



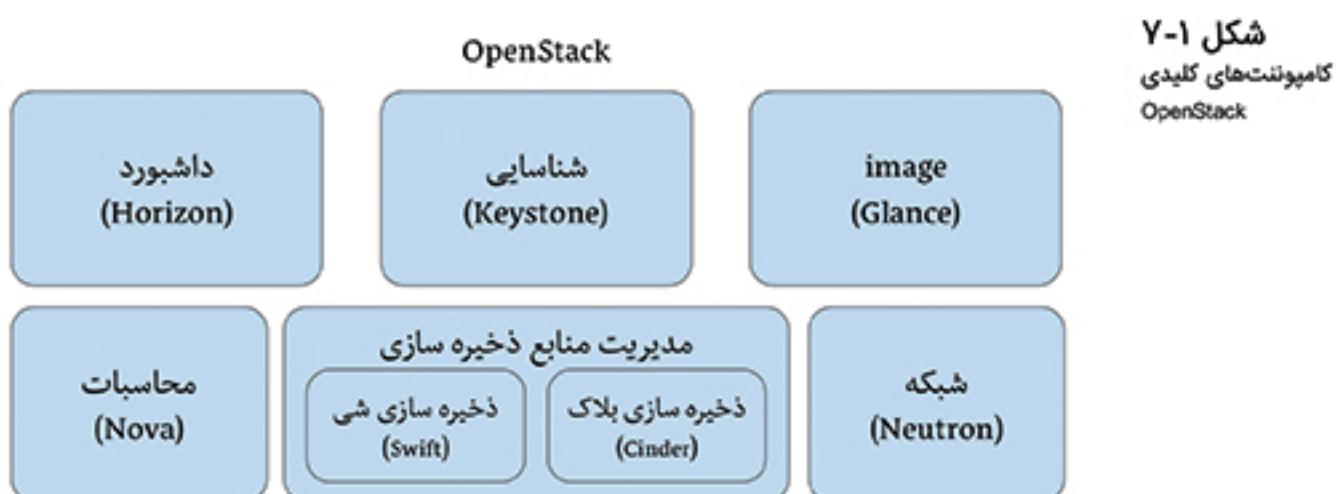
# در OpenFlow رایانش آبری

در این فصل بر نقش OpenFlow در رایانش آبری (Cloud Computing) تمرکز خواهیم کرد، سپس بطور خاص نصب و پیکربندی کامپوننت Neutron را در دستور کار خود قرار خواهیم داد. یکی از وعده‌هایی که SDN و OpenFlow می‌دهند، بهبودی است که از طریق این تکنولوژی یعنی SDN در مراکز داده و زیر ساخت رایانش آبری حاصل خواهد شد. لذا، پرداختن به کاربرد OpenFlow (برای نمونه، پلاگین Floodlight در OpenStack) در مراکز داده و به ویژه در OpenStack (به عنوان یکی از پرکاربردترین نرم افزارهای کنترل و مدیریت رایانش آبری) امری لازم و ضروری به شمار می‌رود. در این فصل یک مقدمه اجمالی از OpenStack و کامپوننت شبکه‌سازی آن یعنی Neutron، به همراه معماری کلی این کامپوننت را شرح می‌دهیم. به طور خاص، نصب و پیکربندی پلاگین Floodlight به عنوان یک کنترلر OpenFlow در این فصل توضیح داده خواهد شد. به خوانندگان علاقمند توصیه می‌شود این فصل را به عنوان نقطه‌ای آغازین برای ورود به حوزه OpenStack در نظر گرفته و مباحث جزئی تر را در مستندات OpenStack جستجو کنند.

## Neutron و OpenStack

OpenStack یک نرم افزار سیستمی رایانش آبری است (که برخی اوقات از آن تحت عنوان سیستم عامل رایانش آبری نیز یاد می‌شود)، که سرویس IaaS<sup>۱</sup> را ارائه می‌کند. OpenStack که تحت لیسانس Apache منتشر شده، نرم افزاری رایگان و متن باز است. بنیاد OpenStack در سپتامبر ۲۰۱۲ تحت عنوان یک شرکت با نهادی غیرانتفاعی تشکیل شد و از آن زمان مدیریت پروژه OpenStack را به عهده

دارد. این نهاد به جامعه توسعه‌دهندگان OpenStack برای ترویج و توسعه آن پاری می‌رساند. بدنه اصلی OpenStack را تعدادی پروژه به عنوان اجزای سازنده آن تشکیل داده‌اند. این اجزای سازنده، مجموعه‌ای از عناصر محاسباتی، منابع ذخیره‌سازی و منابع شبکه‌سازی در یک مرکز داده را کنترل می‌کنند. OpenStack محیطی را فراهم می‌آورد که مدیران بتوانند به کنترل و ارائه منابع ذکر شده، از طریق یک رابط کاربری گرافیکی (GUI) تحت وب بپردازند. معماری ماژولار OpenStack و اجزای سازنده آن (به همراه نام آنها) در شکل زیر نشان داده شده است:



