بخش تحقیق پروژه

عرفان رفیعی اسکویی – 98243027

امیرحسین ثابتی – 98243015

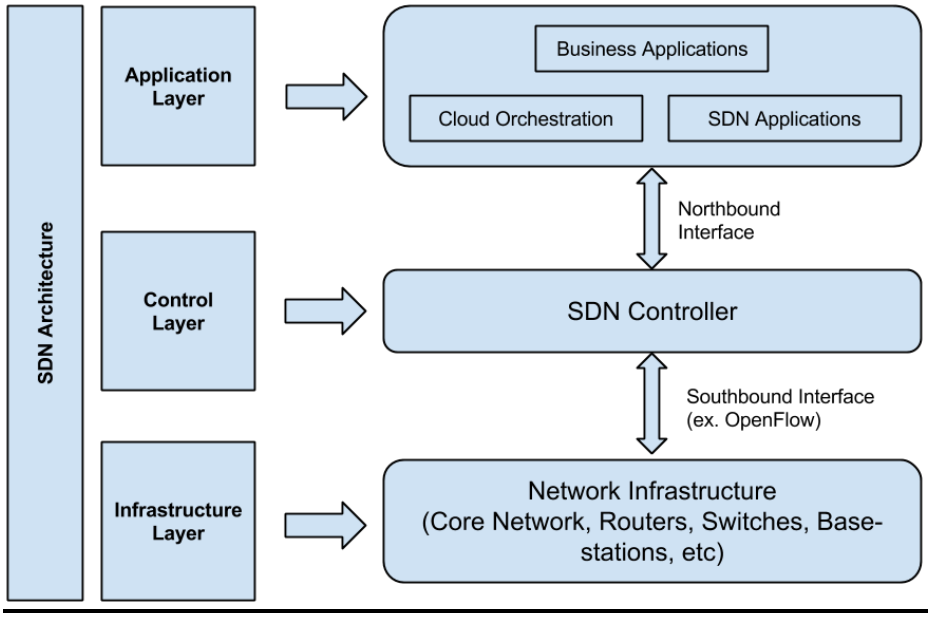
1) SDN :

Software-Defined Networking (SDN) رویکردی برای شبکه‌سازی است که از کنترل‌کننده‌های مبتنی بر نرم‌افزار یا Application programming interfaces (API) برای ارتباط با زیرساخت‌های سخت‌افزاری و ترافیک مستقیم در شبکه استفاده می‌کند.

هدف SDN بهبود کنترل شبکه با بهینه سازی شرکت ها و ارائه دهندگان خدمات برای پاسخ سریع به نیازهای در حال تغییر کسب و کارها است.

این مدل با شبکه های سنتی فرق دارد، آنها از دستگاه های سخت افزاری اختصاصی (یعنی روترها و سوئیچ ها) برای کنترل ترافیک شبکه استفاده می کنند. اما SDN می تواند یک virtual network ایجاد و کنترل کند یا اینکه یک سخت افزار سنتی را از طریق نرم افزار کنترل کند.

معماری SDN :



Application layer :

شامل برنامه های معمولی شبکه یا عملکردهایی است که سازمان ها استفاده می کنند. در حالی که یک شبکه سنتی از یک ابزار تخصصی مانند فایروال یا متعادل کننده بار استفاده می کند، SDN دستگاه را با برنامه ای جایگزین می کند که از یک کنترل کننده برای مدیریت رفتار data استفاده می کند.

Control layer :

SDN متمرکز که به عنوان مغز شبکه تعریف شده توسط نرم افزار عمل می کند را نشان میدهد. این کنترلر بر روی یک سرور قرار دارد و سیاست ها و جریان های ترافیکی را در سراسر شبکه مدیریت می کند.

Infrastructure layer :

از سوییچ های فیزیکی در شبکه تشکیل شده است. این سوئیچ ها ترافیک شبکه را به مقصد خود هدایت می کنند.

API's :

این سه لایه با استفاده از API های مربوط به northbound و southbound با هم ارتباط برقرار می کنند. برنامه ها از طریق northbound با کنترلر صحبت می کنند. کنترلر و سوئیچ ها با استفاده از southbound با هم ارتباط برقرار می کنند

چرا SDN مهم است؟

1) افزایش کنترل با سرعت و انعطاف‌پذیری بیشتر:

توسعه‌دهندگان به‌جای برنامه‌نویسی دستی چندین دستگاه سخت‌افزاری خاص، می‌توانند با برنامه‌نویسی یک کنترل‌کننده مبتنی بر نرم‌افزار استاندارد، جریان ترافیک روی شبکه را کنترل کنند. همچنین انعطاف بیشتری در انتخاب تجهیزات شبکه به وجود میاید، زیرا می توان یک پروتکل واحد را برای ارتباط با هر تعداد دستگاه سخت افزاری از طریق یک کنترل کننده مرکزی انتخاب کرد.

