The Internet Protocol Journal

September 2009 Volume 12, Number 3

A Quarterly Technical Publication for Internet and Intranet Professionals

In This Issue

You can download IPJ back issues and find subscription information at:
www.cisco.com/ipj

ISSN 1944-1134

From The Editor

This journal has covered numerous emerging technologies since we started publishing in June 1998. It would be an interesting excercise to look at which of these technologies have been successfully deployed, which ones have been rejected, and which ones are still emerging or slowly being deployed. In this issue we examine another emerging technology, or perhaps "a new concept" would be a better term, because a collection of new and old technologies are coming together to form what is collectively known as *Cloud Computing*. In a two-part article on cloud computing, T. Sridhar gives an overview of the concepts underlying this area of development. Part 1 of the article is subtitled "Models and Technologies." It will be followed by Part 2: "Infrastructure and Implementation Topics," which will be published in our next issue.

In the last year, I have had one of my credit cards "compromised" (unauthorized charges posted to the account) and subsequently replaced twice. This situation is always annoying and worrisome. Most likely, these breaches resulted from the card information being captured through an online purchase transaction. I am sure I will never know the full story, and luckily the credit card companies are pretty good about detecting fraudulent charges and quickly resolving the matter. When you start thinking about the number of network and server elements involved in a typical e-commerce transaction, it isn't entirely surprising that someone with criminal intentions could exploit a weakness in the overall system. Our second article, by Michael Behringer, explores the topic of "end-to-end security" in more detail.

Those of you who have been subscribers to this journal for several years have probably noticed that your subscription has been "auto-renewed" once a year without requiring any renewal action on your part. Starting with the December 2009 issue, we will no longer extend your subscription when it expires unless you renew it by visiting the IPJ "Subscriber Services" webpage. You will need to use your e-mail address and Subscription ID in order to gain access to your record, where you can renew, update your delivery address, or change delivery method. IPJ is available on paper, as well as online in both HTML and PDF formats. You can also contact us at **ipj@cisco.com** regarding your renewal. The expiration date and Subscription ID are printed on the back of the journal for subscribers in the United States, and on the envelope for our international subscribers. We believe that this new renewal policy will result in fewer undeliverable or unwanted copies being mailed out—a plus for the environment.

—Ole J. Jacobsen, Editor and Publisher
ole@cisco.com

Cloud Computing—A Primer
Part 1: Models and Technologies

by T. Sridhar

Cloud computing is an emerging area that affects IT infrastructure, network services, and applications. Part 1 of this article introduces various aspects of cloud computing, including the rationale, underlying models, and infrastructures. Part 2 will provide more details about some of the specific technologies and scenarios.

The term "cloud computing" has different connotations for IT professionals, depending upon their point of view and often their own products and offerings. As with all emerging areas, real-world deployments and customer success stories will generate a better understanding of the term. This discussion starts with the *National Institute of Standards and Technology* (NIST) definition:

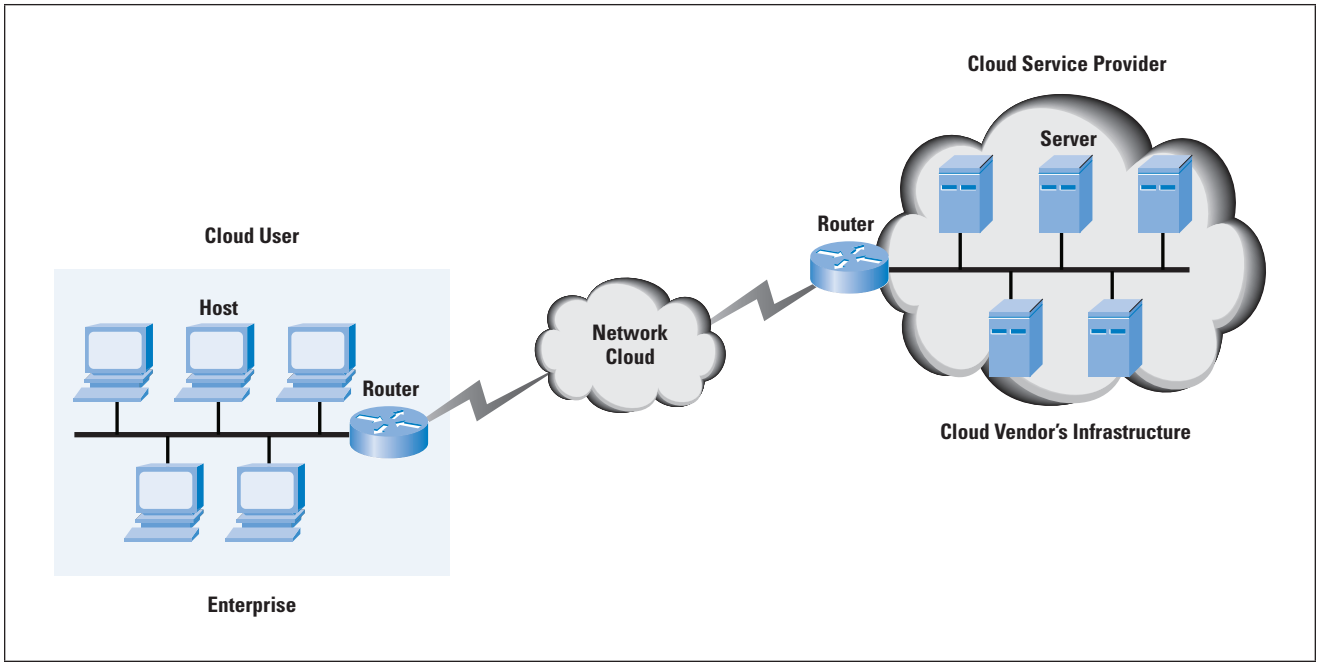
> "Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

The following is a list of characteristics of a cloud-computing environment. Not all characteristics may be present in a specific cloud solution.

- *Elasticity and scalability:* Cloud computing gives you the ability to expand and reduce resources according to your specific service requirement. For example, you may need a large number of server resources for the duration of a specific task. You can then release these server resources after you complete your task.
- *Pay-per-use:* You pay for cloud services only when you use them, either for the short term (for example, for CPU time) or for a longer duration (for example, for cloud-based storage or vault services).
- *On demand:* Because you invoke cloud services only when you need them, they are not permanent parts of your IT infrastructure—a significant advantage for cloud use as opposed to internal IT services. With cloud services there is no need to have dedicated resources waiting to be used, as is the case with internal services.
- *Resiliency:* The resiliency of a cloud service offering can completely isolate the failure of server and storage resources from cloud users. Work is migrated to a different physical resource in the cloud with or without user awareness and intervention.
- *Multitenancy:* Public cloud services providers often can host the cloud services for multiple users within the same infrastructure. Server and storage isolation may be physical or virtual—depending upon the specific user requirements.