بخش محاسبات Nova بوده (که بخش اصلی از یک سیستم IaaS محسوب می‌شود) و کنترل کننده اصلی رایانش آبری است. Nova در Python نوشته شده و کتابخانه‌های خارجی زیادی را مانند SQLAlchemy (برای دسترسی پایگاه داده)، Kombu (برای ارتباط پروتکل Advanced Message Queuing Programming Language (AMQP)) و Eventlet (Queuing Concurrent Programming) مورد استفاده قرار می‌دهد. Nova قادر است مجموعه‌ای از منابع کامپیوتی را مدیریت کرده و به صورت اتوماتیک درآورد. همچنین می‌تواند با تکنولوژیهای مجازی‌سازی پرکاربرد موجود و توسعه‌های مربوط به HPC تعامل داشته باشد. این برنامه بدون نیاز به سخت افزار یا نرم افزاری اختصاصی، با توان مقیاس پذیری افقی<sup>۷</sup> بر روی کامپیوترهای معمولی، طراحی شده است؛ این کامپوننت توانایی ادغام با

#### High-performance Computing<sup>۸</sup>

<sup>۷</sup> مقیاس پذیری افقی یا به عبارتی Scale Horizontally، به افزایش تعداد گره‌ها با تعداد سیستم‌های مشابه در یک بستر اشاره دارد. این عبارت، کاملاً مشابه Scale Out بوده که در برابر مقیاس پذیری عمودی (Scale Vertically) قرار می‌گیرد. مقیاس پذیری عمودی به افزایش منابع در درون یک گره یا سیستم اشاره دارد. (متترجم)

فن‌آوریهای واسطه<sup>۱</sup> (Third Party) و سیستمهای قدیمی را دارد. پلتفرم‌های مجازی ساز Xen Server و KVM که گزینه‌های معمول تکنولوژی هایپروایزر هستند، همچنین فن‌آوری Container لینوکس مانند LXC با حتی تکنولوژی Hyper-V شرکت مایکروسافت، از جمله محصولاتی هستند که OpenStack از آنها پشتیبانی می‌کند.

دو کامپوننت در OpenStack برای مدیریت ذخیره‌سازی هستند که در ادامه شرح داده می‌شوند:

**Swift**: برای مدیریت ذخیره‌سازی شیء بکار می‌رود. Swift همچنین با نام OpenStack Object Storage نیز شناخته می‌شود. این برنامه یک سیستم ذخیره‌سازی مقیاس‌پذیر و با قابلیت افزونگی است. فایلها و اشیاء در چندین دیسک روی سرورهای گوناگون در مرکز داده نوشته می‌شوند. نرم افزار OpenStack مسئول تضمین یکپارچگی داده و همانندسازی (Replication) در یک کلاستر است. با اضافه کردن سرورهای جدید، کلاسترهاي ذخیره‌سازی بسادگی قابل گسترش بوده و می‌توان فضای ذخیره‌سازی آن را افزایش داد. اگر یک سرور یا هارد دیسک دچار مشکل شود، OpenStack محتویات آن را در مکانهای جدید (دیگر هارد دیسکها) از یک کلاستر، همانندسازی کرده و داده را به آن مکان جدید منتقل می‌کند. از آنجایی که OpenStack به جای فعالیت روی قابلیتهای سخت افزاری، از الگوریتمهای نرم افزاری برای تضمین توزیع و همانندسازی داده در سراسر تجهیزات مختلف استفاده می‌کند، می‌توان از هارد دیسکها و سرورهای ارزان قیمت برای مدیریت ذخیره‌سازی استفاده کرد.

**Cinder**: این برنامه تجهیزات ذخیره‌سازی سطح بلاک را برای استفاده و تعامل با نمونه‌های پردازشی OpenStack فراهم می‌آورد. Cinder با نام OpenStack Block Storage نیز شناخته می‌شود. سیستم ذخیره‌سازی بلاک، مسئول مدیریت ایجاد کردن، متصل کردن و جدا کردن تجهیزات سطح بلاک، جهت استفاده سرورها از این تجهیزات است. ذخیره‌ساز سطح بلاک در OpenStack برای اجرای سناریوهای حساس، همچون سیستمهای فایل قابل گسترش (Expandable File Systems)، ذخیره‌ساز پایگاه داده یا برای فراهم کردن یک سرور با دسترسی به یک دستگاه ذخیره‌سازی سطح بلاک خام (Raw Block Level) مناسب است. فضاهایی که ذخیره‌سازی سطح بلاک (به این فضاهای اصطلاحاً Volume گفته می‌شود) را ارائه می‌دهند کاملاً با کامپوننتهای Nova و Horizon یکپارچه هستند. این امر کاربران آبر را قادر

<sup>۱</sup> مفهوم واسطه با همان Third Party به شرکتها، تکنولوژیها یا هر عنصری که بین دو موجودیت اصلی قرار گرفته و سرویس‌دهی می‌کنند، گفته می‌شود. به عنوان مثال سیستم عامل و بندوز و کاربری که با آن مشغول به کار هستند، به عنوان دو عنصر اصلی این مجموعه به حساب می‌آیند و به نرم افزارهایی که بصورت جداگانه توسط دیگر شرکتها نوشته شده و روی این سیستم عامل قرار می‌گیرند، نرم افزارهای واسطه گفته می‌شود. (منترجم)

می‌سازد به سادگی نیازمندیهای ذخیره‌سازی خود را مدیریت کنند. همچنین قابلیت قدرتمندی برای پشتیبان گیری داده‌های ذخیره شده در Volume‌ها، از طریق مدیریت Snapshot فراهم می‌شود. Snapshot‌ها می‌توانند برای ایجاد Volume جدیدی از نوع بلاک، روی ذخیره‌ساز بکار روند و یا به سادگی می‌توانند بازیابی شوند.