2) زیرساخت شبکه قابل تنظیم:

با SDN مدیران می توانند خدمات شبکه را پیکربندی کنند و virtual resources را برای تغییر زیرساخت شبکه به صورت real time از طریق یک مکان متمرکز اختصاص دهند و همچنین جریان داده‌ها را از طریق شبکه بهینه کنند و برنامه‌هایی را که نیاز به دسترسی بیشتر دارند، اولویت‌بندی کنند.

3) امنیت قوی:

SDN نرم‌افزار قابلیت دید را در کل شبکه ارائه می‌دهد و دید جامع‌تری از تهدیدات امنیتی ارائه می‌دهد. اپراتورها با SDN می توانند مناطق جداگانه ای برای دستگاه هایی ایجاد کنند که به سطوح مختلف امنیتی نیاز دارند و یا بلافاصله دستگاه های در معرض خطر را قرنطینه کنند تا نتوانند بقیه شبکه را آلوده کنند.

SDN چطوری کار میکند؟

در SDN، نرم افزار از سخت افزار جدا می شود. SDN صفحه کنترلی را که تعیین می‌کند ترافیک به نرم‌افزار ارسال شود، حرکت می‌دهد و صفحه داده‌ای را که در واقع ترافیک را در سخت‌افزار به جلو می‌برد، کنار میگذارد و به مدیران شبکه اجازه می‌دهد تا کل شبکه را بر اساس یک صفحه شیشه‌ای برنامه‌ریزی کنند و نه

. device by device

موارد استفاده SDN :

Service provider networks - Campus networks - DevOps

تاثیرات SDN :

SDN تأثیر عمده ای بر مدیریت زیرساخت های فناوری اطلاعات و طراحی شبکه داشته است. همانطور که فناوری SDN پیشرفته تر میشود، نه تنها طراحی زیرساخت شبکه را تغییر می دهد، بلکه نقش آن را نیز در حوزه IT نیز تغییر می دهد.

معماری های SDN می توانند کنترل شبکه را با استفاده از پروتکل های باز مانند OpenFlow برنامه ریزی کنند. به همین دلیل، شرکت ها می توانند کنترل aware software را در لبه های شبکه خود اعمال کنند. این امکان دسترسی به سوئیچ‌ها و مسیریاب‌های شبکه را فراهم می‌کند، به جای استفاده از سیستم عامل بسته و اختصاصی که معمولاً برای پیکربندی، مدیریت، ایمن‌سازی و بهینه‌سازی منابع شبکه استفاده می‌شود.

تقریباً همه افراد در بازار مالی به شبکه‌های قدیمی وابسته هستند که می‌توانند غیرقابل پیش‌بینی باشند، مدیریت آن دشوار باشد، تحویل آن کند و در برابر حملات آسیب‌پذیر باشند. با SDN، بخش خدمات مالی در سازمان ها می‌توانند predictive network ایجاد کنند تا پلتفرم‌های کارآمدتر و مؤثرتری را برای اپلیکیشن‌های معاملات مالی فعال کنند.

منابع :

https://www.vmware.com/topics/glossary/content/software-defined-networking.html

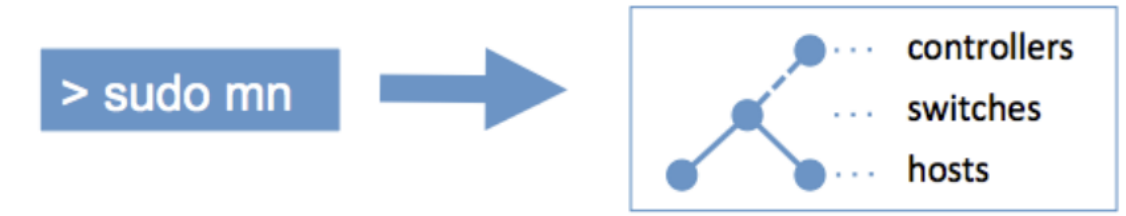
https://www.techtarget.com/searchnetworking/definition/software-defined-networking-SDN

2) Mininet :

Mininetیک شبیه ساز شبکه است که شبکه ای از virtual host ها، سوئیچ ها، کنترلرها و لینک ها را ایجاد می کند. host های Mininet نرم افزار استاندارد شبکه لینوکس را اجرا می کنند و سوئیچ های آن از OpenFlow برای مسیریابی انعطاف پذیر و شبکه های تعریف شده توسط نرم افزار پشتیبانی می کنند.

Mininet از تحقیق، توسعه، یادگیری، نمونه سازی، آزمایش، اشکال زدایی و هر کار دیگری که می تواند از داشتن یک شبکه آزمایشی کامل بر روی لپ تاپ یا رایانه شخصی دیگر بهره مند شود، پشتیبانی می کند.