- *Workload movement:* This characteristic is related to resiliency and cost considerations. Here, cloud-computing providers can migrate workloads across servers—both inside the data center and across data centers (even in a different geographic area). This migration might be necessitated by cost (less expensive to run a workload in a data center in another country based on time of day or power requirements) or efficiency considerations (for example, network bandwidth). A third reason could be regulatory considerations for certain types of workloads.

Figure 1: Cloud Computing Context



Cloud computing involves shifting the bulk of the costs from *capital expenditures* (CapEx), or buying and installing servers, storage, networking, and related infrastructure) to an *operating expense* (OpEx) model, where you pay for usage of these types of resources. Figure 1 provides a context diagram for the cloud.

How Is Cloud Computing Different from Hosted Services?

From an infrastructure perspective, cloud computing is very similar to *hosted services*—a model established several years ago. In hosted services, servers, storage, and networking infrastructure are shared across multiple tenants and over a remote connection with the ability to scale (although scaling is done manually by calling or e-mailing the hosting provider). Cloud computing is different in that it offers a pay-per-use model and rapid (and automatic) scaling up or down of resources along with workload migration. Interestingly, some analysts group all hosted services under cloud computing for their market numbers.

Virtualization and Its Effect on Cloud Computing

It can be argued to good effect that cloud computing has accelerated because of the popularity and adoption of virtualization, specifically server virtualization. So what is virtualization? Here, virtualization software is used to run multiple *Virtual Machines* (VMs) on a single physical server to provide the same functions as multiple physical machines. Known as a *hypervisor,* the virtualization software performs the abstraction of the hardware to the individual VMs.

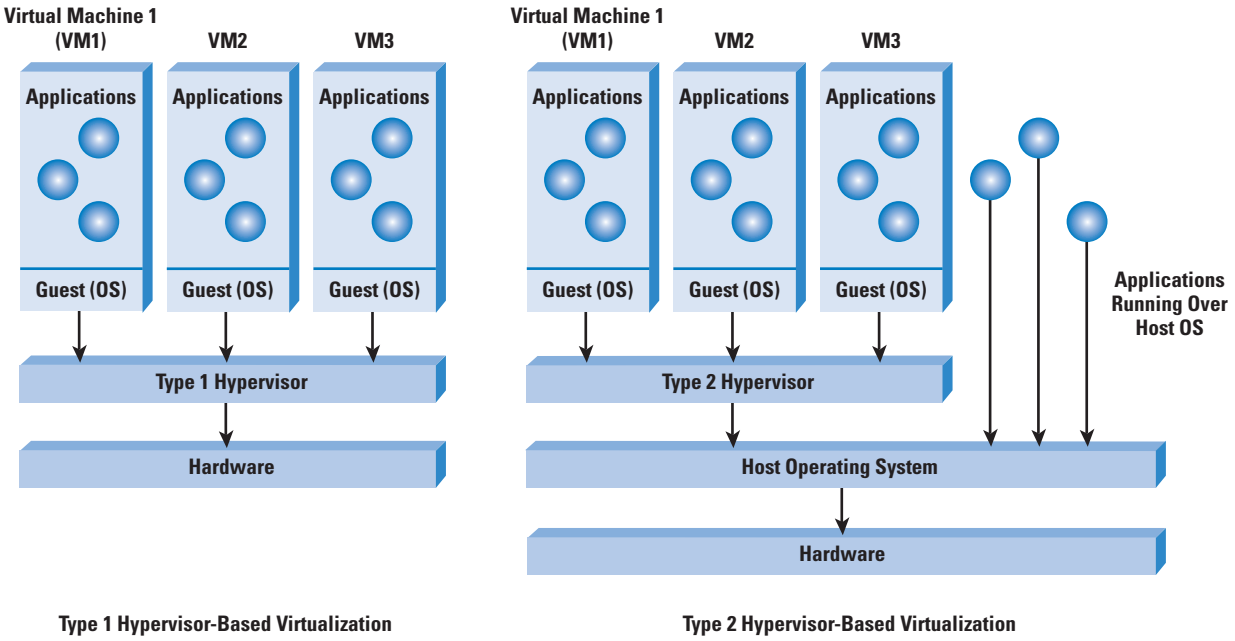
Virtualization is not new—it was first invented and popularized by IBM in the 1960s for running multiple software contexts on its mainframe computers. It regained popularity in the past decade in data centers because of server usage concerns. Data centers and web farms consisted of multiple physical servers. Measurement studies on these server farms noted that individual server usage was often as low as 15 percent for various reasons, including traffic loads and the nature of the applications (available, not always used fully), among others. The consequence of this server sprawl with low usage was large financial outlays for both CapEx and OpEx—extra machines and related power and cooling infrastructure and real estate.

Enter virtualization. A hypervisor is implemented on a server either directly running over the hardware (a *Type 1 hypervisor*) or running over an *operating system* (OS) (a *Type 2 hypervisor*). The hypervisor supports the running of multiple VMs and schedules the VMs along with providing them a unified and consistent access to the CPU, memory, and I/O resources on the physical machine. A VM typically runs an operating system and applications. The applications are not aware that they are running in a virtualized environment, so they do not need to be changed to run in such an environment. Figure 2 depicts these scenarios. The OS inside the VM may be virtualization-aware and require modifications to run over a hypervisor—a scheme known as paravirtualization (as opposed to *full virtualization*).

VM Migration: An Advantage of Virtualization

Some vendors have implemented VM migration in their virtualization solution—a big advantage for application uptime in a data center. What is VM migration? Consider the case of a server with a hypervisor and several VMs, each running an OS and applications. If you need to bring down the server for maintenance (say, adding more memory to the server), you have to shut down the software components and restart them after the maintenance window—significantly affecting application availability. VM migration allows you to move an entire VM (with its contained operating system and applications) from one machine to another and continue operation of the VM on the second machine. This advantage is unique to virtualized environments because you can take down physical servers for maintenance with minimal effect on running applications.

Figure 2: Hypervisors in Virtualization



You can perform this migration after suspending the VM on the source machine, moving its attendant information to the target machine and starting it on the target machine. To lower the downtime, you can perform this migration while the VM is running (hence the name "live migration") and resuming its operation on the target machine after all the state is migrated.

The following are some of the benefits of virtualization in a cloud-computing environment:

- *Elasticity and scalability:* Firing up and shutting down VMs involves less effort as opposed to bringing servers up or down.
- *Workload migration:* Through facilities such as live VM migration, you can carry out workload migration with much less effort as compared to workload migration across physical servers at different locations.
- *Resiliency:* You can isolate physical-server failure from user services through migration of VMs.

It must be clarified that virtualization is not a prerequisite for cloud computing. In fact, there are examples of large cloud service providers using only commodity hardware servers (with no virtualization) to realize their infrastructure. However, virtualization provides a valuable toolkit and enables significant flexibility in cloud-computing deployments.

Major Models in Cloud Computing

This section discusses some popular models of cloud computing that are offered today as services. Although there is broad agreement on these models, there are variations based on specific vendor offerings—not surprising during these early days of cloud computing.

