذارند. این حملات ممکن است تمام شبکه را فلج یا عملکرد آن‌را متوقف کنند. برای شبکه‌های بر مبنای Open Flow حملات منع خدمت می‌تواند بسیار مخرب‌تر باشد؛ زیرا در این شبکه ها یک جریان مداوم بین بررسی‌کننده و سوییچ‌ها وجود دارد. ارتباط مداوم بین سوییچ و بررسی‌کننده ممکن است مهاجمان را تحریک کند تا جریان بین بررسی‌کننده و سوییچ را خارج کرده و فعالیت معمول شبکه را مختل سازند. برخی از حملات منع خدمت موجب تغییر حجم ترافیک می‌شود. تمایز اصلی در حملات منع خدمت اندازه وسیع ترافیک است. روش های شناسایی اندازه وسیع ترافیک، معمول‌ترین روش ها برای شناسایی این حملات است. برای پیشبرد حملات منع خدمت در شبکه های نرم‌افزارمحور، مهاجمان ممکن است حجم زیادی از ترافیک با مشخصه های تصادفی متغیر با استفاده از مولدهای ترافیکی ایجاد کنند تا هر جریان که از منظر سوییچ‌ها جدید است بسته‌های آن جریان برای تصمیم‌گیری، به‌سمت بررسی‌کننده فرستاده شود. چنین حملاتی می تواند دو هدف زیر را داشته باشد. هدف نخست اشباع جداول سوییچ‌ها با قوانین غیر مشروع است که این امر می‌تواند قابلیت سوییچ‌ها را برای پذیرش قوانین جدید مختل سازد. هدف دوم مهاجمان از فرستادن چنین حجم انبوه جریان این است که سیل عظیم جریان‌ها، کنترلر را مشغول سازد تا از پاسخ به جریان‌های مشروع و سوییچ ها بازمانده و از کار بیافتد. روش های رمزنگاری قوی و قابل اعتماد می‌تواند در ایمن‌سازی ارتباطات انحصاری بین بررسی‌کننده و سوییچ‌ها یاری‌رسان باشد [۱۳]. با این حال این روش نیز نمی‌تواند از حملات منع خدمت که از طرف میزبان‌هایی که ترافیک را به شبکه های Open Flow هدایت می‌کنند، جلوگیری کند. اگرچه برخی مطالعات بیان داشته‌اند که شبکه‌های Open Flow نسبت به شبکه‌های سنتی با حملات منع خدمت مشکلات بیشتری داشته‌اند، اما در [۱۹] نشان دادند که شبکه‌های نرم‌افزارمحور می‌تواند راه بهتری برای کنترل RTBH[4] فراهم کند. مولی و همکارانش در[۲۰] یک سازوکار چند دفاعی را در برابر حملات منع خدمت ارایه دادند که ارایه محتوای شبکه از طریق کنترلر SDN کنترل می‌شد. دایو و همکارانش نیز در[۲۱] یک فناوری فیلترکردن IP براساس منبع را برای مقابله با این حملات در SDN ارایه کردند.

۴-۲- حمله جعل هویت در شبکه های نرم‌افزارمحور

[1] Stuxnet
[2] Threat surface
[3] Malicious users
[4] Remote Triggered Black Hole

جعل هویت[1] به فرایندی اطلاق می شـــود که در آن اطلاعات شـبکه به‌صورت عمدی تغییر شکل داده شده تا هویت واقعی شـناسـه ترافیک و یا مهاجم پنهان شـود. برای مثال، کاربران ممکن اسـت از نشانی IPهای جعلی برای دست‌یابی به منابع شـبکه اسـتفاده کنند. جعل هویت اغلب بخشی از یک حمله بزرگ‌تر اسـت و نشـانی‌های جعلی ممکن است بخشی از یک شـــبکـه بات[2] یا شـــبکه زامبی به‌منظور اجرای حملات منع خدمت توزیع شـده باشـد[۲۲]. در حال حاضـر تهدید جعل هویت در شبکه های نرم‌افزارمحور بیش‌تر شامل جعل پروتکل تجزیه و تحلیل نشانی[3](ARP) و جعل IP است.

۴-۲-۱- جعل پروتکل تجزیه و تحلیل نشانی

جعل ARP شـامل اتصـال نشـانی MAC یک مهاجم به یک نشـانی IP مشـروع اسـت. حمله جعل ARP می‌تواند موجب ربوده‌شدن ترافیک از دریافت‌کنندهٔ هدف شود که به‌دنبال آن یـک کـاربر یـا میزبـان قـانونی در شـــبکه از کار می‌افتد. در شـــبکـه‌هـای Open Flow امکـان مســـمومیت ARP بین بررسی‌کننده و سوییچ‌ها وجود خواهد داشت[۲۳].

۴-۲-۲- جعل IP

جعـل IP به‌طورمعمول به‌عنوان دریچه‌ای برای دیگر حملات از جملـه دســـت‌کـاری[4] یـا تقویـت DNS[5] بـه‌کـارگرفتـه می‌شـود. DNS در واقع یک فهرسـت راهنما[6] است که نشانی IP را بـه نام دامنه مربوط می سـازد. به‌منظور تغییر مسـیر ترافیک به سـمت پایگاه های وب غیرمشـروع، یک مهاجم می‌تواند فهرسـت راهنما DNS را دست‌کاری کند. چیزی که به‌طورمعمول در تمامی حملات جعل حضـور دارد این اسـت کـه همـه آنهـا سـعی در تغییر مسـیر ترافیک به سـمت میزبان های غیرمشروع دارند. آنها همچنین می توانند به‌عنوان تلاشی برای دست‌یابی به حملات فرد در میانه تلقی شوند.

۴-۳- حملـه دسـت‌کـاری در شـبکـه هـای نرم‌افزارمحور

دسـت‌کاری شـامل تغییرات غیرمجاز و یا تخریب اطلاعات شـبکـه همـانـنـد توپولوژی جریـان هـا در جـدول جریـان، سـیاسـت ها و فهرسـت‌های دسـترسی است. برای مثال یک مهاجم ممکن اسـت سـعی در تزریق قوانین جریان را داشته باشـد که منجر به ناهنجاری در رفتار شـبکه خواهد شد. آنها همچنین می‌تواننـد جـدول جریان یا قوانین حفاظ[7] را تزریق کرده تـا موجـب ردکردن میزبـان هـای قانونی یا موجب ورود میزبان های غیرقانونی شـوند. همچنین مهاجمان ممکن است سـعی در دسـت‌کاری اطلاعات توپولوژی را داشـته باشند که متعاقباً منجر به ربوده‌شـدن بخشـی از ترافیک شـوند. در SDN جلوگیری از حملـات دســـت‌کـاری بـه‌دلیـل مـاهیت ساختاری بسیار حایز اهمیت است[۲۴]. SDN برای جلوگیری از دسـت‌کاری غیرعمدی می‌تواند بسـیار کارامد باشد. زیرا در این شـبکه، بسـته‌ها می‌توانند پیش از رفتن به مسـیر مقصـد به‌منظور تعیین صـحت ویژگی مشـخصـه ها مورد بررسی قرار گیرند. به‌عبارتی در این شـبکه‌ها دسـت‌کاری با اسـتفاده از بازرســی توزیعی و نظارت در نقاط مختلف شـبکه کاهش اثر داده می شـود. برای مقابله با دست‌کاری بررسی‌کننده بایستی به‌طور مرتب روش های رمزنگاری را بررسی کرده و اتصالات را از نظر مشروعیت مورد ارزیابی قرار داد.