Command :



Mininetیک راه آسان برای به دست آوردن رفتار صحیح سیستم و عملکرد و آزمایش توپولوژی ها ارائه می دهد.

Mininet چگونه کار میکند؟

تقریباً هر سیستم عاملی منابع محاسباتی را با استفاده از process abstraction مجازی سازی می کند. Mininet از مجازی سازی مبتنی بر فرآیند برای اجرای بسیاری از هاست ها و سوئیچ ها بر روی یک هسته سیستم عامل استفاده می کند.

Mininet می‌تواند کرنل یا سوئیچ‌های OpenFlow فضای کاربر یا کنترل‌کننده‌هایی برای کنترل سوئیچ‌ها و میزبان‌ها برای برقراری ارتباط از طریق شبکه شبیه‌سازی شده ایجاد کند. Mininet سوئیچ ها و هاست ها را با استفاده از جفت های virtual ethernet (veth) متصل می کند. Mininet در حال حاضر به کرنل لینوکس وابسته است، در آینده ممکن است از سیستم عامل های دیگر مانند Solaris containers یا FreeBSD jails! پشتیبانی کند.

کد Mininet تقریباً کاملاً پایتون است، به جز یک بخش کوچک که به زبان C است.

منابع :

http://mininet.org/overview/#:~:text=Mininet%20is%20a%20network%20emulator,routing%20and%20Software%2DDefined%20Networking.

http://mininet.org/

https://opennetworking.org/mininet/

3) Floodlight :

کنترلر Floodlight یک open source SDN است و یک برنامه کاربردی مبتنی بر جاوا است که بر روی کنترلر Beacon OpenFlow از دانشگاه استنفورد ساخته شده است. Floodlight مجموعه ای از API های REST و همچنین ماژول هایی که از منطق custom و پیاده سازی پروتکل تشکیل شده اند را برای قابلیت برنامه ریزی ارائه می دهد. همچنین شامل یک مدیر توپولوژی، ماژول حمل و نقل، مدیر دستگاه و فشار دهنده جریان استاتیک برای مدیریت جریان در سوئیچ ها است.

هدف اصلی کنترلر Floodlight ارائه یک لایه کنترل SDN جایگزین برای مدیریت عملکرد لایه 2 و لایه 3 اصلی شبکه های دارای OpenFlow است.

Layer 2:

- Packet forwarding at the Data link layer
- Flow-based switching (OpenFlow Switch).
- Generic Network Protocol Processing
- Quality of Service (QoS) based on packet features

Layer 3:

- Packet forwarding at the network layer.
- Routing using OpenFlow routing protocol methods.
- IP multicast/broadcast support for forwarding packets to multiple destinations in a single go.
- Management and control APIs for external applications such as network security, policy enforcement and management.

ویژگی های Floodlight controller :

1. Floodlight : OpenFlow Switching and Routing پیاده سازی پروتکل OpenFlowرا ارائه می دهد که روشی یکپارچه برای تعامل با سوئیچ ها و روترهای شبکه به منظور پیکربندی، مدیریت، نظارت و ایمن سازی شبکه را به راحتی فراهم می کند.

2. Floodlight : Network Virtualization به شما اجازه می دهد تا به سرعت شبکه های مجازی را در بالای infastructure های فیزیکی موجود خود ایجاد کنید که می تواند شامل سرویس ها یا برنامه های مختلف باشد و به طور جداگانه مدیریت شوند.

3. Floodlight : Network Monitoring به شما کمک می کند تا عملکرد شبکه خود را نظارت کنید و به شما امکان می دهد آمارهای کل (مانند توان عملیاتی و تأخیر) را در زمان واقعی در هر دو لایه سوئیچ و برنامه مشاهده کنید.

4. Floodlight : Application Programming Interfaces (APIs) توسعه دهندگان را قادر می سازد تا برنامه های کاربردی سفارشی بنویسند که می توانند با استفاده از مجموعه ای از API های REST به خوبی تعریف شده با کنترلر تعامل داشته باشند. این امر توسعه برنامه‌هایی مانند web interfaces یا حتی برنامه‌های تلفن همراه را که بدون نیاز به یادگیری عمیق در مورد پروتکل OpenFlow یا هر جزئیات خاص دیگری در مورد نحوه عملکرد داخلی Floodlight با کنترلر تعامل دارند را برای توسعه‌دهندگان بسیار آسان‌تر می‌کند.

موارد استفاده Floodlight controller :

Network virtualization : به کاربران اجازه می دهد چندین شبکه مجازی را در یک دستگاه فیزیکی متصل کنند.

Server load balancing : تقاضای ترافیک را در زمان واقعی در چندین سرور توزیع می کند.