داشبورد OpenStack است. این کامپوننت یک وابط کاربری گرافیکی را برای کاربران و مدیران شبکه ایجاد کرده تا منابع مبتنی بر رایانش آبری را فراهم کنند، آن را اتوماتیک نموده و نهایتاً به آن دسترسی داشته باشند. منابع و خدمات واسط، مانند مانیتورینگ، صدور صورتحساب و دیگر ابزارهای مدیریت می‌توانند در Horizon، سفارشی‌سازی شوند. با استفاده از API بومی OpenStack و یا API با قابلیت سازگاری با سرویس Amazon EC2، توسعه دهنده‌گان می‌توانند دسترسی‌ها را اتوماتیک کرده یا برای مدیریت منابع، ابزارهای سفارشی شده‌ای را بر حسب نیاز ایجاد کنند. API‌های OpenStack با سرویسهای Amazon S3 و Amazon EC2 سازگار هستند. بنابراین اپلیکیشن‌های سمت کاربر که برای سرویسهای وب آمازون طراحی می‌شوند، می‌توانند با OpenStack بکار روند.

کامپوننت شناسایی OpenStack است. این کامپوننت یک دایرکتوری مرکزی از کاربران فراهم می‌آورد که در این دایرکتوری، کاربران به سرویسهای قابل دسترس OpenStack، نگاشت شده‌اند. این کامپوننت همچون یک سیستم احراز هویت متدال در سیستم عامل آبر عمل می‌کند. حتی می‌تواند با سرویسهای دایرکتوری Backend موجود مانند LDAP یکپارچه شود. گواهی‌های استاندارد نام کاربری و رمزعبور، سیستمهای مبتنی بر سخت افزار Token و ورود به سرویسهای وب آمازون، مکانیزم‌های گوناگون احراز هویتی هستند که توسط Keystone پشتیبانی می‌شوند.

Glance که سرویس مربوط به فایلهای Image در OpenStack است، سرویسهای کشف (Discovery)، ثبت (Registration) و تحويل (Delivery) را برای دیسکها و فایلهای Image مربوط به سرورها، فراهم می‌آورد. فایلهای Image ذخیره شده متعلق به سرور می‌توانند به عنوان یک قالب (Template) مورد استفاده قرار گیرند. همچنین این فایلهای Image می‌توانند برای ذخیره و فهرست کردن (Catalog) تعداد بی‌نهایی از نسخ پشتیبان بکار روند. Glance می‌تواند Image‌های سرور و دیسک را در کامپوننتهای گوناگون از جمله Swift، ذخیره کند. یک اینترفیس<sup>۵</sup> REST استاندارد، توسط کامپوننت Glance برای دریافت اطلاعات فایلهای Image ارائه شده است و مشتریها را قادر می‌سازد فایلهای Image خود را به سرورهای جدید ارسال کنند.

---

<sup>۵</sup> REST که مخفف عبارت Representational State Transfer است یک معماری وب سرویس به شمار می‌رود که از HTTP برای انتقال اطلاعات میان کاربر و سرور استفاده می‌کند. کار کردن با REST بسیار ساده‌تر از وب سرویسهای پیچیده‌ای مانند SOAP است. (مترجم)

**Neutron** (که در گذشته با نام Quantum شناخته شده بود) کامپوننت شبکه‌سازی OpenStack است. این کامپوننت شبکه‌ها و آدرس‌های IP را مدیریت می‌کند. Neutron از نسخه منتشر شده Folsom<sup>\*</sup> به بعد، به طور عملیاتی در بخش مرکزی و مورد پشتیبانی پلتفرم OpenStack است. مانند دیگر کامپوننتهای سیستم عامل آبری، مدیران و کاربران می‌توانند Neutron را بکار ببرند تا مسطح استفاده از منابع موجود در مرکز داده را افزایش دهند. Neutron می‌تواند خدمات NaaS را بین تجهیزات فراهم کند (بطور نمونه vNIC)، که این خدمات از طریق دیگر سرویس‌های OpenStack مدیریت می‌شود. OpenStack Neutron مدل‌های شبکه‌سازی را برای گروه‌های مختلف کاربری و یا اپلیکیشنها، فراهم می‌آورد. مدل‌های استاندارد برای جدایی ترافیک شبکه در سرورهای گوناگون، عبارتند از شبکه‌های چند لایه با استفاده از VLAN‌ها یا شبکه‌های مسطح. Neutron همچنین آدرس‌های IP را مدیریت می‌کند. این آدرسها می‌توانند IP‌های ایستای<sup>†</sup> اختصاص داده شده توسط مدیر شبکه باشند یا می‌توانند آدرس دهی IP مبتنی بر DHCP باشند. آدرس دهی IP شناور، به ترافیک بسته‌ها اجازه می‌دهد که بصورت پویا به هر کدام از گره‌های محاسباتی تغییر مسیر دهند. این عمل در نتیجه، جهت‌یابی ترافیک در خلال مهاجرت ماشین مجازی، نگهداری، یا رسیدگی به خطاهای را تسهیل می‌بخشد. معناری قابل توسعه Neutron، راه را برای سرویس‌های ارزش افزوده شبکه ای، مانند فایروال‌ها، سیستم‌های تشخیص نفوذ (IDS)، شبکه‌های مجازی خصوصی (VPN) و Load Balancer را هموار ساخته تا این سیستم‌ها استقرار یافته و مدیریت شوند. کامپوننت شبکه‌سازی OpenStack، کاربران را با یک API اختصاصی تجهیز می‌کند تا توپولوژیهای شبکه غنی بسازند و برای ساختن توپولوژی اپلیکیشن وеб چند لایه، سیاستهای پیشرفته شبکه را پیکربندی کنند. ساختار ماژولار Neutron، توسعه پلاگینهای خلاقانه‌ای را که قابلیتهای موجود در شبکه پیشرفته را ارائه می‌دهند، تسهیل می‌بخشد (مانند تولنل زنی L2-in-L3 برای عبور از محدودیتهای VLAN عدد ۴۰۹۶، تضمینهای QoS خاص end-to-end و بکارگیری پروتکلهای مانیتور کردن، مانند NetFlow و پلاگینهای OpenFlow). علاوه بر این، توسعه دهنده‌گان می‌توانند سرویس‌های شبکه‌ای پیشرفته‌ای را ایجاد کنند که این سرویسها درون شبکه Tenant‌های مربوط به OpenStack که از پلاگینها استفاده می‌کنند، یکپارچه می‌شوند. برای مثال Load-Balancing-aaS و VPN-aaS، Firewall-aaS، IDS-aaS، Data-Center-Interconnect-aaS تعدادی از سرویس‌های پیشرفته شاخص هستند. با استفاده از Neutron، کاربران می‌توانند شبکه‌های

