CHAPTER 6

Network Security in the Cloud

The cloud can't exist without a network. It is the network that glues cloud-based applications to its users. It is the network that connects applications to the Internet, making them widely available. It is also the network that provides redundant paths between cloud-based applications and users, which makes them business worthy and reliable. Finally, the network can provide a number of security functions that further enable end-to-end security in the cloud.

Boot integrity of the network infrastructure is a prerequisite to trust and enables security functions in the network. The concepts, architecture, and technology components we discussed in the previous chapters on platform trust, attestation, and asset tagging are all equally applicable to the network infrastructure. In this chapter, we look beyond the integrity of the server platforms, and cover concepts relating to network security functions and their essential role in enabling trusted clouds. We look at how companies like M2Mi are automating the many steps required to enable the network security functions via high-level programmatic APIs, and we show how this automation is having a direct impact on the security, scale, and automation of clouds. We will also briefly examine software-defined networks (SDN), an emerging technology bringing solutions that seem to address some of key requirements of cloud computing and that has implications for network security.

As mentioned in previous chapters, cloud computing provides an on-demand virtual infrastructure enabling consumers of the cloud to easily manage their applications. One of the goals of cloud computing is to provide services that abstract the complexity of the cloud and make it simple to manage applications contained within the cloud. Application owners should be able to easily manage their applications without having to know the complexity or the details of the cloud and how is constructed. One of the most important components of the cloud is the network, so we begin with that.

The Cloud Network

The network can be thought of as the glue that holds cloud applications and users together. If the network is the glue, then one might ask how it works. What would a cloud-based network look like? Let's address these questions by examining what a basic network is and work our way to some complex examples found in modern cloud-based networks.

Network Security Components

The most basic network consists of computers connected to a switch, as shown in Figure 6-1. In this case the computers' network port has a cable that connects to a switch. The network switch is the device that enables communications between computers in the network. This simple type of network is commonly found in homes and/or small offices.

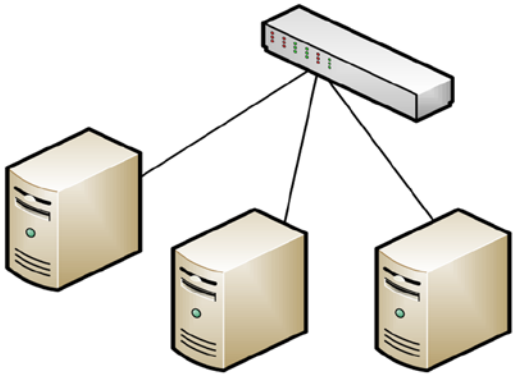

Figure 6-1. *Computers connect to the network through a switch*

If we wish to connect this network to the Internet, then we need to add possibly two network devices. The first device is a firewall, which is used to protect the network from malicious attacks. The second device is a router, used to forward network traffic from the local network to the Internet. Quite often the functions of the firewall and router are consolidated in one device. This scheme is depicted in Figure 6-2.

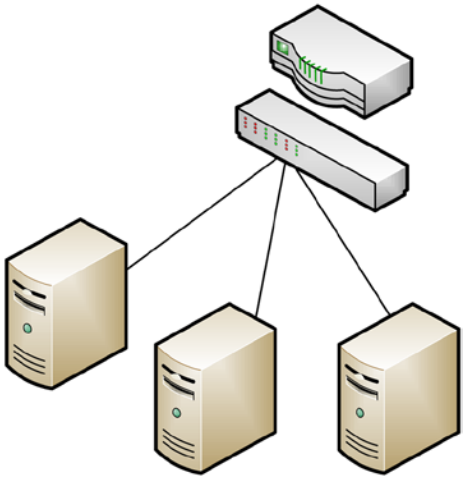

Figure 6-2. *Simple network with a switch and a firewall*

The main concerns and functions of networks are to allow communication between devices connected to the network. In a modern data center hosting a cloud computing environment, the network is much more complex. Nonetheless, it is composed of many of the devices found in a simpler network, except they are in greater numbers and have increased functionality. For example, in a data center there would be a large number of racks housing servers. The servers are connected to switches contained at the top of each rack. These are commonly referred to as *access switches*, or top of rack (TOR) switches; see Figure 6-3. These switches are normally deployed in pairs to provide failover capability and redundancy.