۴-۴- ارتـقـای امـتـیـاز[8] در شـــبکـه هـای نرم‌افزارمحور

هنگام ورود به سـامانه، مهاجمان سـعی می‌کنند تا امتیازات دسترسی خود را افزایش داده تا به منابع و کاربردهایی که به مجوز ویژه مورد نیاز اسـت، دسـترسـی پیدا کنند. شـناسایی حملات ارتقای امتیاز نیاز به یک فرایند بازرسـی هوشـمند و توانمنـد دارد. از آنجـاکـه امتیـازهـا در مـاژول هـای کنترل دسـترسـی یا ماژول های کنترل مجوز گماشـته می‌شـوند، حملات تشـدید اغلب این ماژول‌ها را هدف قرار داده و سـعی می‌کننـد تـا اطلاعات آنها را دسـت‌کاری کنند. علاوه براین، بسـیاری از حملات فعلی، تهاجم خود را توسـط حمله به یک کاربرد مشـروع و مصـالحه آن شـروع می‌کند و سامانه‌هایی با سـطوح مجوزهای محدود و سـاده که تنها شـامل دو یا سـه سـطح مجوز باشـند نسبت به این حمله بیشتر آسیب پذیرتر خواهند بود.

۵- دسـته‌بندی تهدیدها در شـبکه‌های نرم‌افزارمحور

در این بخش، هفت بردار تهدید[9] ممکن را که در شـبکه‌های نرم‌افزارمحور شـناسـایی شده‌اند با توجه به شکل (۵) توصیف

[1] Spoofing
[2] Botnet
[3] Address Resolution Protocol
[4] Tampering
[5] Domain Name System

[6] directory
[7] firewall
[8] Elevation of privilege
[9] threat vector

خواهیم کرد [۲۵]. در این شکل بخش‌های آسیب‌پذیر یک شبکه نرم‌افزارمحور با توجه به بردارهای تهدید مربوطه شماره‌گذاری شده‌اند. البته هدف از بیان این مشکلات، این نیست که بگوییم شبکه های نرم‌افزارمحور ذاتاً ناامن تر از شبکه‌های فعلی هستند. نکته مهم این است که شبکه‌های نرم‌افزارمحور تهدیدهایی با ماهیت متفاوت دارند که نیازمند ارایه راهکارهای متفاوتی نیز هستند. درصورتی که یک شبکه نرم‌افزارمحور به‌خوبی طراحی و نصب شود، نه‌تنها از نظر توانایی های عملیاتی بلکه از نظر دوام در مقابل تهدیدهای سایبری، یک پیشرفت قابل توجه محسوب می‌شود.

بردار تهدید اول در رابطه با جریان ترافیک جعلی و ساختگی است که می‌تواند به‌منظور حمله به سوییچ‌ها یا بررسی‌کننده‌ها به‌کار رود. این تهدید می‌تواند غیر عمدی و به‌وسیلهٔ یک دستگاه خراب و یا به‌صورت عمدی توسط یک کاربر خراب‌کار انجام شود. مهاجم می‌تواند با استفاده از عناصر مختلف شبکه مانند سوییچ‌ها، کارگزارها یا رایانه‌های شخصی یک حمله منع خدمت[1] را علیه سوییچ‌های دیگر و یا بررسی-کننده انجام دهد. با یک سازوکار احراز هویت ساده می‌توان از این مشکل جلوگیری کرد، اما درصورتی‌که مهاجم کنترل کارگزاری که اطلاعات کاربران را در اختیار دارد به دست آورد، می‌تواند با استفاده از نشانی سخت‌افزاری مبدأ اتصالات و پورت‌های احراز هویت‌شده، جریان مجاز اما جعلی را به شبکه تزریق کند. استفاده از یک سامانه تشخیص نفوذ[2] با قابلیت تشخیص ترافیک غیر نرمال، یکی از راه حل‌های مقابله با این تهدید است. این راه حل می‌تواند با کنترل و پایش رفتار سوییچ‌ها تکمیل شود. به‌عنوان مثال می‌توان حد بالایی برای نرخ درخواست‌های کنترلی سوییچ‌ها در نظر گرفت.

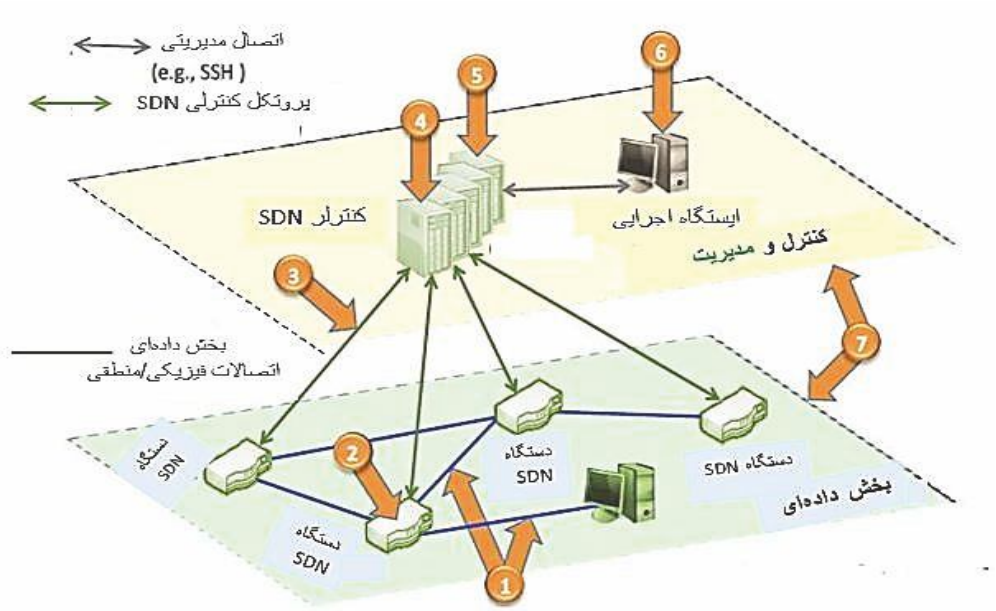
بردار تهدیدی دوم، حملات با استفاده از آسیب‌پذیری های سوییچ است. این حملات در یک سوییچ می‌تواند با کندکردن حرکت بسته‌ها در شبکه موجب حذف آنها یا رونوشت‌کردن یا منحرف‌کردن ترافیک بسته‌ها به‌منظور سرقت داده ها یا حتی تزریق ترافیک و درخواست های جعلی به‌منظور overload کردن بررسی‌کنندهٔ شبکه و سوییچ‌های مجاور به کار رود. راه حل نرم‌افزاری مانند مدیریت اعتماد برای اجزای نرم‌افزاری[۲۶] و سازوکارهایی به‌منظور نظارت و تشخیص رفتارهای غیر نرمال تجهیزات شبکه می تواند برای مقابله با این نوع تهدیدات مفید باشند.

بردار تهدید سوم، حملات به ارتباطات لایه کنترل شبکه‌های نرم‌افزارمحور است که می تواند به‌منظور ایجاد حمله منع خدمت یا سرقت داده انجام شود. در شبکه‌های نرم‌افزارمحور از ارتباط امن بر مبنای TLS/SSL به‌منظور ارتباط کنترلی امن بین سوییچ‌ها و بررسی‌کننده استفاده می‌شود؛ اما همان‌طور که در دنیای امنیت شناخته شده است، این نوع اتصال تضمین‌کنندهٔ ارتباط امن بین بررسی‌کننده و دستگاه‌های شبکه نیست و سبب ممانعت از سرقت اطلاعات نمی شود. پژوهش‌های متعددی ضعف ارتباطات بر مبنای پروتکل های TLS/SSL و همچنین نقطهٔ قوت آن یعنی زیرساخت کلید همگانی[3] (PKI) را گزارش کرده‌اند [۲۷].

توان امنیتی این ارتباطات به اندازهٔ ضعیف ترین جزء آن است که این جزء می تواند یک مجوز امضاشده، یک مرجع صدور گواهی[4] CA مورد حمله قرار گرفته و یا یک برنامه یا کتابخانه برنامه‌نویسی آسیب پذیر باشد. برای مثال پیاده‌سازی‌های زیادی از SSL آسیب‌پذیر به حمله فرد در میانه[5] در سامانه‌های حساس در سطح دنیا در حال استفاده است [۲۸]. علاوه‌بر این مدل TLS/SSL برای برقراری و اطمینان از اعتماد بین بررسی‌کننده و سوییچ، کافی نیست.