\* همانطور که نویسنده در کتاب ذکر کرده است، نسخه Folsom، کامپوننتهای شبکه‌سازی را به درون خود اضافه کرد. البته در اصل از نسخه HAVANA که در سال ۲۰۱۲ منتشر شد، کامپوننت Neutron ارائه شد و قبلاً از آن Quantum مورد استفاده بوده است. در هنگام ترجمه این کتاب، نسخه Newton آخرین نسخه OpenStack است که انتشار یافته است. (متترجم)

<sup>†</sup> Static IP Address

<sup>‡</sup> عبارت Tenant که در OpenStack با نام Project هم خوانده می‌شود، به مجموعه‌ای از منابع محاسباتی، شبکه‌ای و ذخیره‌سازی گفته می‌شود که روی بستر ابر قرار داشته و برای دسترسی یک با چندین کاربر ایجاد شده است. (متترجم)

خود را ایجاد، ترافیک را کنترل و سرورها و دستگاه‌ها را به یک یا چند شبکه متصل کنند؛ در حالی که مدیران شبکه می‌توانند از مزایای تکنولوژی SDN (به طور مثال OpenFlow) بهره‌مند شوند تا سطوح بالایی از Multi Tenancy و مقیاس‌پذیری را فراهم کنند.

### معماری شبکه‌سازی OpenStack

کامپوننت Neutron قادر است مجموعه‌ای از سرویس‌های Backend را با نام پلاگین بکار ببرد که از یک مجموعه در حال رشد از تکنولوژی‌های شبکه، پشتیبانی می‌کند. این پلاگینها ممکن است بصورت جداگانه یا بصورت بخشی از نسخه منتشر شده اصلی Neutron توزیع شوند. سرویس شبکه‌سازی Neutron یک سرویس مجازی شبکه است که یک API کارآمد فراهم می‌کند تا اتصال شبکه و آدرس‌دهی IP را تعریف کند. این سرویس شبکه‌سازی، توسط تجهیزات مربوط به دیگر سرویس‌های OpenStack (مانند OpenStack Compute) مورد استفاده قرار می‌گیرد. API شبکه‌سازی OpenStack ویزگی‌هایی همچون شبکه مجازی، Subnet و انتزاعهای پورت را بکار برد و تابع OpenStack شبکه‌سازی را توصیف کند. در اکوسیستم شبکه‌سازی OpenStack:

- شبکه، یک بخش لایه ۲ ایزوله شده است که می‌توان آن را مشابه با مفاهیم VLAN در شبکه فیزیکی، قلمداد کرد.
  - یک دسته از آدرس‌های IPv6 یا IPv4 و حالت‌های پیکربندی مرتبط به آن آدرس IP، یک Subnet است.
  - نقطه ارتباطی برای متصل کردن یک دستگاه (همچون کارت شبکه مربوط به سرور مجازی) به یک شبکه مجازی را، پورت می‌نامند. همچنین یک پورت، پارامترهای پیکربندی شبکه را (مانند MAC و IP آدرسها) که بدان مرتبط هستند، توصیف می‌کند.
- با ایجاد و پیکربندی شبکه‌ها و Subnet‌ها، کاربران می‌توانند طیف گسترده‌ای از تopolوژی‌های شبکه OpenStack Compute مانند OpenStack می‌چشیده و غنی را پیاده‌سازی کنند و سپس دیگر سرویس‌های Tenant را آموختند تا اینترفیس‌های مجازی را به پورتهای این شبکه‌ها متصل کنند. Neutron به ویژه از Tenant‌هایی که شبکه‌های خصوصی گوناگون دارند حمایت می‌کند و Tenant‌ها را قادر می‌سازد رنج آدرس IP خود را انتخاب کنند. سرویس شبکه‌سازی OpenStack سناریوهای شبکه سازی آبری پیشرفته را فراهم می‌آورند. از جمله این سناریوهای می‌توان ساخت اپلیکیشن‌های وب چند لایه یا توانمند سازی اپلیکیشن‌ها برای مهاجرت به آبر، بدون تغییر آدرس‌های IP، نام برد.
- مدیران شبکه آبری را قادر می‌سازند تا پیشنهاداتی برای یک شبکه انعطاف‌پذیر و سفارشی ارائه کنند.