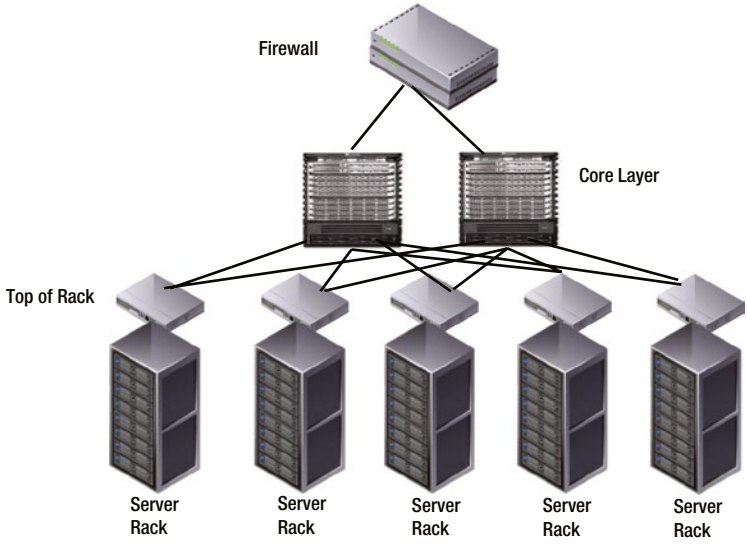

Figure 6-3. *Network racks connected to distribution switches*

There could be tens, hundreds, and even thousands of these racks distributed in a data center. The access switches are connected in turn to *distribution switches*, otherwise known as aggregation switches. These switches aggregate the access switch connections and provide the pathway out of the network into a firewall or a router.

There are a number of optional, but commonly found components in cloud-centric networks, such as load balancers, intrusion detection devices, and application delivery controllers (ADCs). The idea behind these components is to inspect network traffic and perform a function upon it. Let's look at each of these briefly.

Load Balancers

The main function of load balancers is to balance traffic between web servers and application clients. For example, a website could be composed of several web servers in order to handle a high number of client requests and provide redundancy in case one fails. The load balancer distributes the client requests among the web servers, based on a distribution algorithm such as a round robin or web server load.

Intrusion Detection Devices

These devices monitor the network by looking for malicious malware such as viruses or cyber-attacks attempting to penetrate sensitive systems. When these attacks are discovered, an intrusion detection system can log the event and notify network administrators, or it could possibly take an action to prevent the attack, such as creating a firewall policy rule to block attacks.

Application Delivery Controllers

These devices can be considered an evolution of load balancers. They can load balance network traffic and perform advanced tasks such as inspecting traffic to detect and avoid IP fragmentation, data rate shaping, SSL offloading, and analyzing data and transactions in real time. They can also protect against targeted attacks like cross-site scripting, SQL injection, cookie poisoning, forceful browsing and invalid input.

End-to-End Security in a Cloud

When an architect designs a data center to host a public, private, or hybrid cloud, a primary consideration is end-to-end security. The architect analyzes security all the way from application clients, such as a laptops and hand-held devices, to the data center, where applications are housed. The path of the client requests is noted, and how the data traverses the devices, hosts, virtual machines, and backend storage is studied.

For example, a typical web application could flow as follows: From a web browser through a firewall over the Internet, it arrives at a data center's router, passes through a firewall and distribution switch, to reach a load balancer and application delivery controller. The load balancer redirects the traffic to an application server or web server running in a virtual machine; the application receiving the traffic may then access backend data based on the nature of the traffic. This flow is shown in Figure 6-4.

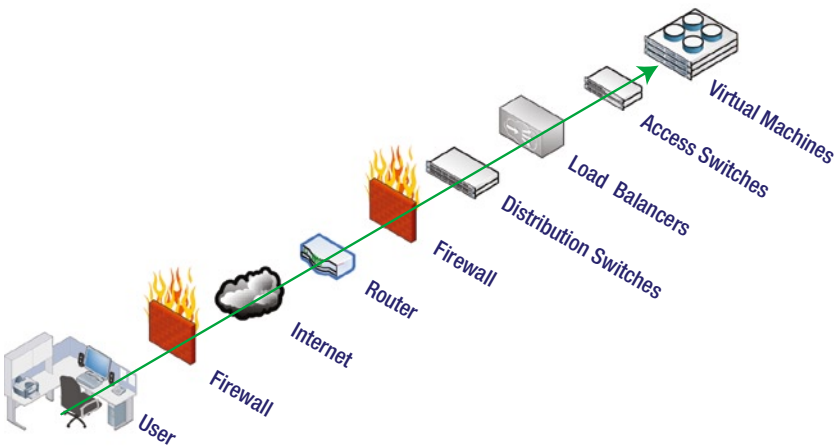

Figure 6-4. *Trajectory of a user request*

While this chain may seem excessively lengthy, it is actually just one of many traffic flows to consider. An architect will diagram and note all possible network and data flows and track them. The architect then looks at each participant of the end-to-end flow and considers how each step needs to be secured, as well as thinks about what would happen to the others if its security were compromised. For example, a security architect may consider how to secure the backend block storage used by databases and virtual machines. Backup, application, and administrative access to the block storage are examined. After analyzing the network flow, the architect may decide to encrypt selected data and apply enhanced firewall rules to restrict access.