Software as a Service

Consider the case of an enterprise with its set of software licenses for the various applications it uses. These applications could be in human resources, finance, or customer relationship management, to name a few. Instead of obtaining desktop and server licenses for software products it uses, an enterprise can obtain the same functions through a hosted service from a provider through a network connection. The interface to the software is usually through a web browser. This common cloud-computing model is known as *Software as a Service* (SaaS) or a hosted software model; the provider is known as the *SaaS Provider.*

SaaS saves the complexity of software installation, maintenance, upgrades, and patches (for example, for security fixes) for the IT team within the enterprise, because the software is now managed centrally at the SaaS provider's facilities. Also, the SaaS provider can provide this service to multiple customers and enterprises, resulting in a multitenant model. The pricing of such a SaaS service is typically on a per-user basis for a fixed bandwidth and storage. Monitoring application-delivery performance is the responsibility of the SaaS provider. `Salesforce.com` is an example of a SaaS provider. The company was founded to provide hosted software services, unlike some of the software vendors that have hosted versions of their conventional offerings.

Platform as a Service

Unlike the fixed functions offered by SaaS, *Platform as a Service* (PaaS) provides a software platform on which users can build their own applications and host them on the PaaS provider's infrastructure. The software platform is used as a development framework to build, debug, and deploy applications. It often provides middleware-style services such as database and component services for use by applications. PaaS is a true cloud model in that applications do not need to worry about the scalability of the underlying platform (hardware and software). When enterprises write their application to run over the PaaS provider's software platform, the elasticity and scalability is guaranteed transparently by the PaaS platform.

The platforms offered by PaaS vendors like Google (with its *AppEngine*) or `Force.com` (the PaaS offering from `Salesforce.com`) require the applications to follow their own *Application Programming Interface* (API) and be written in a specific language. This situation is likely to change but is a cause for concerns about lock-in. Also, it is not easy to migrate existing applications to a PaaS environment. Consequently, PaaS sees the most success with new applications being developed specifically for the cloud. Monitoring application-delivery performance is the responsibility of the PaaS provider. Pricing for PaaS can be on a per-application developer license and on a hosted-seats basis. Note that PaaS has a greater degree of user control than SaaS.

Infrastructure as a Service

Amazon is arguably the first major proponent of *Infrastructure as a Service* (IaaS) through its *Elastic Computing Cloud* (EC2) service. An IaaS provider offers you "raw" computing, storage, and network infrastructure so that you can load your own software, including operating systems and applications, on to this infrastructure. This scenario is equivalent to a hosting provider provisioning physical servers and storage and letting you install your own OS, web services, and database applications over the provisioned machines. Amazon lets you rent servers with a certain CPU speed, memory, and disk capacity along with the OS and applications that you need to have installed on them (Amazon provides some "canned" software for the OS and applications known as *Amazon Machine Images* [AMIs], so that is one starting point). However, you can also install your own OSs (or no OS) and applications over this server infrastructure.

IaaS offers you the greatest degree of control of the three models. You need to know the resource requirements for your specific application to exploit IaaS well. Scaling and elasticity are your—not the provider's—responsibility. In fact, it is a mini do-it-yourself data center that you have to configure to get the job done. Interestingly, Amazon uses virtualization as a critical underpinning of its EC2 service, so you actually get a VM when you ask for a specific machine configuration, though VMs are not a prerequisite for IaaS. Pricing for the IaaS can be on a usage or subscription basis. CPU time, storage space, and network bandwidth (related to data movement) are some of the resources that can be billed on a usage basis.

In summary, these are three of the more common models for cloud computing. They have variations and add-ons, including *Data Storage as a Service* (providing disk access on the cloud), communications as a service (for example, a universal phone number through the cloud), and so on.

Public, Private, and Internal Clouds

We have focused on cloud service providers whose data centers are external to the users of the service (businesses or individuals). These clouds are known as *public clouds*—both the infrastructure and control of these clouds is with the service provider. A variation on this scenario is the *private cloud*. Here, the cloud provider is responsible only for the infrastructure and not for the control. This setup is equivalent to a section of a shared data center being partitioned for use by a specific customer. Note that the private cloud can offer SaaS, PaaS, or IaaS services, though IaaS might appear to be a more natural fit.

An *internal cloud* is a relatively new term applied to cloud services provided by the IT department of an enterprise from the company's own data centers. This setup might seem counterintuitive at first—why would a company run cloud services for its internal users when public clouds are available? Doesn't this setup negate the advantages of elasticity and scalability by moving this service to inside the enterprise?

It turns out that the internal cloud model is very useful for enterprises. The biggest concerns for enterprises to move to an external cloud provider are security and control. CIOs are naturally cautious about moving their entire application infrastructure and data to an external cloud provider, especially when they have several person-years of investment in their applications and infrastructure as well as elaborate security safeguards around their data. However, the advantages of the cloud—resiliency, scalability, and workload migration—are useful to have in the company's own data centers. IT can use per-usage billing to monitor individual business unit or department usage of the IT resources and charge them back. Controlling server sprawl through virtualization and moving workloads to geographies and locations in the world with lower power and infrastructure costs are of value in a cloud-computing environment. Internal clouds can provide all these benefits.

This classification of clouds as public, private, and internal is not universally accepted. Some researchers see the distinction between private and internal clouds to be a matter of semantics. In fact, the NIST draft definition considers a private cloud to be the same as an internal cloud. However, the concepts are still valid and being realized in service provider and enterprise IT environments today.

When Does Cloud Computing Make Sense?

Outsourcing your entire IT infrastructure to a cloud provider makes sense if your deployment is a "green field" one, especially in the case of a startup. Here, you can focus on your core business without having to set up and provision your IT infrastructure, especially if it primarily involves basic elements such as e-mail, word processing, collaboration tools, and so on. As your company grows, the cloud-provided IT environment can scale along with it.

Another scenario for cloud usage is when an IT department needs to "burst" to access additional IT resources to fulfill a short-term requirement. Examples include testing of an internally developed application to determine scalability, prototyping of "nonstandard" software to evaluate suitability, execution of a one-time task with an exponential demand on IT resources, and so on. The term *cloud bursting* is sometimes used to describe this scenario. The cloud resources may be loosely or tightly coupled with the internal IT resources for the duration of the cloud bursting. In an extremely loosely coupled scenario, only the results of the cloud bursting are provided to the internal IT department. In the tightly coupled scenario, the cloud resources and internal IT resources are working on the same problem and require frequent communication and data sharing.

In some situations cloud computing does not make sense for an enterprise. Regulation and legal considerations may dictate that the enterprise house, secure, and control data in a specific location or geographical area. Access to the data might need to be restricted to a limited set of applications, all of which need to be internal. Another situation where cloud computing is not always the best choice is when application response time is critical. Internal IT departments can plan their server infrastructure and the network infrastructure to accommodate the response-time requirements. Although some cloud providers provide high-bandwidth links and can specify *Service-Level Agreements* (SLAs) (especially in the case of SaaS) for their offerings, companies might be better off keeping such demanding applications in house.