Port Mirroring : ترافیک را از یک دستگاه یا پورت به دستگاه یا پورت دیگر کپی می کند.

تونل زنی : اتصالات ایمن را از طریق اینترنت بین دو یا چند کامپیوتر متصل به شبکه های مختلف برقرار می کند.

پشتیبانی از کیفیت خدمات(QoS) : پارامترهایی را تنظیم می کند که ترافیک شبکه را بر اساس نیازهای کاربر اولویت بندی می کند.

مشاهده جریان : جریان ترافیک شبکه را برای مسائلی مانند خطاها، تاخیرها و افت بسته ها نظارت می کند.

مسیریابی : کنترل جامع بسته ها را بر اساس پروتکل های لایه 4/3 فراهم می کند.

(پروتکل های لایه 4/3 به پروتکل های شبکه ای اشاره دارند که در لایه های سوم و چهارم مدل OSI (Open Systems Interconnection) کار می کنند. پروتکل اصلی لایه 3 IP (پروتکل اینترنت) است که خدمات آدرس دهی، مسیریابی بسته و ارسال بسته را ارائه می دهد. پروتکل های لایه 4 شامل Transmission Control Protocol (TCP)، User Datagram Protocol (UDP)،

Real-Time Transport Protocol (RTP) و Internet Control Message Protocol (ICMP) است. این پروتکل ها نحوه قالب بندی و انتقال داده ها از طریق شبکه را کنترل می کنند).

معماری Floodlight controller :

Floodlight controller با معماری مدرن، ماژولار و قابل توسعه با اجزای زیر طراحی شده است:

1 – Core Manager Layer : لایه مدیریتی را برایREST API ها، پیکربندی خودکار دستگاه و امنیت را فراهم می کند.

2 - Application Layer : رابط هایی با کاربری آسان را برای کنترلرها فراهم می کند تا با برنامه ها تعامل داشته باشند و قابلیت های آنها را برای کاربران نهایی گسترش دهند.

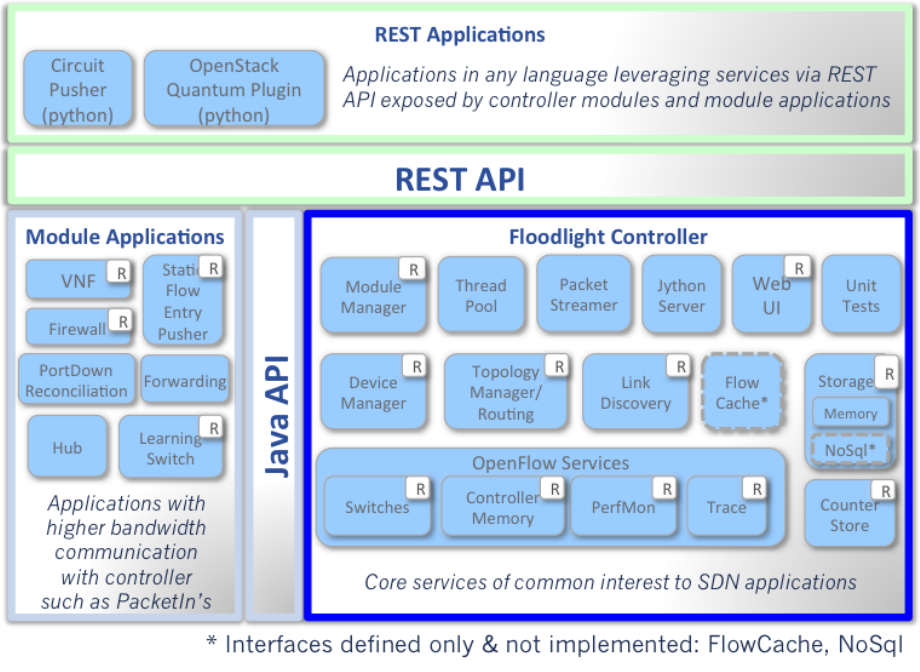
3- Network Protocol Layer : ارتباط بین کنترلرهای مختلف را از طریق رسانه های مختلف شبکه مانند اترنت و وای فای‌امکان پذیر می کند.

4- Device Connectivity Layer : رابطی را برا ی نصب اجزای سنجش ترافیک مانند سوئیچ ها و روترها به طور مستقیم در سیستم فراهم می کند.

5- Application State Management Layer : تمام حالت های برنامه فعال و پیکربندی‌شده را ردیابی می کند.

6 - User Interface Layer : پیکربندی سرویس های مختلف سیستم را از طریق یک مرورگر یا رابط کاربری گرافیکی برنامه تلفن همراه فعال می‌کند.

عکس داخل سایت :



منابع :

https://floodlight.atlassian.net/wiki/spaces

https://www.sciencedirect.com/topics/computer-science/floodlight-controller