Network security: End-to-End security: Firewalls

In the example above, we have explored the components of a network. The network in which an application resides must be secure if that application is to be secure. There are a number of means by which this is accomplished.

Firewalls and routers are the front-line defense in a network. Most modern routers have firewall capabilities, such as screening for malformed packets and blocking inappropriate protocols and ports. Modern firewalls can filter inbound traffic and sessions, and apply policies that block unwanted traffic.

Firewalls can also support dedicated virtual private networks (VPNs) for remote office connectivity and encrypt traffic between branch offices.

Network security: End-to-End security: VLANs

Virtual local area networks, otherwise known as VLANs, allow the segmentation of network traffic over a network. VLANs are typically assigned based on requirements such as application, bandwidth, or user access. For example, in a private cloud there could be VLANs for the engineering, human resources, and accounting divisions. Another example is a dedicated VLAN that is used by system and network administrators for managing servers and network devices.

There are a number of ways to lock down and secure VLANs so they don't become compromised:

- Strictly controllling physical security, physical access to network, and server hardware.
- Not using VLAN1 as the primary network data VLAN; this is the default VLAN, therefore it is easily compromised.
- Disabling high-risk protocols on any switch or firewall network port that does not require them; for example, protocols such as CDP and PAgP do not need to be enabled on all ports.
- Pruning VLANs not in use; this will prevent unwanted access from a rogue computer on the network.
- Controlling inter-VLAN routing by using firewall policies.

End-to-End Security for Site-to-Site VPNs

Many companies use public clouds to achieve certain cloud benefits or they consolidate IT resources into a private cloud. Both require that companies be able to connect private and public clouds. The secure and logical way to do this is by using virtual private networks (VPNs). These connections are commonly referred to as site-to-site VPNs.

The basic concept of a site-to-site VPN is that it extends a private network across a public network such as the Internet. In the case of cloud computing, a VPN can connect to a remote cloud located in a remote corporate data center, or to a publicly hosted cloud provider such as Rackspace, Amazon, or Softlayer, as shown in Figure 6-5.

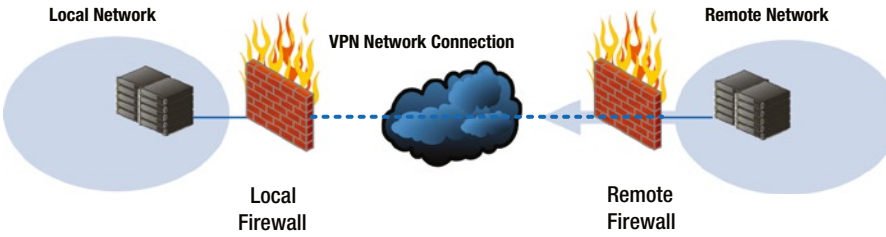

Figure 6-5. *Joining remote network into the local network using a VPN*

A VPN provides a tunnel connection between specified VPN endpoints, usually firewalls. These connections are typically authenticated and then secured using encryption techniques. This prevents networked traffic from being analyzed via sniffing techniques. For example, an attacker could possibly see the traffic at the packet level, but after analyzing it, would only see encrypted traffic.

The legacy methods to establish VPNs were to use hardware-based firewalls or routers. In cloud computing environments, it is now becoming more common to use software-based appliances to establish VPNs, which allows greater flexibility, fine-grained security, and quick configuration and provisioning times, as shown in Figure 6-6.

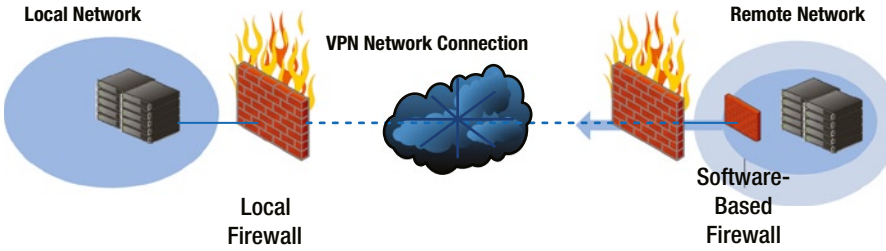

Figure 6-6. *Joining a remote network to the local network using a software VPN*

The two most common site-to-site VPNs used for connecting to remote clouds are IPSEC and SSL/TLSEC. IPSEC is a Layer 3 VPN with an encrypted Layer 3 tunnel between the peers. SSL is a higher layer security protocol than IPSEC, working at the application layer rather than at the network layer.