- باعث توسعه API می‌شود. این API به مدیران شبکه آبری اجازه می‌دهد قابلیت‌های دیگری از API را به نمایش بگذارند. این قابلیت‌های جدید عموماً به عنوان یک بخش توسعه یافته در API معرفی می‌شوند و به تدریج به قسمتی از API شبکه‌سازی OpenStack مرکزی، تبدیل می‌شوند.

شبکه OpenStack Compute اصلی، مدل بسیار ساده‌ای از جدایی ترافیک را از طریق IP Table ها و VLAN های لینوکس، پیاده‌سازی می‌کند. شبکه‌سازی OpenStack ایندۀ وجود پلاگینی را معرفی می‌کند که عبارت است از پیاده‌سازی Backend API مربوط به API شبکه‌سازی OpenStack. یک پلاگین می‌تواند تکنولوژی‌های گوناگونی را برای پیاده‌سازی درخواست‌های API بکار برد. برخی از پلاگینهای شبکه‌سازی OpenStack ممکن است VLAN های اولیه لینوکس و IP Table ها را بکار ببرند، در حالی که دیگر پلاگینها می‌توانند تکنولوژی‌های پیشرفته تر مانند تونل زنی L2-in-L3 با OpenFlow را برای فراهم آوردن قابلیت‌های مشابه بکار ببرند.

ماژول اصلی سرور شبکه سازی OpenStack، عبارت است از Neutron-server، که برنامه‌ای اجرا شده در پس زمینه (Deamon) به زبان Python بوده و API مربوط به ماژول شبکه‌سازی OpenStack یعنی OpenStack Network API را نمایش می‌دهد. این ماژول (یعنی Neutron-server) برای پردازش اضافی، درخواست‌های کاربر را به پلاگین پیکربندی شده شبکه سازی OpenStack انتقال می‌دهد.

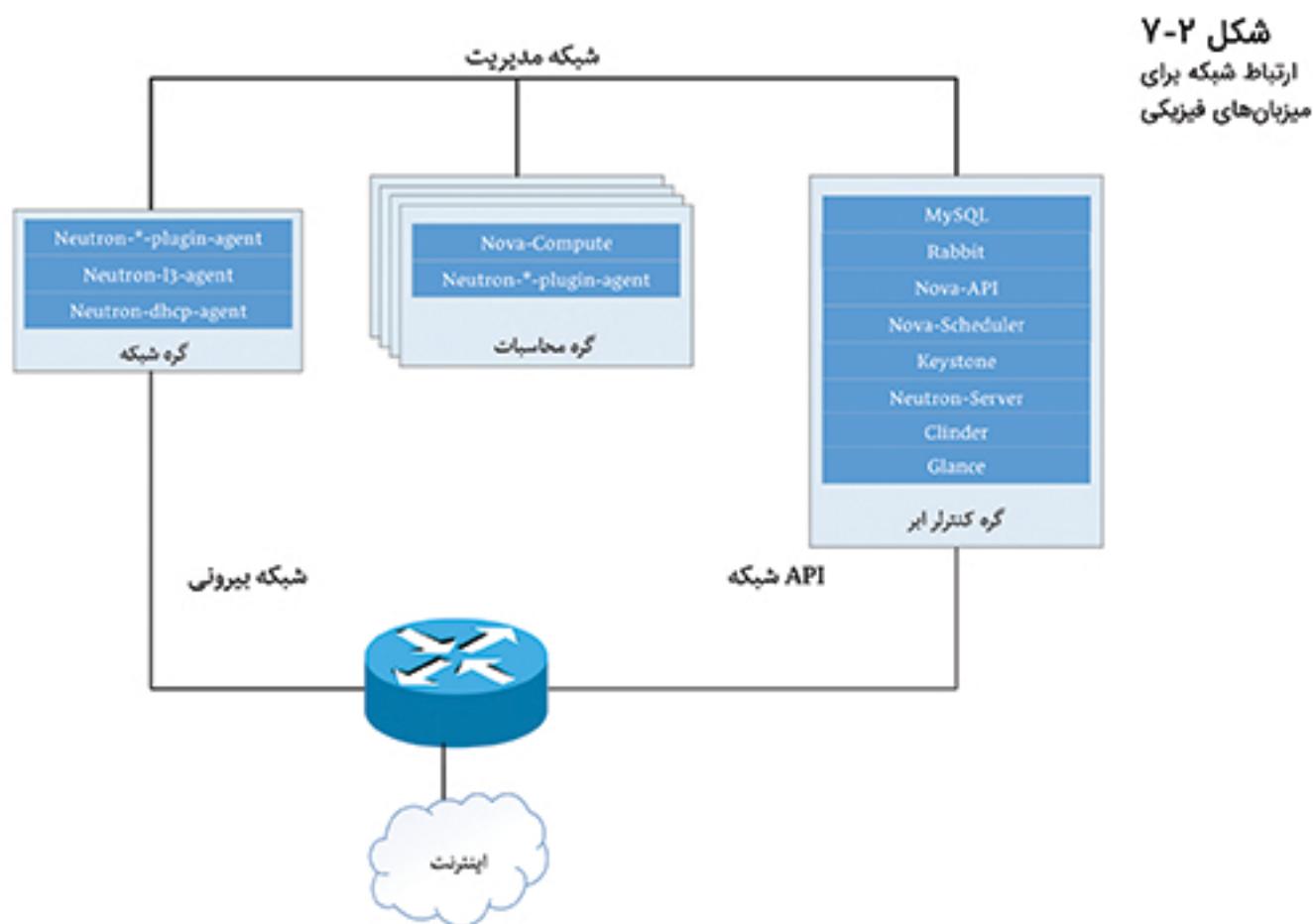
پلاگین بطور معمول به یک پایگاه داده برای ذخیره‌سازی همیشگی داده‌ها نیاز دارد. اگر مجموعه استقرار یافته شما، از یک کنترل کننده (یک میزبان) برای اجرای کامپوننتهای OpenStack Compute متمرکز استفاده کند، می‌توانید سرور شبکه سازی OpenStack را بر روی همان میزبان مستقر سازید. با این حال، ماژول شبکه سازی OpenStack کاملاً مستقل است و می‌تواند بر روی سرور جداگانه‌ای استقرار یابد. بر اساس این استقرار، شبکه سازی OpenStack Agent هایی است که ممکن است مورد نیاز باشند:

- پلاگین Agent، با نام `neutron-*-agent`، روی هر هایپروایزدی اجرا می‌شود تا یک سوئیچ محلی را پیکربندی کند. Agent مورد استفاده که باستی اجرا شود، وابسته به پلاگین انتخابی خواهد بود. می‌توان اینگونه عنوان کرد که برخی پلاگینها در واقع به Agent احتیاج ندارند.

• DHCP Agent با نام `neutron-dhcp-agent`، سرویسهای DHCP را برای شبکه‌های Tenant مربوط به می‌آورد.

- لایه ۲ با نام `neutron-l3-agent`، ارسال بسته در لایه ۲ یا عملیات NAT را جهت تسهیل دسترسی ماشینهای مجازی درون شبکه Tenant، به شبکه بیرونی فراهم می‌کند. این Agent ها با بخش پردازش Neutron مرکزی از طریق روش فراخوانی راه دور (RPC) یا با استفاده از API استاندارد مربوط به کامپوننت شبکه سازی OpenStack، به تعامل می‌پردازند. کامپوننت

شبکه‌سازی OpenStack برای احراز هویت و کسب اجازه از تمام درخواستهای API، بر Keystone تکیه می‌کند. کامپوننت Nova از طریق فراخوانیهای API استاندارد با کامپوننت شبکه‌سازی OpenStack یعنی Neutron، تعامل می‌کند. در فرایند ایجاد ماشین مجازی، Nova با این API ارتباط برقرار می‌کند تا هرگدام از کارت‌های شبکه مجازی در ماشین مجازی را به شبکه‌ای خاص وصل کند. کامپوننت API شبکه‌سازی OpenStack به تعامل می‌پردازد و کاربران Tenant و مدیران شبکه را قادر می‌سازد که سرویسها را از طریق محیط گرافیکی OpenStack ایجاد و مدیریت کنند. در ایجاد کامپوننت شبکه‌سازی OpenStack استاندارد، چهار دسته شبکه مرکز داده فیزیکی مجزا دخیل هستند؛ مانند آنچه در تصویر نشان داده شده است:



- شبکه مدیریت<sup>۱</sup>: برای ارتباط درونی بین کامپوننتهای OpenStack بکار می‌رود. تخصیص آدرسهای IP بایستی بگونه‌ای انجام پذیرد که تنها در داخل شبکه مرکز داده قابل دسترس باشند.
- شبکه داده<sup>۲</sup>: برای ارتباط داده‌ای ماشین مجازی درون تنظیمات آبری بکار می‌رود. بر اساس نوع پلاگین شبکه‌سازی‌ای که مورد استفاده قرار می‌گیرد، نیازمندیهای آدرس دهی IP نیز متفاوت می‌شود.
- شبکه بیرونی: برای فراهم نمودن دسترسی اینترنت برای ماشینهای مجازی در برخی توپولوژیهای شبکه بکار می‌رود. آدرس‌های IP روی این شبکه باید بوسیله هر میزبانی در اینترنت قابل رویت و در دسترس باشد.
- شبکه API: تمام API‌های درون OpenStack را از جمله API شبکه‌سازی OpenStack، برای Tenant‌ها نمایش می‌دهد. آدرس‌های IP در این شبکه باید بوسیله هر شخصی در اینترنت قابل دسترس باشد.

[ نصب کامل و دستورالعملهای پیکربندی OpenStack Neutron را می‌توان در راهنمای مدیریت کامپوننت شبکه‌سازی OpenStack یافت. برای اطلاعات بیشتر به آدرس زیر مراجعه نمایید:  
<http://wiki.openstack.org/wiki/neutron> ]

### پلاگین‌های Neutron

ایجاد شبکه آبری غنی از طریق بهبود راه حل‌های شبکه‌سازی سنتی، کاملاً چالش برانگیز است. شبکه‌سازی سنتی نه از نظر طراحی، قابل قیاس با ویژگیهای آبر است و نه می‌تواند از عهده پیکربندی اتوماتیک عناصر شبکه برآید. شبکه‌سازی OpenStack مفهوم یک پلاگین را معرفی کرده که نوعی پیاده‌سازی API از شبکه‌سازی OpenStack، محسوب می‌شود. به منظور پیاده‌سازی درخواستهای API منطقی، یک پلاگین می‌تواند تکنولوژیهای گوناگونی را مورد استفاده قرار دهد. برخی پلاگینها ممکن است ویژگی IP Tables موجود در لینوکس و VLAN را مورد استفاده قرار دهند، در حالی که پیاده‌سازیهای دیگر ممکن است از تکنولوژیهای پیشرفته‌تر مانند تونل‌زنی L2-in-L3 و OpenFlow استفاده کنند. پلاگینها می‌توانند مشخصه‌های متفاوتی را برای نیازهای سخت افزاری،