زمانی که یک مهاجم، دسترسی به لایه کنترلی را داشته باشد، توان لازم را به‌منظور اجرای حملات منع خدمت توزیع‌شده به‌واسطهٔ سوییچ‌های تحت نظر بررسی‌کننده خواهد داشت. این خلأ اعتماد حتی می‌تواند سبب ایجاد یک سیاه‌چالهٔ مجازی به‌منظور جذب و نشت داده‌ها به‌وجود آورد. استفاده از اعتماد چندگانه به‌نحوی که هر زیردامنه یا هر بررسی‌کننده یک مرجع صدور گواهی مربوط به خود را داشته باشد، یکی از راه‌حل‌های احتمالی است. راه‌حل دیگر، امن سازی ارتباطات با رمزنگاری آستانه‌ای در سطح بررسی‌کننده‌های تکرارشده است [۲۹]؛ به‌طوری‌که یک سوییچ به‌منظور دریافت یک پیام معتبر نیازمند دریافت n بخش اشتراکی از جانب بررسی‌کننده است.

[1] Denial Of Service attack(DOS)
[2] Intrusion Detection System(IDS)
[3] Public key infrastructure
[4] Certification Authority
[5] man-in- the- middle- attacks



(شکل-۵): نقشه بردارهای حملات در شبکه‌های نرم افزار محور[۲۵]

بردار تهدید چهارم، شامل حملات به آسیب‌پذیری‌های بررسی‌کننده است که به‌احتمال جدی‌ترین نوع تهدید به شبکه‌های نرم افزار محور است. یک بررسی‌کنندهٔ مخرب می تواند تمام شبکه را تحت تأثیر قرار دهد. استفاده از سامانه تشخیص نفوذ ممکن است به‌تنهایی کافی نباشد؛ زیرا روش‌های تشخیص نفوذ بر اساس الگو عمل می‌کنند و پیداکردن الگوی وقایعی منجر به واکنشی خاص می‌شود و همچنین مشخص‌کردن یک الگو به‌عنوان مخرب کار دشواری است. راه‌حل‌های ممکن استفاده از تکرار بررسی‌کننده‌ها به‌منظور تشخیص و حذف رفتار غیرنرمال، بهره گیری از گوناگونی[1] در کنترلرها، پروتکل‌ها، زبان‌های برنامه‌نویسی و...، بازیابی دوره‌ای سامانه به‌منظور بردن آن به یک وضعیت شفاف و قابل اعتماد و همچنین امن‌سازی عناصر داخلی کنترلر مانند کلیدهای رمزنگاری هستند. علاوه بر این‌ها سیاست‌های امنیتی و قوانینی که برای برنامه‌سازی شبکه استفاده می‌شود، می‌توانند مؤثر باشند.

بردار تهدید پنجم، نیز همانند بردار تهدید سوم فقدان سازوکارها به‌منظور اعتماد بین بررسی‌کننده و برنامه‌های مدیریتی است. البته تفاوت آنها در این است که نوع تأیید تجهیزات با تأیید برنامه‌ها متفاوت است. راه حل آن وجود سازوکارهایی به‌منظور تضمین قابلیت اعتماد برنامه‌های مدیریتی در زمان حیات آن‌هاست.

بردار تهدید ششم همانند شبکه‌های سنتی، حملات به آسیب پذیری‌های پایانه‌های مدیریتی[2] است. این تهدید در شبکه‌های نرم‌افزارمحور، برای دسترسی به بررسی‌کننده انجام می‌شود. تفاوت در اینجاست که در شبکه‌های نرم‌افزارمحور، این حمله اثرات بسیار مخربی دارد؛ زیرا به جای یک یا چند پایانهٔ اجرایی، کل شبکه با احاطه بر بررسی‌کننده می‌تواند در اختیار مهاجم قرار گیرد. همچنین مهاجم می‌تواند کل شبکه را دوباره از طریق بررسی‌کننده برنامه ریزی کند. راه حل‌های ممکن در اینجا استفاده از پروتکل‌هایی است که نیاز به وارسی اعتبار دوطرفه دارند. همچنین استفاده از سازوکارهایی برای بازیابی، سبب می شود که پس از راه اندازی مجدد، سیستم به حالت درست و قابل اطمینانی بازگردانده شود.

بردار تهدید هفتم، فقدان منابع قابل اعتماد برای تشخیص مقابله و بازیابی امن است. به‌منظور شناسایی حقایقی در مورد یک رخداد امنیتی، نیازمند اطلاعات قابل اعتماد از همهٔ اجزا و دامنه‌های شبکه هستیم و علاوه‌براین، داده‌ها درصورتی مفید خواهند بود که از اعتماد به آن‌ها از نظر صحت و مجازبودن بتوان اطمینان حاصل کرد. مقابله و حل مشکل، نیازمند اطلاعات لحظه‌ای، مطمین و قابل اعتمادی است که بازیابی سریع و درست عناصر شبکه‌ای را به یک حالت درست تضمین می کند. راه حل از طریق آزمایش و نظارت سازوکارهای معمول در لایه‌های داده و

[1] Diversity

[2] Administrative stations

کنترل اســت؛ اما برای اینکه مؤثر واقع شــوند می‌بایســت داده‌های مربوطه پاک‌نشدنی[1] باشند و همچنین تضمین شود کـه غیرقابل تغییر و امن هســتند. همچنین رخداد نماهای[2] ثبـت وقایع باید به‌صــورت راه دور و در محیطی امن ذخیره شوند.

جـدول (۱) هفت بردار تهدید توصــیف‌شــدهٔ بالا را جمع‌بندی کرده و همچنین وضـعیت خاص‌بودن این حملات برای شــبکه های نرم‌افزارمحور را نشــان می دهد. طبق این جدول، تهدیدهای شـماره ۳ تا ۵ در شـبکه های سنتی ظاهر نمی شــوند و خاص شــبکه های نرم‌افزارمحور هســتند. این تهدیدها ناشــی از جداســازی لایه کنترل و لایه داده و نتیجهٔ معرفی موجودیتی جـدیـد در این شــبکـه هـا یعنی "کنترلر منطقی متمرکز" هســتند. تأثیر این حملات در شــبکه های نرم‌افزارمحور بیشــتر بوده و به شــکلی متفاوت بروز می کنند؛ بنابراین نیاز اســت که به‌نحو متفاوتی با آنها برخورد و مقابله شـود. بررسـی هفت بردار تهدید؛ نشـان می‌دهد که گسـترهٔ حملات[3] در شــبکه های نرم‌افزارمحور نسـبت به شــبکه‌های سـنتی بیشتر است و روش‌های مقابله‌ای متفاوتی را می‌طلبد. با توجه به این تهدیدهای می‌بایست امنیت و قابلیت اطمینان جزء نخستین مراحل طراحی این‌گونه شبکه‌ها باشد.

در بخش بعدی در مورد بستر بررسی امن و قابل اعتماد برای شبکه‌های نرم‌افزارمحور به‌منظور مقابله با این تهدیدات می‌پردازیم.