Site-to-site VPNs were typically built using IPSEC, but now SSL-based VPNs are becoming popular. Major vendors such as Citrix and VMWare provide SSL VPN products to enable remote cloud access. Also, firewall vendors such as Vyatta and Juniper offer software appliances that can be used to enable VPNs and provide a higher level of security through advanced firewall features.

Network security:End-to-End security: Hypervisors and Virtual Machines

One concern within modern data centers is that of securing virtual machines. In public and private clouds, these virtual machines may share the same network and compute resources, not only between company departments but also between separate companies in a public/hybrid cloud environment.

Hypervisor Security

In a cloud, each server has a hypervisor virtualization layer installed, such as Xen, VMWare, KVM, and Hyper-V. As discussed in Chapter 3, an important component for securing a cloud is to establish trust across virtual machines. This is accomplished by using servers that have trusted platform hardware modules that allow the server to verify the boot process of the server's management domain virtual machine. The objective is to protect virtual machines against attacks such as kernel rootkits or viruses. Boot integrity and attestation have been covered in Chapter 3 and Chapter 4.

Another important way to secure the hypervisor is by locking down management access to the hypervisor. A best practice is to reserve a VLAN to isolate access to the management interface. This separates management traffic from data or application traffic.

The same could be said for all the guest virtual machines: traffic is isolated from other guest virtual machines. If one of the guests is compromised by an attacker, it may inject malicious traffic into the network. Inter-VLAN routing should not be performed by the virtual switch in the hypervisor. Best practice is to force traffic up to the firewall and allow the firewall to control inter-VLAN routing. This protects guests from one another in a multitenant cloud, as shown in Figure 6-7.

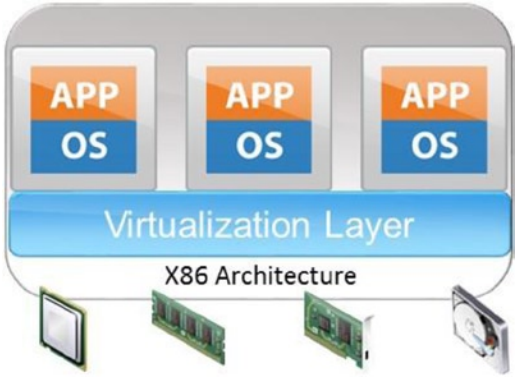

Figure 6-7. *Virtualization layer managing guest virtual machines*

Resources shared by the hypervisor and guests should be removed or restricted. Features such as shared folders can be exploited by attackers, moving from a virtual machine guest to gain access to the hypervisor by placing executable files on the shared resource and then executing them.

Virtual Machine Guest Security

Virtual machine guest security is similar to hardening an operating system. Accounts need to be restricted, and the operating system is maintained up to date and patched. The main concern with virtual machine guests is that virtual machines live in a shared environment. Therefore, extra steps should be taken to protect them from potentially nosy guests. A virtual machine guest, for instance, should restrict traffic from other virtual machine guests and only allow traffic from intended sources. Virtual machine guests should carry internal firewalls configurable to allow only the protocols necessary for the applications installed to function correctly. For example, this includes HTTP or HTTPs traffic from the Internet, SQL traffic to a backend database, and management traffic via SSH from an administrative VLAN.

***Secure Storage*:** Mission-critical applications used in public or hybrid clouds require a higher level of security to comply with corporate security policies or to meet other compliance requirements. For instance, data in shared networked storage environments needs to be encrypted. Users need to know where data is before figuring out how to protect it. Therefore, a complete and accurate inventory of systems, software, and data located in the cloud is necessary at all times. Encrypted data is intrinsically protected, so policies should enforce automatic encryption of data before it is stored or moved to the cloud. In the case of a hybrid cloud, connections between the internal network and the cloud should also be encrypted.

***Virtual Appliances*:** Network security devices such as firewalls, switches, and load balancers at one time could be found only in hardware. Now vendors have started to supply appliances in prepackaged virtual machines. This allows users to spin up

instances of their software when specific capabilities are needed. For example, if a new group of applications is deployed, a new load balancer may need to be created along with it. If a new network segment is created dynamically, a new firewall may need to be created to support that. In the opposite case, if a network segment is deprovisioned, then the firewall could be spun down.

Software-Defined Security in the Cloud

Another concept that has evolved in association with cloud computing is the software-defined networks (SDN). Applications in the cloud can be dynamic in size, location, and lifetime. This puts increased pressure on coming up with the means to secure the cloud in this challenging environment. Software-defined security was conceived to address these concerns.

The term *software defined security* evolved from *software defined networking* (SDN). SDN was conceived to solve similar problems found in dynamic, challenging networks like those in cloud computing. So there is a bit of overlap between the two, since both address matters of security in the network space.