<sup>۱</sup> Management Network

<sup>۲</sup> Data Network

ویژگیها، کارائی، مقیاس پذیری یا ابزارهای اپراتور به همراه داشته باشند. OpenStack طیف وسیعی از پلاگینها را مورد پشتیبانی خود قرار می‌دهد. بنابراین، مدیر بستر آبری قادر است گزینه‌های متفاوتی را در نظر گرفته و تصمیم بگیرد که کدام تولوژی شبکه با طرح مورد استفاده خود، مناسب است. در میان پلاگینهای گوناگون برای Neutron، ما به بررسی پلاگین کنترلر Floodlight در کامپونت Neutron خواهیم پرداخت.

با بکارگیری پلاگین Neutron، کنترلر Floodlight می‌تواند به عنوان یک عنصر Backend در شبکه برای OpenStack اجرا شود. یک مدل NaaS را از طریق یک REST API به نمایش می‌گذارد که توسط Floodlight پیاده‌سازی شده است. این راه حل شامل دو کامپونت اصلی است: پلاگین Floodlight که Neutron را به Floodlight متصل می‌کند و یک مازول VirtualNetworkFilter در FloodLight که باعث پیاده‌سازی Neutron API می‌شود. مازول VirtualNetworkFilter عملیات جداسازی شبکه لایه ۲ مبتنی بر MAC را در شبکه‌های OpenFlow پیاده‌سازی می‌کند و آن را از طریق یک REST API نمایش می‌دهد. این مازول بطور پیش فرض در Floodlight گنجانده شده است و برای فعال شدن و اجرا به Neutron و یا OpenStack متنکی نیست. VirtualNetworkFilter می‌تواند از طریق تغییر فایل پیکربندی که در ادامه این فصل توضیح داده خواهد شد، فعال شود. پلاگین RestProxy برای اجرای بخشی از سرویس Neutron متعلق به OpenStack، طراحی شد. کنترلر Floodlight به همراه پلاگین Big Switch Neutron، از نسخه منتشر شده OpenStack Grizzly پشتیبانی می‌کند.

**[ پشتیبانی کنترلر Floodlight در OpenStack از دو طریق فعال می‌شود:**

۱. پلاگین مربوط OpenStack Neutron در مخزن اصلی Big Switch Neutron  <http://github.com/openstack/neutron>

دستورالعملهای زیر با استفاده از اسکریپتهای Devstack (که توسط شرکت Big Switch توسعه داده شده‌اند) برای راه اندازی کنترلر Floodlight و نسخه Grizzly سیستم عامل OpenStack روی یک ماشین مجازی با سیستم عامل Ubuntu، بکار می‌روند. به عنوان پیشنباز نصب و راه اندازی یک ماشین مجازی با سیستم عامل Ubuntu نسخه 12.04.1 یا نسخه بالاتر، لازم است. نتیجه این فرایند، نصب یک نسخه از OpenStack، به همراه کنترلر Floodlight است که بعنوان یک مازول برای Neutron در سرویس دهنده می‌کند. Tenant‌های محیط آبری، شبکه‌های مجازی و دیگر عناصر مجازی را می‌توان با داشبورد OpenStack یعنی Horizon ایجاد نمود.

لازم است برای عملکرد صحیح کامپوننت Neutron، یک کنترلر Floodlight را اجرا کنید. کنترلر Floodlight را می‌توانید بر روی یک ماشین مجازی مجازاً اجرا کنید یا می‌توانید گُدمب منبع Floodlight را بصورت یک فایل Zip دانلود کرده و پس از Unzip کردن، آن را کامپایل نمایید. سپس با انجام مراحل ساده زیر در ماشین مجازی Ubuntu، آن را اجرا کنید. قبل از ادامه کار، اطمینان حاصل کنید که به اینترنت دسترسی دارید.

```
$ sudo apt-get update
$ sudo apt-get install zip default-jdk ant
$ wget --no-check-certificate
https://github.com/floodlight/floodlight/archive/master.zip
$ unzip master.zip
$ cd floodlight-master; ant
$ java -jar target/floodlight.jar -cf
src/main/resources/neutron.properties
```

برای آنکه از قابل بودن VirtualNetworkFilter روی ماشین مجازی Ubuntu اطمینان حاصل کنید، فرمان زیر را وارد کنید:

```
$ curl 127.0.0.1:8080/networkService/v1.1
{"status": "ok"}
```

زمانیکه از در حال اجرا بودن کنترلر Floodlight مطمئن شدید، با استفاده از اسکریپت-Install، برنامه OpenStack را نصب کنید. اسکریپت فوق مراحل نصب را به شرح زیر انجام می‌دهد:

۱. سوچیج OVS را روی ماشین مجازی، پیغام‌های پیکربندی می‌کند که این پیکربندی باعث می‌شود سوچیج روی پورت خاصی به کنترلر Floodlight گوش دهد.
۲. سپس، OpenStack و پلاگین پرایسی REST Big Switch را روی ماشین مجازی نصب می‌کند.
۳. اگر قصد نصب نسخه Grizzly سیستم عامل OpenStack را دارید، از فرمان زیر استفاده کنید:

```
$ wget https://github.com/openstack-
dev/devstack/archive/stable/grizzly.zip
$ unzip grizzly.zip
$ cd devstack-stable-grizzly
```