An interesting variation of these scenarios is when companies outsource their web front ends to a cloud provider and keep their application and database servers internal to the enterprise. This setup is useful when the company is ramping up its offerings on the web but is not completely certain about the demand. It can start with a small number of web servers and scale up or down according to the demand. Also, acceleration devices such as *Application Delivery Controllers* (ADCs) can be placed in front of the web servers to ensure performance. These devices provide server load balancing, *Secure Sockets Layer* (SSL) front ends, caching, and compression. The deployment of these devices and the associated front-end infrastructure can be completely transparent to the company; it only needs to focus on the availability and response time of its application behind the web servers.

Cloud Computing Infrastructure

The most significant infrastructure discussion is related to the data center, the interconnection of data centers, and their connectivity to the users (enterprises and consumers) of the cloud service.

A simple view of the cloud data center is that it is similar to a corporate data center but at a different scale because it has to support multiple tenants and provide scalability and elasticity. In addition, the applications hosted in the cloud as well as virtualization (when it is used) also play a part.

A case in point is the *MapReduce* computing paradigm that Google implements to provide some of its services (other companies have their own implementations of MapReduce). Put simply, the MapReduce scheme takes a set of input key-value pairs, processes it, and produces a set of output key-value pairs. To realize the implementation, Google has an infrastructure of commodity servers running Linux interconnected by Ethernet switches. Storage is local through inexpensive *Integrated Drive Electronics* (IDE) disks attached to each server.

Jobs, which consist of a set of tasks, are scheduled and mapped to the available machine set. The scheme is implemented through a *Master* machine and *Worker* machines. The latter are scheduled by the Master to implement Map and Reduce tasks, which themselves operate on chunks of the input data set stored locally. The topology and task distribution among the servers is optimized for the application (MapReduce in this case). Although Google has not made public the details of how the back-end infrastructure is implemented for Google Apps and Gmail, we can assume that the physical and logical organization is optimized for the tasks that need to be carried out, in a manner similar to what is done for MapReduce.

SaaS vendors can partition their cloud data center according to load, tenant, and type of application that they will offer as a service. In some cases they might have to redirect the traffic to a different data center, based on the load in the default data center. IaaS provides the greatest degree of control for the user, as discussed earlier. Even here, the topology and load assignment can be based on the number and type of servers that are allocated.

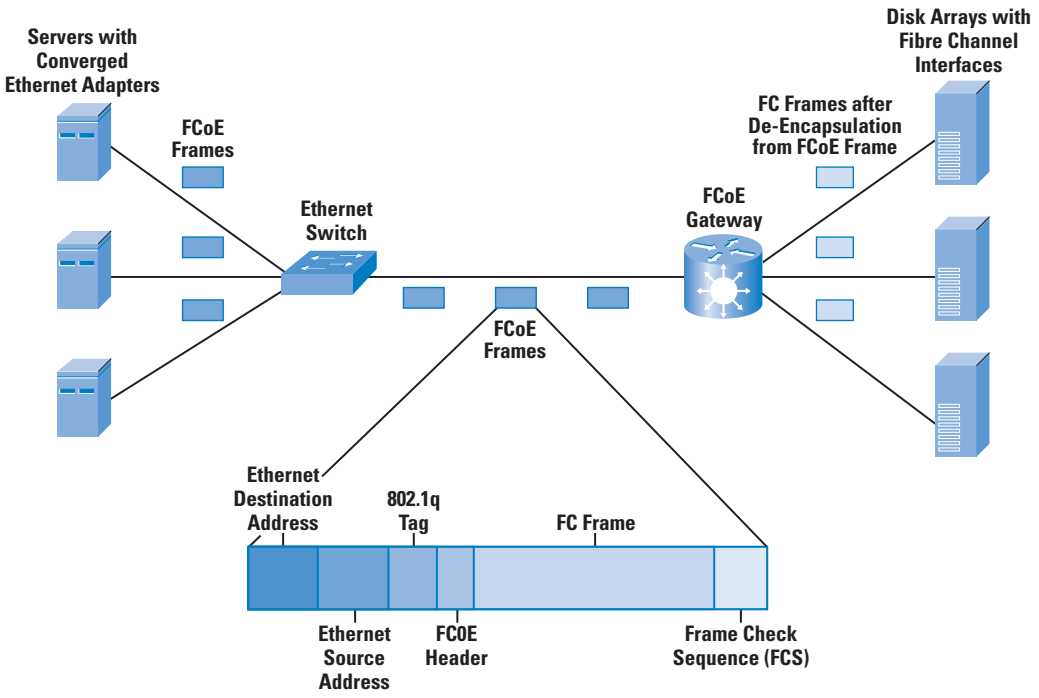
Storage Infrastructure

Storage plays a major part in the data center and for cloud services, especially in environments with virtualization. Storage can be locally attached or accessible through a network—the most popular storage network technologies being *Fibre Channel* and Ethernet. For such network access of storage, servers are equipped with Fibre Channel or Ethernet adapters through which they connect to a Fibre Channel or Ethernet switch. The switch provides the connectivity to storage arrays. Fibre Channel is more popular, though *Network Attached Storage* (NAS) devices with Ethernet interfaces also have a strong presence in the data center. Another Ethernet-based storage option is the *Internet Small Computer System Interface* (iSCSI), which is quite popular among smaller data centers and enterprises because of the cost benefits. This technology involves running the SCSI protocol on a TCP/IP-over-Ethernet connection.

Fibre Channel connections to the storage network necessitate two types of network technologies in the data center: Ethernet for server-to-server and server-to-client connectivity and Fibre Channel for server-to-storage connectivity. A recent initiative in data-center technology is a converged network, which involves the transport of *Fibre Channel over Ethernet* (FCoE). FCoE removes the need for each server to have an Fibre Channel adapter to connect to storage. Instead, Fibre Channel traffic is encapsulated inside an Ethernet frame and sent across to a FCoE gateway that provides Ethernet-to-FCoE termination to connect to Fibre Channel storage arrays (refer to Figure 3). Some storage products provide FCoE functions, so the Ethernet frame can be carried all the way to the storage array. An adapter on the server that provides both "classical" Ethernet and FCoE functions is known as a *Converged Network Adapter* (CNA). Cloud-computing environments can reduce the data-center network complexity and cost through this converged network environment.

Another area in which storage is important is in virtualization and live migration. When a VM migrates to a different physical machine, it is important that the data used by the VM is accessible to both the source and the target machines. Alternatively, if the VM is migrated to a remote data center, the stored data needs to be migrated to the remote data center too. Also, in a virtualized environment, the Fibre Channel, Ethernet, or converged adapter driver should support multiple VMs and interleave its storage traffic to the storage devices. This interleaving is done in consonance with the hypervisor and a designated VM (paravirtualized environments often use this tool), as appropriate.