Initially, software-defined networking focused on making the network control plane programmable through application programing interfaces (APIs) and protocols. The concept evolved to meet the needs of a dynamic IT infrastructure. Provisioning storage, virtual machines, switches, load balancers, and firewalls in such environments required APIs so they could be automated through workflows and orchestration engines.

SDN OVERVIEW

SDN is an approach to computer networking in which the control plane for network switches is extracted and centralized on one or more servers. Figure 6-8 illustrates this concept. The data plane is illustrated in the figure by API and switch Silicon, whereas the control plane is illustrated by Network Intelligence and OS. In traditional networking, every switch has both a data plane and a control plane. In SDN, switches only have a data plane and support for communicating with a remote (and centralized) control plane. The original protocol defined for this communication is called OpenFlow, although recently other protocols have been introduced by certain networking vendors.

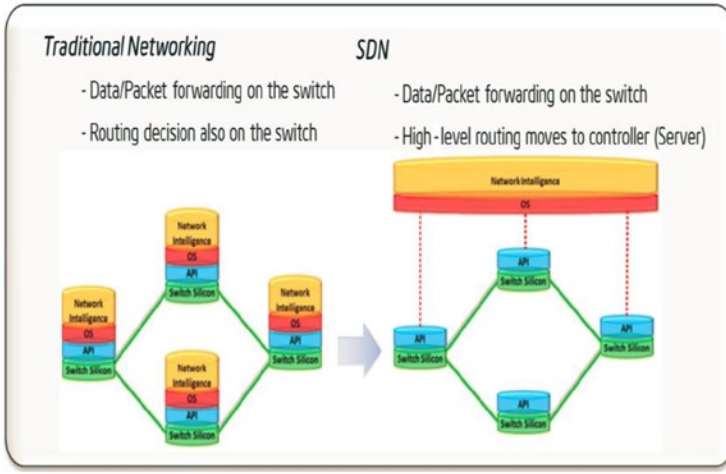

Figure 6-8. *Traditional Networking vs. SDN*

The software representing the centralized control plane is known as the SDN controller and runs on a server platform, illustrated in Figure 6-8.

SDN provides the following advantages:

Unmatched Network Agility: Programmability and automation provide dramatic improvements in service agility and provisioning time.

Choice in Networking Hardware: Standards-based OpenFlow switches provide choice in networking hardware for the first time ever.

Optimized Network Operations: Automation of network provisioning tasks and integration with data center resource orchestration platforms drives dramatic reduction in network operation tasks and requirements.

Centralized view of the network

Figure 6-9 illustrates usage models that have been identified as getting significant benefits from SDN.

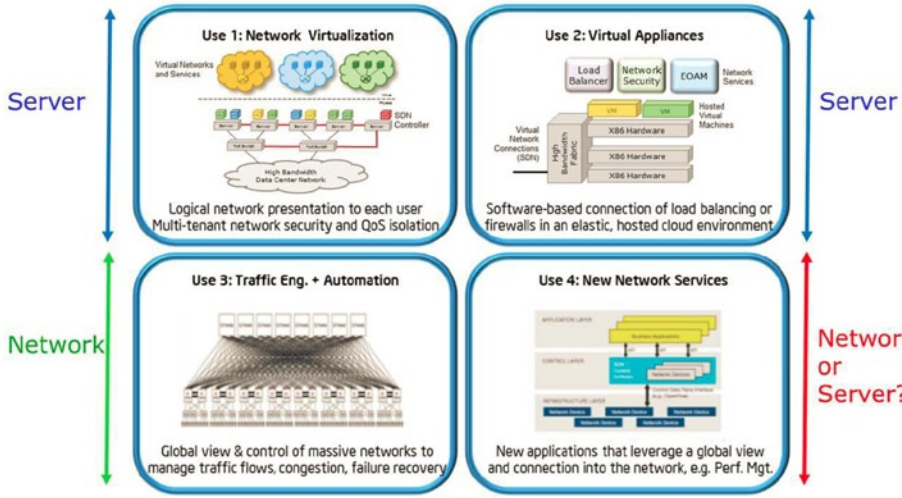

Figure 6-9. *SDN use cases*

The SDN network virtualization usage model provides tenants with their own virtual (and isolated) networks while running on top of a common physical network infrastructure. The virtual appliance usage model enables the instantiation of security on demand in order to fulfill the specific needs of a virtual machines or group of virtual machines.

Specific advantages of these two usage models in cloud multitenant (IaaS) data centers include:

- A VPN would be created support for unrestricted VM migration (i.e., VM migration across subnets)
- Improved visibility (for network management software) of intra-node traffic (i.e., VM to VM running on the same node)
- Improved virtual network management by allowing tenants to manage their virtual networks without interfering with the cloud provider or other tenants
- Improved flexibility to deploy virtual security appliances (e.g., firewalls, intrusion detection/prevention systems, etc.)