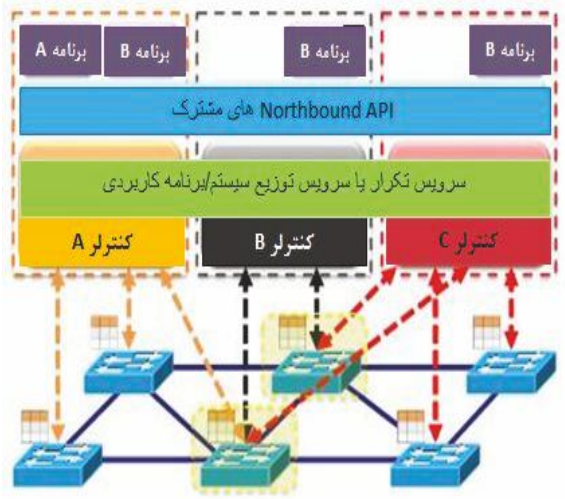
۵-۱- بســتر بررســی‌کنندهٔ امن و قابل اعتماد برای شبکه‌های نرم‌افزارمحور

در این بخش راه حـل هـای مقـابلـه‌ای لازم را برای بردارهای تهـدیـد ۱ تـا ۷ ارایه می‌کنیم. این راه حل ها در طراحی یک شـبکهٔ نرم‌افزارمحور می‌بایسـت لحاظ شـوند. شـکل (۶) یک شـکل سـاده‌شده را از چنین طراحی نشان می دهد. در ادامه روش‌هایی به‌منظور مقابله با بردارهای تهدید شـناسـایی‌شـده در شبکه‌های نرم‌افزارمحور ارایه شده است.

تکرار[4] یکی از فنون مهم بهبود قـابلیت اعتماد ســامانه‌های شبکه‌ای توزیع شده است [۳۰]. همان‌طورکه در شکل۶ دیده می‌شـود، بررسـی‌کننده با سـه نمونه A، B و C تکرار شـده اسـت. همچنین برنامه‌های سـطح شـبکه نیز می توانند تکرار شوند. در این شکل، برنامه B نیز سه بار تکرار شده است. این روش ترکیبی، تحمل‌پذیری سـخت‌افزار و نرم‌افزار را در مقابل حوادث و خراب‌کاری‌ها بیشتر می‌کند. به کمک تکرار می‌توان عیوب[5] را پوشـاند و برنامه یا بررسـی‌کنندهٔ مشـکل‌دار و یا مخرب را عایق[6] کرد. درصـورت از هم‌گسـسـته‌شـدن شبکه، برنامه B می‌تواند به‌منظور برنامه ریزی کل سـوییچ‌های شبکه به کار رود؛ ولی برنامه A که تکرار نشـده اسـت، قادر به اجرا نخواهد بود.

(جدول-۱): بردارهای تهدید شبکه‌های نرم‌افزارمحور در مقابل تهدیدهای عمومی شبکه‌ها [۲۵]

تهدیدات	مخصوص SDN	نتیجه در شبکه نرم افزار محور
بردار اول	خیر	میتواند راهی برای حملات منع خدمت باشد.
بردار دوم	خیر	تأثیر بالقوه‌ی آن افزایش یافته است.
بردار سوم	بله	ارتباطات با کنترل کننده متمرکز منطقی میتواند مورد سوء استفاده قرار گیرد.
بردار چهارم	بله	کنترل غیر مجاز کنترل کننده ممکن است کل شبکه را به خطر بیاندازد.
بردار پنجم	بله	برنامه های مخرب میتوانند توسعه یافته و در کنترلر مستقر شوند.
بردار ششم	خیر	تأثیر بالقوه‌ی آن افزایش یافته است.
بردار هفتم	خیر	اطمینان از بازیابی سریع و تشخیص زود هنگام رخداد خطا بسیار مهم است.



(شکل-۶): معماری شبکه نرم افزار محور امن [۲۵]

[1] Indelible
[2] Log
[3] Attack Surface
[4] Replication
[5] Fault
[6] Isolated

گوناگونی[1] روش دیگری به‌منظور افزایش استحکام سامانه‌های امن است [۳۱]،[۳۲]. تکرار و گوناگونی بررسی‌کننده‌ها، یک انتخاب خوب برای افزایش امنیت است. اصل پشت سر این سازوکار، اجتناب از عیوب یکسان است که می‌توانند شامل اشکالات نرم‌افزاری[2] یا انواع آسیب‌پذیری‌ها باشند. برای مثال می‌دانیم که سیستم‌عامل های موجود از خانواده‌های متفاوت، تعداد بسیار کمی آسیب‌پذیری مشترک دارند [۳۲] یعنی گوناگونی سیستم‌عامل‌ها تأثیر حملات روی آسیب پذیری‌های مشترک را کم می‌کند. در شبکه‌های نرم‌افزارمحور، یک برنامهٔ مدیریتی می‌تواند روی کنترلرهای مختلف اجرا شود که این‌کار با تعریف یک واسط برنامه‌نویسی استاندارد و مشترک و یک تجرید مناسب در بخش بالایی شبکهٔ نرم‌افزارمحور که محل اجرای برنامه‌های شبکه‌ای است، قابل انجام خواهد بود.

سازوکارهای خوددرمان‌کننده[3]، در شرایط مورد حمله قرارگرفتن طولانی؛ بازیابی پویا و واکنشی[4] می‌تواند سامانه را به حالت درست برگردانده و اجزای آسیب دیده را جایگزین کند. جایگزینی بایستی با نسخه جدید و گوناگونی اجزا صورت گیرد که این مسیله توان دفاع را در مقابل حملاتی که آسیب‌پذیری‌های مشخصی هدف قرار می‌دهند، افزایش می دهد.

تخصیص پویای تجهیزات[5]، یکی دیگر از راه‌کارهای امن‌سازی شبکه‌های نرم‌افزارمحور است. اگر یک سوییچ فقط به یک بررسی‌کننده نسبت داده شود، لایهٔ کنترلی آن تحمل‌پذیری لازم در مقابل عیوب و حملات را نخواهد داشت. زمانی که بررسی‌کنندهٔ مربوطه خراب شود، عملیات کنترلی سوییچ از کار می‌افتد. بنابراین نیاز است که این سوییچ به‌صورت پویا و امن به چند بررسی‌کننده متصل باشد. البته می‌بایست قابلیت‌هایی چون رمزنگاری آستانه‌ای به‌منظور تشخیص بررسی‌کنندهٔ خراب‌کار و همچنین احراز هویت بررسی‌کننده‌ها را برای جلوگیری از حمله فرد در میانه، در این راه‌کار لحاظ کرد. سوییچی که به بررسی‌کننده‌های مختلفی می‌تواند وصل شود، به‌صورت خودکار در مقابل عیوب و حملات تحمل‌پذیر است. از جمله مزایای جانبی این قابلیت می توان به افزایش بازدهی (توازن بار به‌وسیلهٔ چند بررسی‌کننده) و کاهش تأخیر(پاسخ‌گویی سریع چند بررسی‌کننده نسبت به بررسی‌کنندهٔ واحد) اشاره کرد [۳۳]. به‌منظور داشتن قابلیت اتصال به چند بررسی‌کننده و داشتن قابلیت‌های امنیتی مربوطه، نیاز است که قابلیت برنامه‌ریزی در سمت لایهٔ داده (سوییچ‌ها) با به‌کارگیری پردازنده مناسب، افزایش یابد.

اعتماد بین تجهیزات و بررسی‌کننده در امنیت شبکه‌های نرم‌افزارمحور بسیار مهم است. برقراری اعتماد بین دستگاه‌ها و بررسی‌کننده، یک نیازمندی مهم برای قابلیت اعتماد در لایه کنترلی است. اگرچه سوییچ‌ها باید این اجازه را داشته باشند که به‌صورت پویا به کنترلرها متصل شوند، اما باید یک ارتباط قابل اعتماد وجود داشته باشد. یک راه حل در اینجا استفاده از فهرست سفید دستگاه‌های قابل اعتماد و مجاز در سمت بررسی‌کننده است؛ اما این راه‌کار انعطاف‌پذیری مطلوب را در شبکه‌های نرم‌افزارمحور ندارد. راه حل دیگر، اعتماد به همهٔ سوییچ‌هاست تا زمانی که غیر قابل اعتمادبودن آنها محرز شود. یک رفتار غیر نرمال یا مخرب از هر طرف ارتباط می‌تواند به‌وسیلهٔ سوییچ یا بررسی‌کننده با استفاده از الگوریتم‌های تشخیص ناهنجاری یا تشخیص خرابی، شناسایی و گزارش شود. زمانی که میزان اعتماد سوییچ یا بررسی‌کننده از یک حد آستانه‌ای مشخص و قابل قبول کمتر شود، سوییچ به‌صورت خودکار، توسط همهٔ دستگاه‌ها و بررسی‌کننده‌های دیگر، قرنطینه می‌شود.