Figure 3: FCoE in a Cloud Data-Center Environment



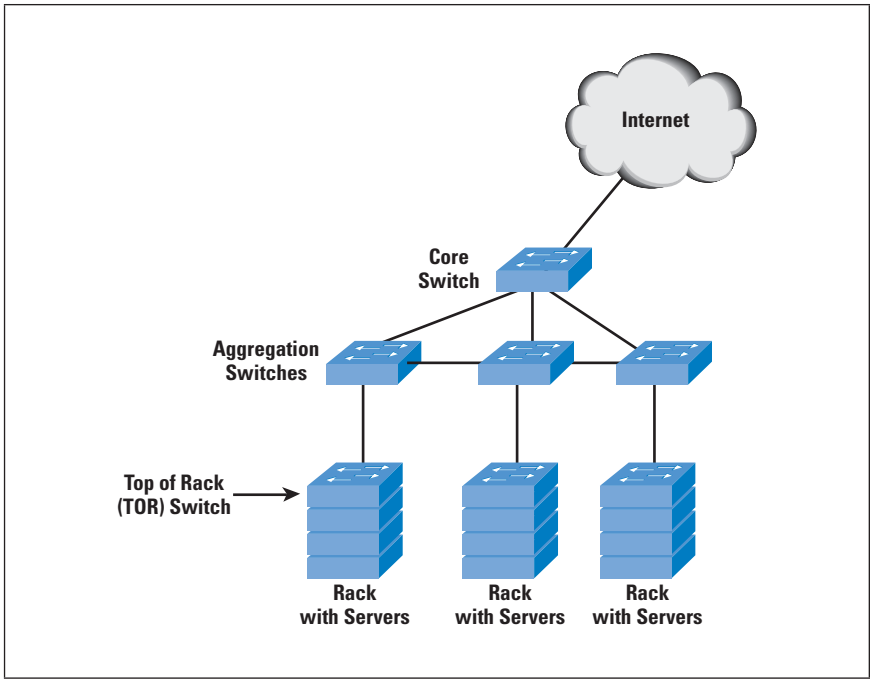
Cloud Computing: Effect on the Network

The previous discussion indicated that the network is a big part of cloud computing. A cloud user connects to the network to access the cloud resources, as indicated earlier in Figure 1. The cloud is accessible through a public network (the Internet) or through a private network (dedicated lines or *Multiprotocol Label Switching* [MPLS] infrastructure, for example). Response-time guarantees depend upon this connectivity. Some cloud vendors offer dedicated links to their data centers and provide appropriate SLAs for uptime or response time and charge for such SLAs. Others might implement a best-effort scheme but provide tools for monitoring and characterizing application performance and response time, so that users can plan their bandwidth needs.

The most significant effect on the network is in the data center, as indicated previously. Let us start with the network architecture or topology. The most common network architecture for enterprises is the three-layer architecture with access, aggregation or distribution, and core switches. The data center requires a slightly different variation to this layering, as proposed by some vendors. The data center consists mainly of servers in racks interconnected through a *Top-of-Rack* (TOR) Ethernet switch which, in turn, connects to an aggregation switch, sometimes known as an *End-of-Rack* (EOR) switch (Figure 4).

The aggregation switch connects to other aggregation switches and through these switches to other servers in the data center. A core switch connects to the various aggregation switches and provides connectivity to the outside world, typically through Layer 3 (IP). It can be argued that most of intra-data center traffic traverses only the TOR and the aggregation switches. Hence the links between these switches and the bandwidth of those links need to account for the traffic patterns. Some vendors have proposed a fat-tree or a leaf-spine topology to address this anomaly, though this is not the only way to design the data-center network. Incidentally, the fat-tree topology is not new—it has been used in *Infiniband* networks in the data center.

Figure 4: Example Data-Center Switch Network Architecture



The presence of virtualized servers adds an extra dimension. Network connections to physical servers will need to involve "fatter pipes" because traffic for multiple VMs will be multiplexed onto the same physical Ethernet connection. This result is to be expected because you have effectively collapsed multiple physical servers into a single physical server with VMs. It is quite common to have servers with 10-Gbps Ethernet cards in this scenario.

New Protocols for Data-Center Networking

Numerous initiatives and standards bodies are addressing the standards related to cloud computing. From the networking side, the IEEE is working on new protocols and the enhancement of existing protocols for data centers. These enhancements are particularly useful in data centers with converged networks—the area is often known as *Convergence Enhanced Ethernet* (CEE).

A previous section indicated the importance of FCoE for converged storage network environments. The IEEE is working to enable FCoE guarantees (because Fibre Channel is a reliable protocol as compared to best-effort Ethernet) through an Ethernet link in what is known as "Lossless Ethernet." FCoE is enabled through a *Priority Flow Control* (PFC) mechanism in the 802.1Qbb activities in the IEEE. In addition, draft IEEE 802.1Qau provides end-to-end congestion notification through a signaling mechanism propagating up to the ingress port, that is, the port connected to the server *Network Interface Card* (NIC). This feature is useful in a data-center topology.

A third draft IEEE 802.1aq defines shortest-path bridging. This work is similar to the work being done in the IETF TRILL *(Transparent Interconnect of Lots of Links)* working group. The key motivation behind this work is the relatively flat nature of the data-center topology and the requirement to forward packets across the shortest path between the endpoints (servers) to reduce latency, rather than a root bridge or priority mechanism normally used in the *Spanning Tree Protocol* (STP). The shortest-path bridging initiative in IEEE 802.1aq is an incremental advance to the *Multiple Spanning Tree Protocol* (MSTP), which uses the *Intermediate System-to-Intermediate System* (IS-IS) link-state protocol to share learned topologies between switches and to determine the shortest path between endpoints.