Taking advantage of SDN in cloud multi-tenant (IaaS) data centers does not require changing physical network switches. All of the advantages mentioned above can be obtained by adding SDN support to virtual switches (software switches that allow virtual machines to communicate inside and outside the physical server) and putting in place an SDN controller that communicates with the virtual switches and that provides interfaces for a virtualization management infrastructure to create and manage virtual networks.

SDN makes it easier to intercept traffic directed to a virtual machine and redirect it to a security appliance such as a firewall or an intrusion detection and prevention system. Given that trusted compute pools prescribe and enable higher levels of protection for critical workloads, a tenant's security personnel might like:

- Tenant-defined and specified IPS/IDS/firewalls security appliances for their workloads and applications, rather than the generic ones that the Cloud Service Provider supplies.
- Security appliances run on trusted compute pools to ensure integrity, protections, and control policies.

There are two major concepts for security and software-defined networks. The first is APIs, used to make changes to network and hardware elements found in a data center. The second is orchestration, namely taking these APIs and putting them to use in a logical manner.

- *APIs.* There are a number of APIs and software solutions that support the notion of SDN. Vendors such as Juniper, Cisco, and VMWare all supply pre-packaged APIs to enable the control and management of their hardware and software. The APIs can be used by developers to manage, orchestrate, and automate their cloud resources.
- *Orchestration*. Security tasks in the cloud typically require the invocation of a number of APIs to fulfill a set of tasks. For example, when a new virtual machine instance is created, a security policy will need to be provisioned in order to allow traffic to flow to it. The following is an example orchestration initiated after the new virtual machine instance is created:
  - A load balancer may need to be created or updated to accommodate the new virtual machine instance.
  - VLANs may need to be created to allow traffic to the virtual machine.
  - The firewall's rules are updated to regulate traffic to it.
  - Monitoring rules can be added to observe traffic and user access.

Ideally, orchestration should be atomic in the sense of transactions: if the task fails at any point during the orchestration, a smooth rollback of the API executions that did manage to complete in the chain of API invocations would be completed transparently.

All of these concepts and network technology elements play a critical role in real-world cloud computing environments built on cloud management software like OpenStack, Eucalyptus, Amazon AWS, Virtustream xStream, and so on. OpenStack, discussed in the next section, will be introducing a first-order mapping to the network security primitives and components we have discussed in the previous sections.

OpenStack

OpenStack is the leading open-source package for managing cloud environments. Knowledge of the basics of OpenStack provides understanding of what's needed to manage and secure a cloud computing environment.

OpenStack is a Python-based cloud computing management application developed collaboratively by Rackspace and NASA. Later, as the technology grew in popularity, companies such as Dell, Red Hat, HP, Intel, IBM, and Citrix got involved and started contributing to the project. OpenStack is a collection of open-source components delivering a massively scalable cloud operating system. It can be thought of as a service (IaaS) software package designed to manage end-to-end cloud infrastructure.

The management of cloud infrastructure can be quite complicated, since it is composed of a number of different resources: servers, hypervisors, storage, hard drives, network, and racks. OpenStack was designed to manage all these resources in a modular fashion.

OpenStack consists of a set of inter-related projects that address the various resources of a cloud computing platform. Its services are interoperable with existing cloud services like AWS, which heightens its appeal. As of this writing, there are seven projects: Nova, Swift, Glance, Cinder, Neutron, Horizon and Keystone, with a few more in proposal and blueprint development:

- *Nova* provides the ability to provision virtual servers on demand.
- *Swift* is similar to Amazon's S3, a highly scalable and durable object storage system used to store data accessible through RESTful HTTP APIs.
- *Glance Image Service* provides services for discovering, registering, and retrieving virtual machine images.
- *Cinder* provides block storage for virtual environments. This is similar to Amazons EC2's Elastic Block storage, where the block storage volumes are network-attached and can persist independently from the life of an instance.
- *Neutron* provides networking as a service functionality to OpenStack. This involves configuring network components such as virtual switches, firewalls, hardware switches, load balancers, and more.
- *Horizon Dashboard* is the web-based dashboard for exposing the cloud management capabilities of OpenStack.
- *Keystone* provides identity, token, catalog, and policy services for projects in the OpenStack family. For example, before a Glance call is made, authentication is processed by Keystone. Glance depends on Keystone and the OpenStack Identity API to handle authentication of client requests.
- *Ceilometer* was created to allow the metering of cloud environments. Metering includes virtual machine instances, CPU and RAM usage, network data I/O, and block storage hourly usage.