اعتماد بین برنامه‌ها و نرم‌افزار بررسی‌کننده یکی دیگر از راه‌کارهای امنیتی است. با گذشت زمان مؤلفه های نرم‌افزار، قدیمی شده و اشکالاتی در آنها پدیدار می‌شود و یا مورد حمله قرار می‌گیرند. بنابراین یک مدل اعتماد پویا مانند آنچه در [۲۶] ارایه شده نیاز است. در این نوع مدل‌ها براساس ویژگی‌های مشاهده‌شدهٔ طرف مقابل مانند دسترس‌پذیری، قابلیت اتکا[6]، یک‌پارچگی، امنیت، قابلیت نگهداری و قابلیت اطمینان، میزان اعتماد مشاهده و اندازه‌گیری می‌شود. این مدل، اعتماد بین موجودیت های سامانه را تعریف و نظارت می‌کند.

دامنه‌های امنیتی مجزا[7] روشی بسیار معمول است که در سامانه‌های مختلف استفاده می‌شود. برای مثال، در سیستم‌عامل‌ها برنامه‌های سطح کاربر اجازه دسترسی به

[1] Diversity
[2] Software bugs
[3] Self-healing
[4] Proactive and reactive recovery
[5] Dynamic device association
[6] Reliability
[7] Isolated security domains

بخش‌های سطح هسته را ندارند. کرومیوم یک مثال خوبی برای طراحی دامنه‌های امنیتی مجزا است که بسیار کارآمد عمل می‌کند [۳۴]. کرومیوم از جعبه‌های شنی[1] برای جداسازی موتور رندر از هستهٔ مرورگر استفاده می‌کند. بنابراین در بسیاری از حمله‌ها که موتور رندر را تحت تأثیر قرار می‌دهند، هستهٔ سیستم مصون خواهد ماند. دامنه‌ها یا محدوده‌های امنیتی در بستر کنترلی یک شبکهٔ نرم‌افزارمحور می‌توانند به‌وسیلهٔ روش‌هایی چون استفاده از جعبه شنی و مجازی‌سازی ایجاد شوند. در این روش‌ها اجازه انجام عملیات و ارتباطات حداقلی بین دامنه‌های امنیتی داده می‌شود.

مؤلفه‌های امن[2] یکی از بلوک‌های ساختاری لازم برای طراحی یک سامانه امن و قابل اعتماد را فراهم می‌کنند. این مؤلفه‌ها باید طوری باشند که درصورت تهاجم به سیستم، داده‌های حساس مانند کلیدهای رمزنگاری از دست نروند.

ترمیم و به‌روزرسانی پویا و امن نرم‌افزار، راه‌کار امنیتی دیگری است. از آنجایی که هیچ نرم‌افزاری بدون اشکال یا آسیب‌پذیری نیست، ترمیم و به‌روزرسانی برای کاهش اندازه پنجره آسیب‌پذیری (کاهش زمان وجود آسیب پذیری) لازم است. بنابراین بستر کنترلی باید سازوکارهایی به‌منظور اطمینان از روش‌های امن و منعطف به‌روزرسانی را فراهم آورد. راه حل‌هایی مانند آنچه در [۳۵] ارایه شده است، می‌تواند برای نیل به این هدف سودمند باشد. در جدول۲ تهدیدهایی که به‌کمک هر راه حل و سازوکار، قابل مقابله هستند، مشخص شده‌اند [۲۵].

راه حل‌های ارایه شده در جدول ۲ سازوکارهای پایه‌ای لازم را به‌منظور طراحی و پیاده‌سازی یک بستر کنترلی امن و قابل اعتماد برای شبکه‌های نرم‌افزارمحور فراهم می‌آورند. این بستر امن می‌تواند از سازوکارهای سنتی مانند محیط حفاظ، سامانه تشخیص نفوذ و سایر ابزارها نیز استفاده کند [۳۶]. ابزارهای سنتی تأمین امنیت، در شبکه‌های نرم‌افزارمحور را می‌توان به‌صورت نرم‌افزاری پیاده‌سازی کرد و در قالب برنامه‌های تحت شبکه سازماندهی کرد.

۶- توسعه نرم‌افزارهای امنیتی

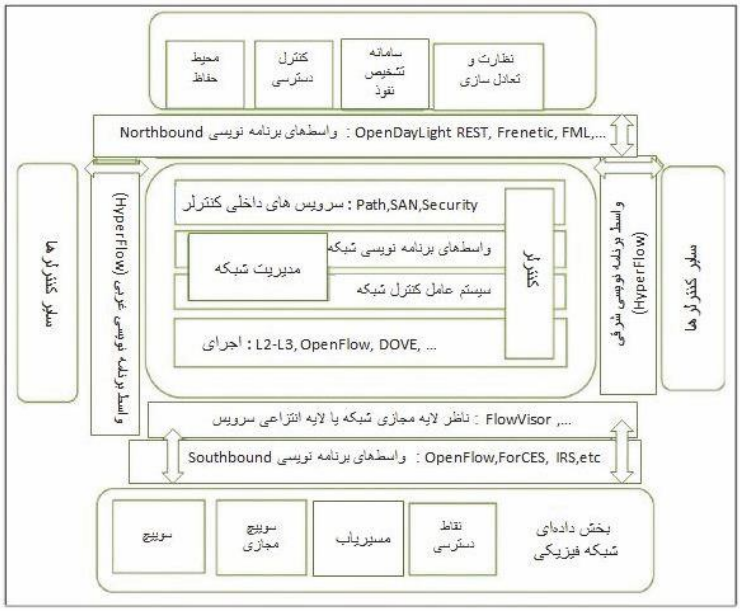
برای مقابله با چالش‌های امنیتی علاوه‌بر داشتن یک معماری امنیتی مناسب و راه‌کارهای فهرست‌شده در جدول۲، می‌بایست از قابلیت برنامه ریزی این شبکه‌ها نیز استفاده کرد. همان‌طور که در شکل (۷) دیده می‌شود، در بالاترین لایه یک شبکهٔ نرم‌افزارمحور، با استفاده از توابعی که کنترلرها فراهم می‌کنند؛ می توان نرم افزارهایی را برای انجام کارهایی چون مدیریت ترافیک، توازن بار، مقابله با خرابی پیوند، نظارت امنیتی و غیره توسعه داد. همان‌طور که درقبل نیز گفته شد، ترافیک ناشناخته‌ای که به سوییچ‌ها می‌رسد، برای تعیین تکلیف به بررسی‌کننده ارسال می‌شود. بسیاری از حمله‌ها در همین ترافیک ناشناخته پنهان هستند. بنابراین می‌توان نرم‌افزارهایی را هم برای مقابله با چالش‌های امنیتی، توسعه داده و بر روی بررسی‌کننده اجرا کرد تا مقابله با حملات شناخته‌شده و همچنین حملات جدید را بر عهده گیرند. شکل (۷) محل قرارگیری این برنامه‌های امنیتی را در کنار سایر اجزای یک شبکهٔ نرم‌افزارمحور، نشان می‌دهد. همان‌طور که در بالاترین لایهٔ این شکل دیده می‌شود، محیط حفاظ، کنترل دسترسی جریان، سامانه تشخیص نفوذ (IDS) در کنار سایر نرم‌افزارها با استفاده از توابع برنامه‌نویسی در بررسی‌کننده‌ها، به کار گرفته می‌شوند.