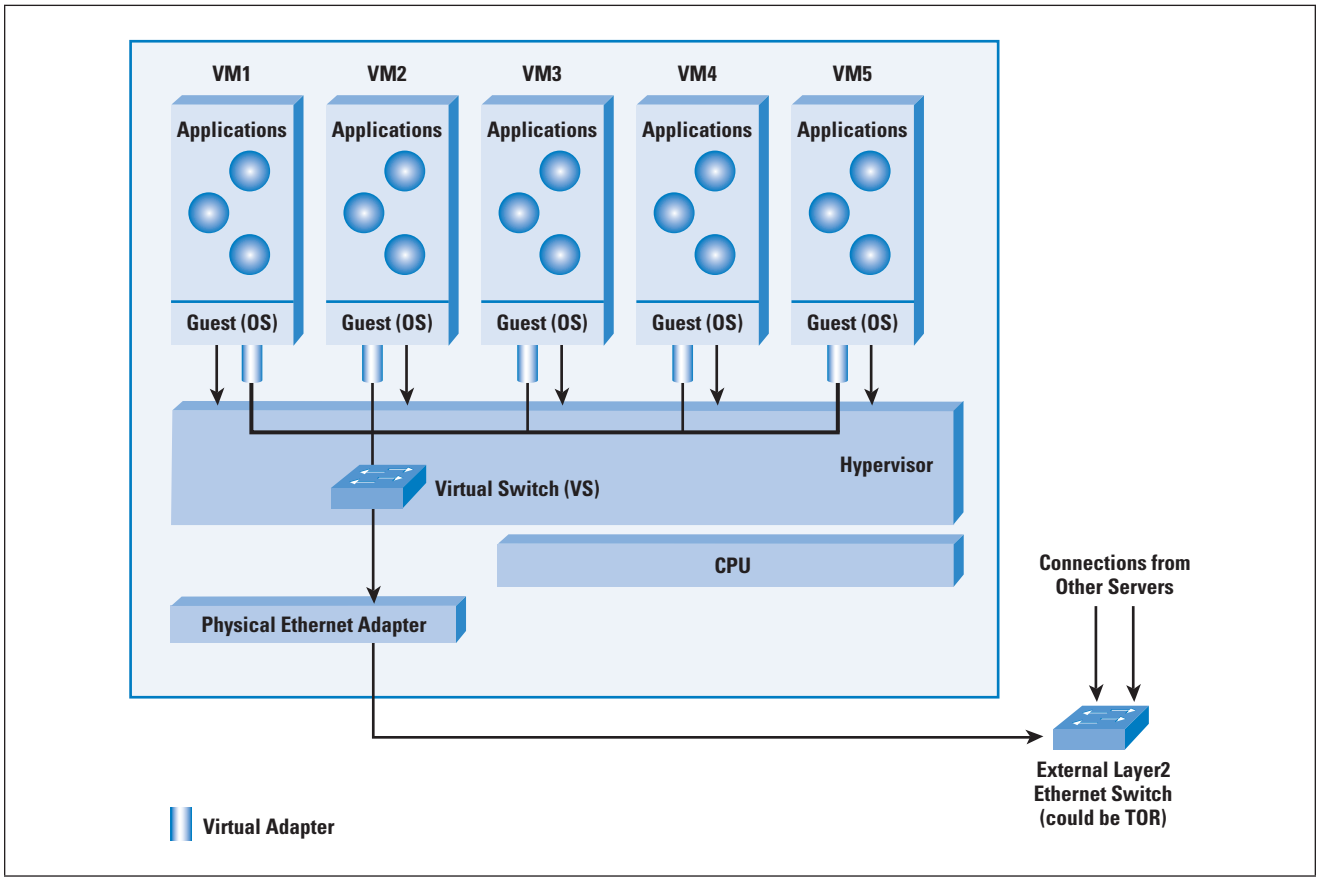
The fourth draft 802.1Qaz is also known as *Enhanced Transmission Selection* (ETS). It allows lower-priority traffic to burst and use the unused bandwidth from the higher-priority traffic queues, thus providing greater flexibility.

Virtualized Network Equipment Functions

Though cloud computing does not depend upon virtualization, several cloud infrastructures are built with virtualized servers. In an environment with physical servers, switches are used to connect servers to other servers. Firewalls and application-delivery controllers are other types of equipment that you can use in a data center on the connection to external clients. With a virtualized environment, you can move some or all of these functions to reside inside a server.

Consider the case of the software-based *Virtual Switch* as shown in Figure 5. You can use the Virtual Switch to switch between VMs inside the same physical server and aggregate the traffic for connection to the external switch. The Virtual Switch is often implemented as a plug-in to the hypervisor. The VMs have virtual Ethernet adapters that connect to the Virtual Switch, which in turn connects to the physical Ethernet adapter on the server and to the external Ethernet switch. To the network manager, the virtual switch can appear as a part of the network. Unlike physical switches, the Virtual Switch does not necessarily have to run network protocols for its operation, nor does it need to treat all its ports the same because it knows that some of them are connected to virtual Ethernet ports (for example, it can avoid destination address learning on the ports connected to the VMs). It can function through appropriate configuration from an external management entity.

Figure 5: Virtual Ethernet Switch in a Virtualized Server Environment



It is possible to implement a virtualized firewall as a VM instead of as a plug-in to the hypervisor. These VMs are self-contained, with an operating system along with the firewall software. The complete package is known as a *firewall virtual appliance*. These VMs can be loaded and configured so that network packets destined for any of the VMs pass through the firewall VM, where they are validated before being passed to the other VMs. Another use of the firewall VM is as a front end to the physical servers in the data center. The disadvantage of a virtual appliance is the performance hit due to its implementation as a software function in a virtualized environment.

Management

Management has several facets in a cloud-computing environment: billing, application-response monitoring, configuring network resources (virtual and physical), and workload migration. In a private cloud or tightly coupled environment, management of the applications may have to be shared between the internal cloud and the private cloud.

You can manage cloud-computing environments in several ways, depending upon the specific area. You can manage the network equipment (physical and virtual) through the *Simple Network Management Protocol* (SNMP) and a network management console. In a virtualized environment, the virtualization vendor often offers a framework to manage and monitor VMs, so this is another part of the equation. Several vendors offer products to act as management front ends for public clouds; for example, Amazon, whose products act as brokers and management consoles for your application deployed over the Amazon cloud offering.

It is clear that this area of management for cloud computing is still evolving and needs to be tied together for a unified management view.

Cloud Computing: Common Myths

Thus far, we have considered the important technologies, terminology, and developments in cloud computing. This section outlines some common myths about cloud computing.

- *Myth: Cloud computing should satisfy all the requirements specified: scalability, on demand, pay per use, resilience, multitenancy, and workload migration.*

  In fact, cloud-computing deployments seldom satisfy all the requirements. Depending upon the type of service offered (SaaS, IaaS, or PaaS), the service can satisfy specific subsets of these requirements. There is, however, value in trying to satisfy most of these requirements when you are building a cloud service.

- *Myth: Cloud computing is useful only if you are outsourcing your IT functions to an external service provider.*

  Not true. You can use cloud computing in your own IT department for on-demand, scalable, and pay-per-use deployments. Several vendors offer software tools that you can use to build clouds within your enterprise's own data center.

- *Myth: Cloud computing requires virtualization.*

  Although virtualization brings some benefits to cloud computing, including aspects such as efficient use of servers and workload migration, it is not a requirement for cloud computing. However, virtualization is likely to see increased usage in cloud deployments.

- *Myth: Cloud computing requires you to expose your data to the outside world.*

  With internal clouds you will never need to expose your data to the outside world. If data security and privacy are concerns, you can develop a cloud model where web front ends are in the cloud and back-end data always resides in your company's premises.

- *Myth: Converged networks are essential to cloud computing.*

  Although converged networks (with FCoE, for example) have benefits and will see increased adoption in data centers in the future, cloud computing is possible without converged networks. In fact, some cloud vendors use only Fibre Channel for all their storage needs today. Use of converged networks in the future will result in cost efficiencies, but it is not a requirement today.