OpenStack Network Security

OpenStack has essential security features. For example, OpenStack's APIs allows exposing firewall, load balancer, switch, and intrusion detection system (IDS) capabilities as infrastructure services. Specifically:

- *LB-aaS* or *load balancing* is an important capability. For example, if an additional virtual machine instance is spun up to meet increased load, then it can be added to an application pool on a load balancer through an API.
- *VPN-aaS* or *VPN* is another popular feature. Picture a new network segment provisioned for a tenant at a remote cloud. A VPN needs to be created after provisioning to enable a secure connection from the tenant's data center to the network segment at the cloud provider.
- *Firewall-aaS* or *firewall* allows tenants to customize firewall rules to meet their application security needs and match corporate security and compliance requirements.
- *VLAN-aaS* or *VLAN* offers tenants the ability to expand their cloud network resources. Often, more IP addresses are needed and logical separation of network resources is required. In this case, a new network segment and VLAN need to be provisioned on demand. VLAN as a service exposes this functionality as an API.

Furthermore, each of these services can be exposed to tenants under a cloud security model. For example, a tenant may be able to create a VPN to its network segment, but not allowed to see VPN resources of other tenants. An administrator may have the ability to see created VPNs, but would be unable to delete it unless special permissions were in place. OpenStack's architecture was designed to provide fine-grained management of cloud resources. It allows cloud administrators and architects to apply role-based controls to network functions and services, as shown in Figure 6-10.

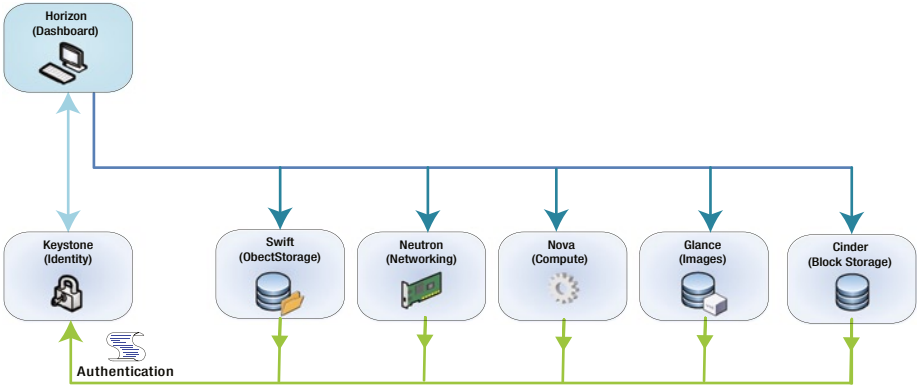

Figure 6-10. *Access to cloud services are managed using roles and privileges*

This gives OpenStack the capability to virtualize network functions. In the case of network security, items such as switch, NAT, DHCP, load balancing, network interface configuration, and firewall security policies can be quickly and securely instantiated.

Network Security Capabilities and Examples

M2Mi Corporation provides cloud network services. The company offers a set of appliances callable by applications like OpenStack through its APIs, providing higher level management, workflow, and analytics tools. The API allows engineers to request specific actions, make changes, or request data without having to have knowledge of vendor-specific capabilities. For example, suppose a new network segment is needed to be provisioned for a new set of applications. Tasks such as segment allocation, DNS provisioning, VLAN provisioning, and network security policy creation are carried out in an orderly manner. The APIs can autodetect device types and perform the above actions as necessary. The API relationships are shown in Figure 6-11.