(جدول-۲): راه حل‌هایی برای بردارهای تهدید[۲۵]

راه حل/سازوکار	بردارهای تهدید
تکرار	۱،۴،۵،۷
گوناگونی	۳،۴،۶
خود درمانی	۲،۴،۶
تخصیص پویای سوییچ	۳،۴
اعتماد بین کنترلر و دستگاه‌ها	۱،۲،۳
اعتماد بین کنترلر و برنامه‌ها	۴،۵
دامنه‌های امنیتی	۴،۵
مؤلفه های امن	۴،۵،۷
ترمیم و بروز رسانی سریع و امن	۲،۴،۶

[1] Sandbox

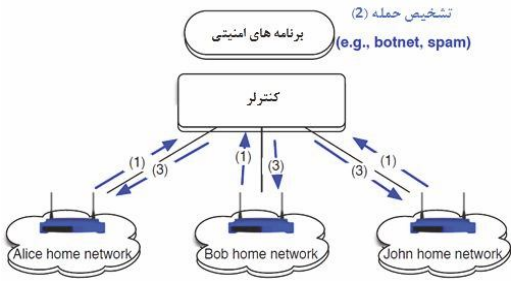
[2] Secure components



(شکل-۷): معماری یک شبکه نرم‌افزارمحور به همراه برنامه‌های امنیتی[۲۳]

۷- امنیت به‌کمک شبکه‌های نرم‌افزارمحور

اگرچه شبکه های نرم‌افزارمحور با برخی چالش های امنیتی روبه‌رو هستند، اما خود این شبکه‌ها می‌توانند برای برقراری امنیت به کار روند. یکی از کاربردهای شبکه‌های نرم‌افزارمحور استفاده از آنها به‌منظور کنترل دسترسی جریان‌های داده است [۳۷]. این کنترل دسترسی در سطح ماشین‌های مجازی انجام می‌گیرد؛ به‌نحوی که قبل از ارسال داده از یک ماشین مجازی، یک بسته بررسی به مقصد ارسال می‌شود و مقصد پس از بررسی دسترسی فرستنده، به وی پاسخ لازم را می‌دهد. بر اساس نوع پاسخ دریافتی، فرستنده بر حسب دسترسی خود می‌تواند عملیاتی را انجام دهد. جداسازی ترافیک شبکه های منطقی مختلف بر روی یک شبکه فیزیکی واحد، یکی دیگر از کاربردهای امنیتی شبکه های نرم‌افزارمحور است [۳۸] که محدودیت‌های شبکه‌های مجازی VLAN را ندارد. در این کاربرد، شبکه‌های مجازی اگرچه بر روی یک یا چند شبکه فیزیکی، قرار دارند؛ ولی از ترافیک ارسالی یکدیگر مطلع نمی‌شوند که در محرمانه‌نگه‌داشتن اطلاعات بسیار مؤثر می‌تواند باشد. حفظ امنیت دستگاه های مربوط به شبکه‌های خانگی[1] توسط شبکه‌های نرم‌افزارمحور به‌صورتی کارا قابل انجام است [۳۹]. همان‌طور که در شکل۸ مشاهده می‌شود، یک سوییچ شبکه نرم‌افزارمحور را می‌توان در یک نقطه دسترسی[2] به‌صورت توکار به کار گرفت به‌نحوی که توسط یک برنامه امنیتی، ترافیک شبکه را نظارت کرده و پس از تشخیص حمله، قوانین لازم برای مقابله با حمله را به جدول سوییچ اضافه کند.



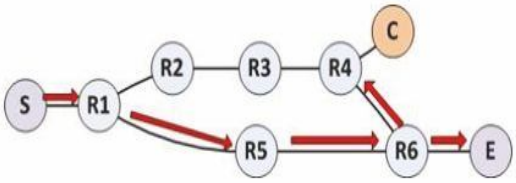
(شکل-۸):به کارگیری شبکه‌های نرم‌افزارمحور برای مقابله با حملات در شبکه‌های خانگی

حفظ امنیت شبکه‌های ابری به‌دلیل وجود حملات از داخل ابر و همچنین پیکربندی پویای این‌گونه شبکه‌ها بسیار مشکل است. به‌کمک شبکه‌های نرم‌افزارمحور الگوریتم‌های جدیدی می‌توان توسعه داد که با استفاده از یک دستگاه نظارت‌کننده در درون ابر، امنیت آن را تا حدود زیادی برقرار کرد [۴۰]. در این الگوریتم مسیریابی نه‌تنها کوتاه‌ترین فاصله بین مبدأ و مقصد به‌دست می‌آید، بلکه کوتاه‌ترین فاصلهٔ این مسیر تا دستگاه نظارت‌کننده حملات نیز محاسبه می‌شود. برای مثال در شکل ۹ کوتاه‌ترین فاصلهٔ بین گره‌های S و E به‌دست آمده و سپس با استفاده از انعطاف‌پذیری شبکه‌های نرم‌افزارمحور، ترافیک در مسیریاب R6 رونوشت و به سمت مسیریاب R4

[1] Home networks

[2] Access Point

هدایت می‌شود. این مسیریاب به یک دستگاه نظارت‌کنندهٔ امنیتی شبکه با نام C متصل است.



(شکل-۹): الگوریتم مسیریابی کوتاه‌ترین مسیرها[۴۰]

در اینجا کم‌ترین فاصل مسیر شناسایی‌شده تا مسیریاب متصل به دستگاه نظارت‌کننده، توسط الگوریتم مسیریابی به‌دست می‌آید. به این ترتیب حملات درونی شبکه‌های ابری توسط دستگاه نظارت کننده، قابل تشخیص خواهند بود.

برای توسعهٔ برنامه های امنیتی مختلف مانند محیط حفاظ نرم‌افزاری و تشخیص حملات منع خدمت و.. در شبکه‌های نرم‌افزارمحور دانش زیادی لازم است. به‌منظور ساده‌سازی برنامه‌نویسی امنیتی در شبکه‌های نرم‌افزارمحور، بستری به نام FRESCO توسعه داده شده است [۱۳]. این بستر یک محیط توسعه با امکان اشتراک پیام‌ها و منابع شبکه بین برنامه ها را به‌وجود می‌آورد. علاوه‌براین امکاناتی را به‌منظور ایجاد برنامه‌های کنترل دسترسی جریان در شبکه های نرم‌افزارمحور فراهم می کند. همچنین در این محیط توسعه توابع پایه‌ای کنترل دسترسی جریان مانند مسدودسازی، ممانعت‌کردن، اجازه‌دادن، تغییر مسیردادن و قرنطینه‌کردن جریان‌های شبکه‌ای، فراهم شده است.

۸- شبیه‌سازی حمله منع خدمت توزیع‌شده در شبکه‌های نرم‌افزارمحور

آزمایش نرم‌افزارها و راه‌کارهای جدید امنیتی در شبکه‌های نرم‌افزارمحور واقعی، زمان‌بر و پرهزینه هستند؛ بنابراین می‌توان از شبیه‌سازهای شبکهٔ نرم‌افزارمحور برای آزمایش و بررسی‌های موردنظر استفاده کرد. شبیه‌ساز مینی‌نت[1] به‌منظور آزمایش شبکه‌های نرم‌افزارمحور توسعه داده شده است [۴۱]. شبیه‌ساز مینی‌نت برنامه‌ای است که می‌تواند شبکه ای مجازی را ایجاد کرده و هسته‌های واقعی، کدهای برنامه و سوییچ‌های شبکه را بر روی یک ماشین (ماشین مجازی، ابر یا سامانهٔ واقعی) به اجرا درآورد. این نرم‌افزار به‌دلیل امکان تعامل‌پذیری توسط واسط‌های برنامه‌نویسی کاربردی[2] می‌تواند در راستای توسعه، آموزش و پژوهش استفاده شود. این شبیه‌ساز محیطی را فراهم می‌کند که به‌کمک آن می‌توان توپولوژی دلخواه یک شبکه نرم‌افزارمحور را به‌صورت گرافیکی شبیه‌سازی کرد و کل شبکه را در یک رایانهٔ واحد در اختیار داشت. علاوه‌براین، می‌توان کلیهٔ قابلیت‌ها و تنظیمات انجام‌شده بر روی شبکهٔ نرم‌افزارمحور شبیه‌سازی‌شده را بدون هیچ تغییری در یک شبکه واقعی به کار گرفت.