۴. اگر قصد نصب نسخه منتشر شده Folsom را دارید، از فرمان زیر استفاده کنید:
- ```
$ wget
https://github.com/bigswitch/devstack/archive/floodlight/folsom.z
ip
```

```
$ unzip folsom.zip
$ cd devstack-floodlight-folsom
```

۵. با ویرایشگر مورد نظر، فایلی با نام `localrc` ایجاد کرده و جزئیات زیر را در آن وارد کنید.  
 فراموش نکنید محتوای مربوط به `<password>` را با رمز عبور مورد نظر خود جایگزین کرده و  
`BS_FL_CONTROLLERS_PORT=<floodlight IP address>`  
 بروز کنید تا بتوان محیط گرافیکی Openstack را روی پورت ۸۰۸۰ مشاهده کرد. اگر  
 Floodlight را در همان ماشین مجازی اجرا کرده‌اید، آدرس IP محلی، یعنی `127.0.0.1` را  
 برای پارامتر `<floodlight IP address>` استفاده کنید؛ در غیر اینصورت، آدرس IP  
 ماشین مجازی یا میزبانی را که Floodlight روی آن اجرا می‌شود، استفاده کنید.

```
disable_service n-net
enable_service q-svc
enable_service q-dhcp
enable_service neutron
enable_service bigswitch_floodlight
Q_PLUGIN=bigswitch_floodlight
Q_USE_NAMESPACE=False
NOVA_USE_NEUTRON_API=v2
SCHEDULER=nova.scheduler.simple.SimpleScheduler
MYSQL_PASSWORD=<password>
RABBIT_PASSWORD=<password>
ADMIN_PASSWORD=<password>
SERVICE_PASSWORD=<password>
SERVICE_TOKEN=tokentoken
DEST=/opt/stack
SCREEN_LOGDIR=$DEST/logs/screen
SYSLOG=True
#IP:Port for the BSN controller
#if more than one, separate with commas
BS_FL_CONTROLLERS_PORT=<ip_address:port>
BS_FL_CONTROLLER_TIMEOUT=10
```

۶. سپس فرمان زیر را وارد کنید:

```
$ ./stack.sh
```

به ياد داشته باشيد که نصب OpenStack فرایندی زمانبر است که نباید تداخلی در آن به وجود بیابد.  
 هر گونه تداخل یا قطعی ارتباط شبکه، منجر به نتایج ناشناخته‌ای می‌شود که ادامه نصب را غیر ممکن  
 می‌سازد. توصیه می‌شود قبل از آغاز نصب، با استفاده از نرم افزار VirtualBox یک Snapshot تهیه  
 کنید. بدین طریق در صورت اختلال در فرایند نصب، می‌توانید به سادگی دستگاه را خاموش و

اصلی را بازیابی کرده و دوباره فرایند نصب را از سر بگیرید. اسکریپت `install-snapshot` برای اجرا، به یک ارتباط شبکه‌ای بدون اختلال نیاز دارد. اگر نصب برنامه با موفقیت به پایان برسد، تصویر زیر را خواهد دید:

**۷-۲**  
نصب موفقیت آمیز  
OpenStack

```
Horizon is now available at http://10.10.2.15 /  
Keystone is serving at http://10.10.2.15:5000/v2.0/  
Examples on using novaclient command line is in  
exercise.sh  
The default users are: admin and demo  
The password: nova  
This is your host ip: 10.10.2.15  
stack.sh completed in 103 seconds.
```

می‌توانید درستی نصب OpenStack و Floodlight را توسط دستورالعمل‌های موجود در لینک زیر بررسی کنید:

<https://floodlight.atlassian.net/wiki/display/floodlightcontroller/Install+Floodlight+and+OpenStack+on+Your+Own+Ubuntu+VM#suk=1>



### خلاصه

پروژه‌ای در حوزه OpenStack بوده که سرویس NaaS را در سراسر تجهیزات شبکه‌ای (کارت‌های شبکه مجازی) که با دیگر سرویس‌های OpenStack همچون Nova، مدیریت می‌شوند، فراهم می‌آورد. در نسخه Folsom، کامپوننت Neutron بخش هسته‌ای و پشتیبانی کننده چهارچوب OpenStack است. در این فصل، اجزای اصلی و کلیدی OpenStack از جمله کامپوننت شبکه سازی Neutron و پلاگین‌های Backend به ویژه پلاگین Floodlight، معرفی شدند. API شامل پشتیبانی از شبکه لایه ۲ و مدیریت آدرسهای IP<sup>۱۰</sup> می‌شود. پلتفرم توسعه‌پذیر API، شامل الحاقاتی برای شبکه ارائه دهنده (Provider Network) بوده که شبکه لایه ۲ Neutron را به VLAN خاصی در مرکز داده فیزیکی نگاشت می‌کند. در ساختار Neutron، مسیریاب‌های لایه ۲ ساده‌ای وجود دارند که مسیریابی بین شبکه‌های لایه ۲ را پشتیبانی می‌کنند. همچنین با پشتیبانی از آدرسهای IP شناور، یک Gateway را برای دسترسی به شبکه‌های بیرونی فراهم می‌کنند. در فصل آخر از این کتاب، منتخبی از پروژه‌های متعدد باز کلیدی حول محور SDN و OpenFlow ارائه خواهد شد.