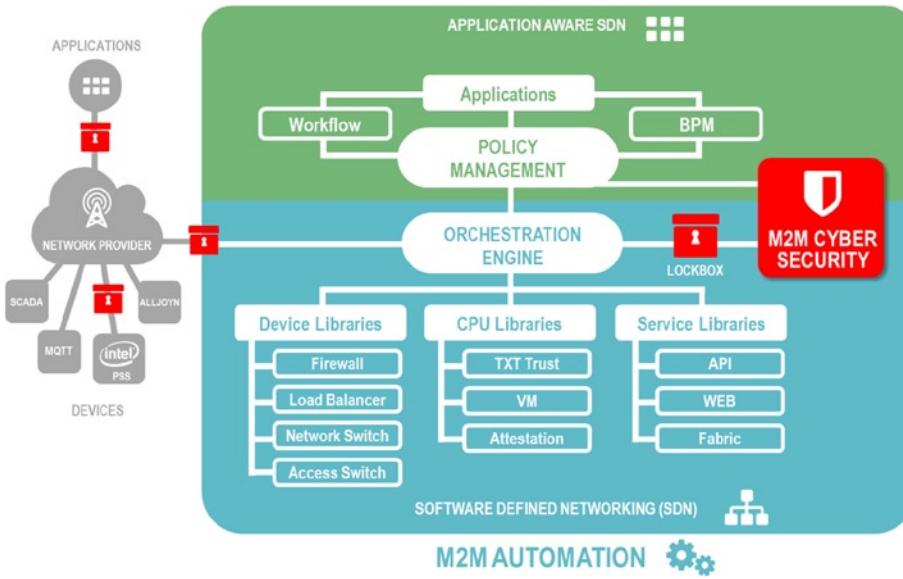

Figure 6-11. *Protecting the cloud using M2M automation from M2Mi*

For example, perhaps a new virtual machine is about to be provisioned and will use VLAN 150. A network administrator typically checks on whether the VLAN already exists on the switch, and if so, on the customers or applications using it. If it isn't there, then the administrator can create it in the switch's VLAN database. The next step enables the VLAN on the physical port connecting the switch to the virtual machine's server. In a data center, this is typically a trunked port, which means the network port can support multiple VLANs with the same physical port. Also, the switch will likely have multiple connections to the physical server using an 802.3ad link aggregated channel. See Figure 6-12.

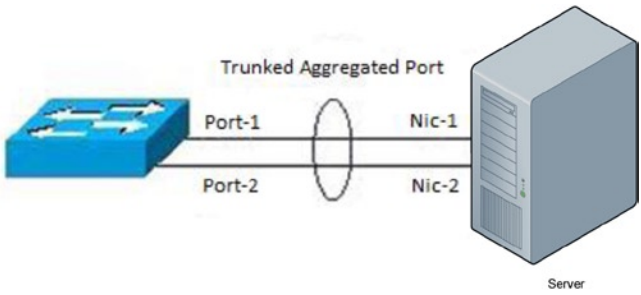

Figure 6-12. *VLAN trunking*

Let's take a look at the commands that would enable the VLAN on the switch. The following commands will illustrate the simplest use case, where there is only one cable connecting the server to the switch. The first command to be sent to a switch will check to see which VLANs have been created on the switch:

```
switch# show vlan brief
VLAN Name                       Status    Ports
---- ------------------------ --------- ----------------------------
                                         Gi1/46, Gi1/48
1    default                    active    Gi1/23, Gi1/24, Gi1/25
137  VLAN0139                   active
140  VLAN0140                   active
141  VLAN0141                   active
142  VLAN0142                   active
```

This command shows that the VLANs 139 through 142 have previously been created. VLAN 143 would need to be added to VLAN database on the switch:

```
switch# configure terminal
switch(config)# vlan 143
switch(config-vlan)# name CustomerA
```

Note that a name can be used for the VLAN. This is used as a tracking mechanism to associate the VLAN with a logical name. Often, data centers will use an application's name or owner as the VLANs name. The next step is to create the VLAN on the port. In this case, the port on the switch that connects to the server is Gi1/5:

```
switch# configure terminal
switch(config)# interface Gi1/5
switch(config-if)# switchport trunk allowed vlan 143
switch(config-if)#
```

The final step is to check the interface to make sure the VLAN was added.

```
Switch# Show interface Gi1/5
Name: Gi1/5
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
   ...
```

Trunking VLANs enabled: 140-143. If adding and removing VLANs is a regular occurrence, then it is desirable to automate the process. The M2Mi APIs provide a VLAN orchestration call that allows all the previous steps to be accomplished in one simple one call:

```
addVLAN("143","Customer A","cisco10.example.com");
```

This call sends the following request to the server:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:jax="http://jaxws.switches.cisco.m2mi.com/">
   <soapenv:Header/>
   <soapenv:Body>
      <jax:VLANOrchestration>
         <!--Optional:-->
         <vlanID>143</vla8nID>
         <!--Optional:-->
         <vlanName>Customer A</vlanName>
         <!--Optional:-->
         <port>Gi1/5</port>
         <!--Optional:-->
         <hostname>cisco10.example.com</hostname>
      </jax:VLANOrchestration>
   </soapenv:Body>
</soapenv:Envelope>
```

The idea is to automate several of the manual steps and remove the element of human error from the configuration. There are other advantages to using APIs. For instance, invocation of the APIs can be limited by users or groups, providing a complete audit trail of all commands that were sent to a device.

Summary

Implementing network security in the cloud requires an in-depth analysis of the hardware and software found in the data center hosting the cloud. There are additional considerations for hybrid cloud or public clouds, with more factors to consider involved in an analysis, such as security issues when traversing the Internet and the quality of the security in the remote data center hosting the cloud.

Security in the cloud is based on best practices evolved over years in order to meet new threats and adapt to new hacking technologies. These best practices can be applied to cloud computing, and a number of companies provide services out of the box to enhance cloud computing security. While many see cloud computing as a technical revolution, the security applied to it is based on hard experience, evolved from known protective measures and standard operating practices. Practices include encrypting data at rest, separation of concerns through delegated administration, application fingerprinting, secure logging, secure backups, auditing, and threat identification.