حملهٔ منع خدمت توزیع‌شده، یکی از مرسوم‌ترین حملات بوده که همان‌طور که درقبل گفته شد، تشخیص و مقابله با آن در شبکه‌های نرم‌افزارمحور اهمیت خاصی دارد. ما به‌منظور آزمایش امنیتی شبکه‌های نرم‌افزارمحور از شبیه‌ساز مینی‌نت استفاده خواهیم کرد و تأثیرات حمله منع خدمت توزیع‌شده را بر روی این شبکه مورد بررسی قرار می‌دهیم. به‌منظور بررسی تأثیرات حملهٔ منع خدمت توزیع‌شده در شبکه‌های نرم‌افزارمحور، توپولوژی با ۱۳سوییچ و۸۴ میزبان در مینی‌نت ایجاد کرده و از بررسی‌کنندهٔ Ryu که کامل‌ترین بررسی‌کننده برای شبکه‌های نرم‌افزارمحور است [۴۲] استفاده می کنیم. به‌کمک ابزار Scapy از یک نود خارج از شبکهٔ نرم‌افزارمحور ترافیک مصنوعی را با نشانی‌های مبدأ جعلی تولید کرده و به گره های هدف شبکهٔ نرم‌افزارمحور ارسال می‌کنیم. مشخصات بسته‌های ترافیک تولیدی را در جدول ۳ می‌بینید.

(جدول-۳): مشخصات بسته‌های ترافیک حمله [۴۳]

نوع حمله	محتوی	شماره پورت مبدأ	نام پروتکل
منع خدمت توزیع شده	فاقد محتوی	۸۰	UDP

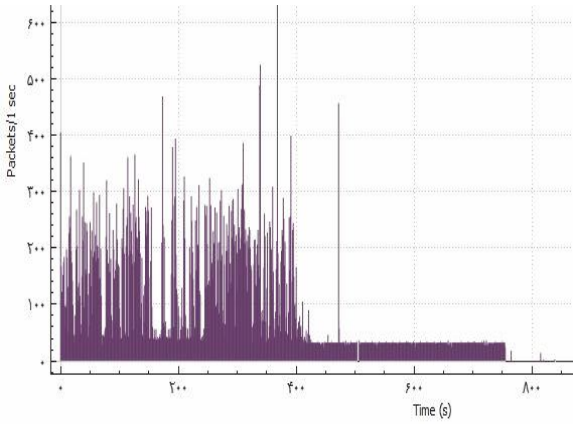
با توجه به این‌که درآمد بسته‌های ورودی وضعیت حمله با جداول موجود در سوییچ‌ها مطابقتی ندارند، به‌عنوان بسته‌های جدید شناخته شده و برای تعیین تکلیف به بررسی‌کننده ارسال می‌شوند و به این ترتیب حجم زیادی از ترافیک به بررسی‌کننده روانه می‌شود.

به‌منظور نظارت بر ترافیک شبکه بین سوییچ‌ها و بررسی‌کننده از ابزار wireshark استفاده می‌کنیم. این نرم‌افزار یک تحلیل‌گر بسته در شبکه است و بسته‌هایی را که در شبکه رد و بدل

[1] Mininet

[2] application programming interface (API)

می‌شوند، با دام انداختن آنها مورد پردازش قرار می‌دهد. یکی از قابلیت‌های خوب شبیه‌ساز مینی‌نت این است که به‌طوردقیق مانند یک شبکهٔ واقعی به نظر می‌رسد، به‌طوری که wireshark این تفاوت را متوجه نمی‌شود و با دادن نشانی‌های درونی شبکهٔ شبیه‌سازی‌شده می‌توان ترافیک هر پیوند دلخواهی از شبکه را نظارت کرد. گراف حاصل از پایش ترافیک بررسی‌کننده توسط این ابزار را در شکل ۱۰ می‌بینید.



(شکل-۱۰): گراف حاصل از شبیه‌سازی حمله منع خدمت توزیع‌شده

همان‌طور که در گراف شکل ۱۰ مشاهده می‌شود، در طول حمله منع خدمت توزیع‌شدهٔ راه‌اندازی‌شده در شبکه، بسته‌های زیادی به سمت بررسی‌کننده ارسال شده است. به‌طورتقریبی در ثانیه ۳۹۱ عملکرد بررسی‌کننده به‌دلیل کمبود منابع، مختل شده و به‌طور تقریبی در ثانیه ۷۴۸ بررسی‌کننده به‌طور کامل از کار افتاده است. در طول شبیه‌سازی میزان مصرف cpu و حافظه و پهنای باند نیز بررسی شده است که نتایج را در جدول ۴ مشاهده می کنید. طبق نتایج حاصل از این بررسی، مهاجم با تزریق مداوم بسته‌های جعلی، پردازش سنگینی را برای کنترلر ایجاد می‌کند که این امر موجب اشغال پهنای باند در کانال سوییچ- بررسی‌کننده می‌شود. با توجه به منابع محدود بررسی‌کننده قادر به پاسخ‌گویی ترافیک عادی شبکه نخواهد بود و درنتیجه از دسترس خارج می‌شود [۴۴].

(جدول-۴): نتایج بررسی میزان مصرف cpu و حافظه

وضعیت شبکه	مصرف cpu	پهنای باند	مصرف حافظه
وضعیت عادی	٪۳۱	۱,۳ G	۸۱ Mb
وضعیت حمله	٪۱۰۰	۰,۳ G	۳۰۸ Mb

۹- جمع‌بندی

ما در این مقاله ضمن معرفی شبکه‌های نرم‌افزارمحور و مزایای آنها، چالش‌های امنیتی آنها را به‌صورت ساختاری معرفی کردیم و راه‌کارهای ارائه‌شده برای این چالش‌ها را مرور کردیم. این راه‌کارها را می‌توان به‌صورت عمده به دوگروه راه‌کارهایی بر اساس معماری امنیتی و راه‌کارهای نرم‌افزاری دسته‌بندی کرد. از طرف دیگر نمونه‌هایی از کاربردهای امنیتی شبکه‌های نرم‌افزارمحور را برشمردیم. باید به این نکته دقت شود که شبکه‌های نرم‌افزارمحور اگرچه درعمل در دنیا به کار گرفته شده است، اما پروتکل‌ها و ابزارهای مربوط به این شبکه‌ها در حال تغییر و تحول هستند. بنابراین هم‌زمان باید مسایل امنیتی مربوط به تغییرات پروتکل‌های آن را نیز در کنار تکامل قابلیت‌های عملیاتی، مدنظر قرار داد.

۱۰- منابع

[1] Caesar, M., Caldwell, D., Feamster, N., Rexford, J., Shaikh, A., & van der Merwe, J. (2005, May). Design and implementation of a routing control platform. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2* (pp. 15-28). USENIX Association.

[2] Hakiri, A., Gokhale, A., Berthou, P., Schmidt, D. C., & Gayraud, T. (2014). Software-defined networking: Challenges and research opportunities for future internet. *Computer Networks*, *75*, 453-471.

[3] Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, *103*(1), 14-76.

[4] Chowdhury, N. M, and Raouf Boutaba, "Network virtualization: state of the art and research challenges" Communications Magazine, IEEE 47, (2009), no.7, 20-26.

[5] Fundation, Open Networking. "Software-defined networking: The new norm for networks" ONF White Paper (2012) , url: https://www.opennetworking.org

[6] http://blogs.cisco.com /data center / cisco - open network - environment-explained, Access Date: October 2016.

[7]http://www.theregister.co.uk/2012/06/14/cisco_one_sdn_openflow_openstack, Access Date: October 2016.

[8]https://technet.microsoft.com/enus/library/mt169373(v=ws.11).aspx (Access Date: October 2016).

[9]https://technet.microsoft.com/enus/library/gg610610(v=sc.12).aspx (Access Date: October 2016).