Cloud Computing: Gaps and Concerns

Cloud-computing technology is still evolving. Various companies, standards bodies, and alliances are addressing several remaining gaps and concerns. Some of these concerns follow:

- *Security:* Security is a significant concern for enterprise IT managers when they consider using a cloud service provider. Physical security through isolation is a critical requirement for private clouds, but not all cloud users need this level of investment. For those users, the cloud provider must guarantee data isolation and application security (and availability) through isolation across multiple tenants. In addition, authentication and authorization of cloud users and encryption of the "network pipe" from the cloud user to the service provider application are other factors to be considered.
- *Network concerns:* When cloud bursting is involved, should the servers in the cloud be on the same Layer 2 network as the servers in the enterprise? Or, should a Layer 3 topology be involved because the cloud servers are on a network outside the enterprise? In addition, how would this work across multiple cloud data centers?
- *Cloud-to-cloud and Federation concerns:* Consider a case where an enterprise uses two separate cloud service providers. Compute and storage resource sharing along with common authentication (or migration of authentication information) are some of the problems with having the clouds "interoperate." For virtualized cloud services, VM migration is another factor to be considered in federation.
- *Legal and regulatory concerns:* These factors become important especially in those cases involving storing data in the cloud. It could be that the laws governing the data are not the laws of the jurisdiction where the company is located.

Conclusion

This article introduced the still-evolving area of cloud computing, including the technologies and some deployment concerns. Definitions and standardization in this area are a work in progress, but there is clear value in cloud computing as a solution for several IT requirements. In Part 2 we will provide a more detailed look at some of the technologies and scenarios for cloud computing.

For Further Reading

[1] Draft NIST Working Definition of Cloud Computing,
http://csrc.nist.gov/groups/SNS/cloud-computing/index.html

[2] "Identifying Applications for Public and Private Clouds," Tom Nolle, Searchcloudcomputing,
http://searchcloudcomputing.techtarget.com/tip/0,289483,sid201_gci1358701,00.html?track=NL-1329&ad=710605&asrc=EM_NLT_7835341&uid=8788654

[3] "The Wisdom of Clouds," James Urquhart's blog on Cloud Computing,
http://news.cnet.com/the-wisdom-of-clouds/

[4] "Virtualization – State of the Art," SCOPE Alliance,
http://www.scope-alliance.org/sites/default/files/documents/SCOPE-Virtualization-StateofTheArt-Version-1.0.pdf

[5] "Live Migration of Virtual Machines," Clark, et al.,
http://www.cl.cam.ac.uk/research/srg/netos/papers/2005-migration-nsdi-pre.pdf

[6] "MapReduce: Simplified Data Processing on Large Clusters," Dean & Ghemawat,
http://labs.google.com/papers/mapreduce.html

[7] "Cloud Computing Drives New Networking Requirements," *The Lippis Report,* 120,
http://lippisreport.com/2009/02/lippis-report-120-cloud-computing-drives-new-networking-requirements/

[8] "A New Approach to Network Design When You Are in the Cloud," *The Lippis Report,* 121,
http://lippisreport.com/2009/03/a-new-approach-to-network-design-in-the-cloud/

[9] "Unified Fabric Options Are Finally Here," *The Lippis Report,* 126,
http://lippisreport.com/2009/05/lippis-report-126-unified-fabric-options-are-finally-here/

[10] "Virtualization with Hyper-V," Microsoft,
http://www.microsoft.com/windowsserver2008/en/us/hyperv-overview.aspx

[11] "Citrix XenServer," Citrix,
http://www.citrix.com/English/ps2/products/feature.asp?contentID=1686939

[12] "VMware Virtual Networking Concepts," VMware,
http://www.vmware.com/files/pdf/virtual_networking_concepts.pdf

[13] "Cisco Nexus 1000v Virtual Ethernet Switch," Cisco Systems,
http://www.cisco.com/en/US/prod/collateral/switches/ps9441/ps9902/data_sheet_c78-492971.html

[14] "Application Delivery Challenge," Layland Consulting,
http://www.edge-delivery.org/dl/whitepapers/Application_Delivery_Challenge.pdf

[15] "Cloud Networking: Design Patterns for 'Cloud-Centric' Application Environments,"
http://www.aristanetworks.com/en/CloudCentricDesignPatterns.pdf

[16] IEEE 802.1Qaz – Enhanced Transmission Selection,
http://www.ieee802.org/1/pages/802.1az.html

[17] IEEE 802.1Qau – Congestion Notification,
http://www.ieee802.org/1/pages/802.1au.html

[18] IEEE 802.1Qbb – Priority Flow Control,
http://www.ieee802.org/1/pages/802.1bb.html

[19] IEEE 802.1aq - Shortest Path Bridging,
http://www.ieee802.org/1/pages/802.1aq.html

[20] IETF Transparent Interconnection of Lots of Links (trill) Working Group,
http://www.ietf.org/dyn/wg/charter/trill-charter.html

T. SRIDHAR received his BE in Electronics and Communications Engineering from the College of Engineering, Guindy, Anna University, Madras, India, and his Master of Science in Electrical and Computer Engineering from the University of Texas at Austin. He can be reached at **TSridhar@leitnet.com**

Why End-to-End Security Is Necessary But Not Sufficient

by Michael H. Behringer, Cisco Systems

End-to-end security relies on protocols and mechanisms that are implemented exclusively on the endpoints of a connection. The most typical example is an HTTPS connection (based, for example, on *Transport Layer Security* (TLS)[1]) to a web server; *IP Security* (IPsec)[2] can also be used for end-to-end security, as was initially proposed as a default connection mechanism for IPv6.

There is a perception that end-to-end security is sufficient as a security solution, and that network-based security is obsolete in the presence of end-to-end security. This article outlines why in practice end-to-end security alone is not sufficient, and why network-based security is also required.

Defining "End"

The traditional definition of an endpoint is a client or server. In this definition end-to-end security starts on the client and ends on the server. Given the multitude of applications running in parallel on an operating system, and given increasing virtualization, this definition is usually no longer precise enough. The operating system can establish a security association on either the session or application level. It can also be terminated on a front end, on behalf of numerous servers, as is the case in many TLS[1] deployments.

Because the main goal of this article is to understand why the network has a role to play in security, the precise definition of an endpoint is not relevant here. Abstractly seen, an endpoint is an entity that communicates over a network with another entity. This definition, albeit vague, is sufficient for the discussion at hand.

End-to-End Security Is Fundamental

Security on the endpoints (client-server, or client-client for peer-to-peer) is an absolute requirement for secure communications. Such a solution contains the following components:

- *Identity:* This component encompasses known and verifiable entity identities on both ends; note that an identity can be temporary for a connection. For example, a user often is identified by username and password, whereas a server may be identified through a server certificate.
- *Protocols* (for example, TLS [1] and IPsec [2]): Protocols are used to dynamically negotiate session keys, and to provide the required security functions (for example, encryption and integrity verification) for a connection. Protocols use algorithms to implement these functions.