[10]https://www.microsoft.com/en/servercloud/products/windows-server-2012-r2/ (Access Date: October 2016).

[11] Casado, M., Freedman, M. J., Pettit, J., Luo, J., Gude, N., McKeown, N., & Shenker, S. (2009). Rethinking enterprise network control. *IEEE/ACM Transactions on Networking (ToN)*, *17*(4), 1270-1283.

[12] Porras, P., Shin, S., Yegneswaran, V., Fong, M., Tyson, M., & Gu, G. (2012, August). A security enforcement kernel for OpenFlow networks. In *Proceedings of the first workshop on Hot topics in software defined networks* (pp. 121-126). ACM.

[13] Shin, S., Porras, P. A., Yegneswaran, V., Fong, M. W., Gu, G., & Tyson, M. (2013, February). FRESCO: Modular Composable Security Services for Software-Defined Networks. In *NDSS*.

[14] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., ... & Turner, J. (2008). OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, *38*(2), 69-74.

[15] S. Sorensen,"Security implications of software-defined networks",(2012), url: http://goo.gl/BiXH2.

[16] S. M. Kerner," Is SDN Secure?"(2013), url: http://goo.gl/lPn2V.

[17] D. Kushner," The Real Story of Stuxnet", (2013). url:http://goo.gl/HIEHQ.

[18] Tankard, C. (2011). Advanced persistent threats and how to monitor and deter them. *Network security*, *2011*(8), 16-19.

[19] Yuzawa Tamihiro," OpenFlow 1.0 actual use case: RTBH of DDoS traffic while keeping the target",(2013).Online,http://packetpushers.net/open flow-1-0-actual-use-case-rtbh-ofddos-traffic-while-keeping-the-target-online.

[20] Mowla, N. I., Doh, I., & Chae, K. (2014, July). Multi-defense Mechanism against DDoS in SDN Based CDNi. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2014 Eighth International Conference on* (pp. 447-451). IEEE.

[21] Dao, N. N., Park, J., Park, M., & Cho, S. (2015, January). A feasible method to combat against DDoS attack in SDN network. In *2015 International Conference on Information Networking (ICOIN)* (pp. 309-311). IEEE.

[22] Yao, G., Bi, J., & Xiao, P. (2011, October). Source address validation solution with OpenFlow/NOX architecture. In *2011 19th IEEE International Conference on Network Protocols* (pp. 7-12). IEEE.

[23] Alsmadi, I., & Xu, D. (2015). Security of software defined networks: A survey. *computers & security*, *53*, 79-108.

[24] Othman, O. M., & Okamura, K. (2013). Securing distributed control of software defined networks. *International Journal of Computer Science and Network Security (IJCSNS)*, *13*(9), 5.

[25] Kreutz, D., Ramos, F., & Verissimo, P. (2013, August). Towards secure and dependable software-defined networks. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking* (pp. 55-60). ACM.

[26] Yan, Z., & Prehofer, C. (2011). Autonomic trust management for a component-based software system. *IEEE Transactions on Dependable and Secure Computing*, *8*(6), 810-823.

[27] Holz, R., Riedmaier, T., Kammenhuber, N., & Carle, G. (2012, September). X. 509 Forensics: Detecting and Localising the SSL/TLS Men-in-the-middle. In *European Symposium on Research in Computer Security* (pp. 217-234). Springer Berlin Heidelberg.

[28] Georgiev, M., Iyengar, S., Jana, S., Anubhai, R., Boneh, D., & Shmatikov, V. (2012, October). The most dangerous code in the world: validating SSL certificates in non-browser software. In *Proceedings of the 2012 ACM conference on Computer and communications security* (pp. 38-49). ACM.

[29] Y. G. Desmedt, "Threshold cryptography", In: European Transactions on Telecommunications 5.4 (1994).

[30] Andrew S. Tanenbaum, "Modern Operating Systems", Pearson; 4 edition, 2014.

[31] S. Neti, A. Somayaji, and M. E. Locasto, "Software diversity: Security, Entropy and Game Theory", In: 7th USENIX HotSec, 2012.

[32] M. Garcia et al, "Analysis of operating system diversity for intrusion tolerance", In: Software Practice and Experience (2013).

[33] B. Heller, R. Sherwood, and N. McKeown, "The controller placement problem", In: HotSDN. ACM, 2012.

[34] A. Barth et al," The Security Architecture of the Chromium Browser", Tech. rep. Stanford University, (2008).

[35] J. H. Perkins et al, "Automatically patching errors in deployed software", In: ACM SIGOPS SOSP, 2009.

[36] N. Foster et al, "Frenetic: a network programming language", In: SIGPLAN Not, 2011.

[37] Lucian Popa, Minlan Yu, Steven Y. Ko, Sylvia Ratnasamy , "CloudPolice: taking access control out of the network", Ion Stoica HotNET, 2010.

[38] A. Blenk, A. Basta, M. Reisslein, & W. Kellerer. Survey on Network Virtualization Hypervisors for Software Defined Networking. arXiv preprint arXiv:1506.07275, 2015.

[39] S. Sundaresan, S. Burnett, N. Feamster, & W. De Donato. BISmark: A testbed for deploying measurements and applications in broadband access networks. In 2014 USENIX Annual Technical Conference (USENIX ATC 14). (2014), pp. 383-394.

[40] Seungwon Shin,Guofei Gu, "CloudWatcher: Network Security Monitoring Using Openflow In Dynamic Cloud Networks" 20th IEEE International Conference on Network Protocols, 2012.

[41] Lantz, Bob, Brandon Heller, and Nick McKeown, "A network in a laptop: rapid prototyping for software-defined networks", In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ACM, (2010), p-19.

[42] R. Khondoker, A. Zaalouk, R. Marx, & K. Bayarou. Feature-based comparison and selection of Software Defined Networking (SDN) controllers. In Computer Applications and Information Systems (WCCAIS), 2014 World Congress on. (2014), pp. 1-7.

[43] Seyed Mohammad Mousavi and Marc St Hilaire," Early Detection of DDoS Attacks against SDN Controllers", International Conference on Computing, Networking and Communications, Communications and Information Security Symposium, (2015) pp. 10-15

[44] Zhang, J., Qin, Z., Ou, L., Jiang, P., Liu, J., & Liu, A. X. (2010, October). An advanced entropy-based DDOS detection scheme. In *2010 International Conference on Information, Networking and Automation (ICINA)* (Vol. 2, pp. V2-67). IEEE.

محمود دی‌پیر مدرک دکترای خود را در رشتهٔ کامپیوتر-سامانه‌های نرم‌افزاری و مدرک کارشناسی ارشد خود را در رشتهٔ کامپیوتر-نرم‌افزار هر دو از دانشگاه شیراز دریافت کرده است. مقطع کارشناسی خود را نیز در همین رشته از دانشگاه هوایی شهید ستاری دریافت کرده است. ایشان هم‌اکنون عضو هیئت علمی دانشکده رایانه و فناوری اطلاعات دانشگاه هوایی شهید ستاری است. زمینه‌های پژوهشی وی شامل داده‌کاوی و امنیت فضای سایبر است و دارای مقالات متعددی در مجلات و کنفرانس‌های معتبر ملی و بین المللی است. نامبرده در پروژه‌های پژوهشی و صنعتی متعددی مشارکت داشته است.

مژگان قصابی مدرک کارشناسی را در رشته فناوری اطلاعات با کسب رتبه نخست از دانشگاه زنجان اخذ کرده و اکنون دانشجوی دوره کارشناسی ارشد رشته فناوری اطلاعات گرایش شبکه‌های رایانه‌ای در دانشگاه علوم تحقیقات تهران می‌باشد که در دوره کارشناسی ارشد نیز در دو نیمسال تحصیلی، رتبه نخست را کسب کرده است. از جمله زمینه‌های پژوهشی مورد علاقهٔ وی می توان به امنیت شبکه‌های نرم‌افزارمحور، محاسبات فراگیر، محاسبات ابری و سیستم‌های خبره اشاره کرد.