- *Algorithms* (for example, *Advanced Encryption Standard* [AES][3], *Triple Digital Encryption Standard* [3DES][4], and *Secure Hash Algorithm* [SHA-1][5]): These algorithms use the previously mentioned session keys to protect data in transit, for example through encryption or integrity checks.
- *Secure implementation:* The endpoint (client or server) that runs one of these protocols mentioned previously must be free of bugs that could compromise security. Web browser security is relevant here. Also malware can compromise security, for example by logging key strokes on a PC.
- *Secure operation:* Users and operators have to understand the security mechanisms, and how to deal with exceptions. For example, web browsers warn about invalid server certificates, but users can override the warning and still make the connection. This concern is a nontechnical one, but is of critical concern today.

For full end-to-end security, all of these components must be secure. In networks with end-to-end security, both ends can typically (depending on the protocols and algorithms used) rely on the fact that their communication is not visible to anyone else, and that no one else can modify the data in transit. End-to-end security is used successfully today, for example, in online banking applications. Correct and complete end-to-end security is required; without it, many applications such as online banking would not be possible.

However, a single security problem in any of the components can compromise the overall security for a connection. Today, most critical are implementation problems on endpoints, as well as human errors, specifically in handling exception cases.

Practical Shortcomings of End-to-End Security

Solutions that rely exclusively on end-to-end security have many potential problems, which fall into two broad categories: those that affect the end user and those that affect the network operator (the service provider, or the enterprise network operator, for example).

The End-User View

As reports on online crime and fraud demonstrate very clearly, even in the perceived presence of end-to-end security it is difficult to ensure that none of the components mentioned previously is "broken." Although protocols and algorithms in use tend to be secure and reliable, the main problems lie in the two main areas of endpoint security (secure implementation component) and lack of user education (secure operation component).

Endpoint security concerns include the presence of malware, as well as bugs in software. Even security professionals have difficulty determining whether a PC contains malware. Such malware can control the connection before it is secured, thereby achieving the ability to see the data, as well as potentially change it in real time. Although endpoint security software such as antivirus solutions as well as zero-day prevention solutions provides good security, they are not always installed, and antivirus software is often not up-to-date. Users also can temporarily disable the solutions. Therefore, the presence of malware remains a security concern. Bugs in software are also relevant, for example in the web browser or the operating system.

The lack of user education is the other important concern on the endpoint: Users must know how to identify a secured connection, for example by the little padlock in a web browser (although not even this security mechanism is completely secure). They must also know how to deal with exceptions such as expired or invalid certificates. Most average users do not entirely understand all these details, leading to breaches of security.

The Network Operator View

In the early days of IPv6 it was postulated that the protocol would come with IPsec end-to-end security built in and always "on," thereby eliminating all security problems. This assumption turned out to be wrong, because many problems remain on the network side—for example, general problems with end-to-end security—and they apply to all variants, such as IPsec, TLS, or *Secure Sockets Layer* (SSL).

Today, most enterprise network operators as well as service providers are skeptical about the ubiquitous use of end-to-end security solutions. The fundamental concern is that the endpoints generally cannot be trusted. The network operator, whether enterprise, university, or service provider, has an obligation to enforce certain policies on the endpoint, for example, to ensure that it does not spread worms, send spam mail, or attack servers. If, however, network operators cannot "see" the traffic of an endpoint because it is end-to-end secured, then they cannot comply with their obligations to control the endpoints.

From a network operator's perspective it is therefore not generally desirable to use end-to-end security for all communications, but only for those that really need it.

Why Network-Based Security Is Essential

There are many examples where network-based security is essential, and where end-to-end security solutions not only do not help, but may actually present an additional problem. In all those cases it is essential to have strong network-based security solutions in place. Some examples explain this in more detail.

The Service Provider with DSL Customers

A service provider with DSL customers needs to control its users' traffic in various ways. However, the provider has no control over the endpoints, because those are the customers' property. Because they also cannot force their customers to use appropriate security software, there is always a certain percentage of infected PCs on any given service provider's network. Critical service provider concerns follow:

- *Control of PCs infected with malware:* Such PCs (also referred to as "bots" or "zombies") can infect other PCs and participate in illegal activities, such as spam mail, click fraud[12], *Denial-of-Service* (DoS) attacks, etc. There is a strong, often legal requirement for providers to identify such infected PCs, to isolate them, and to alert their owners and help them to "disinfect" the PC. Network-based security mechanisms are required, essentially because security on the endpoint has failed.
- *Attacks from the users:* Even in the absence of malware, a service provider's user can participate in illegal activities, such as DoS attacks, or intrusions on web servers or routers. Network-based methods are required to detect such attempts, beginning with simple forms such as IP spoofing [6], and to prevent or block them. One example is network-based solutions against DoS attacks[7,8].
- *Control of bandwidth:* Many service providers need to enforce bandwidth limits on some applications or users because they violate service agreements. Also here, applications are necessary to control the PCs, and to limit their usage of the service to remain within contracted boundaries. Service providers today employ a large number of network-based security mechanisms, ranging from visibility solutions to enforcement of certain policies. Endpoint security does not solve these problems, because the PC is not under control of the service provider, and is typically untrusted.
- *Services:* Service providers also try to differentiate themselves from their competition by offering managed services, for example managed security services[9]. Those services are also network-based, and they complement endpoint security solutions that their customers use.

The Service Provider with Customers Under Attack

Service providers may also be required to help their customers when they are under attack. DoS attacks illustrate why endpoint security may not be sufficient, and network-based security is required. Under a DoS attack, a web server, for example, may receive more traffic than it can handle. Such attacks can also overload network resources, such as subscriber lines or routers; therefore, endpoint security is not able to solve such attacks. Massive overprovisioning would be the only way to handle DoS attacks, but this approach is commercially not generally feasible. Network-based solutions based on flow analysis and selective discard of flows are required to help in such situations.

The Enterprise Network

At first glance it seems that enterprises should have full control over the PCs in the enterprise. In such a case, it would be possible to rely completely on end-to-end security. However, this assumption is unrealistic. Numerous current shortcomings make this approach impractical today:

- Enterprise PCs can also get infected with malware, leading to the same problem as for service providers described previously: the need to monitor and control the behavior of a PC in the network. Solutions to control endpoints are themselves network-based; for example, network endpoint assessment[10] and user authentication (802.1x)[11].
- Attacks from users, or against services within the enterprise, also exist in an enterprise environment, as explained previously for service providers. Solutions are network-based.
- The enforcement of *Quality of Service* (QoS) is also a security concern: Users could wrongly classify all their traffic as "high-priority." In the absence of full application control on the PC (which is impractical today), the network needs to control flows from the PC, and potentially enforce a QoS policy. If all flows were encrypted end-to-end, this control would be "blind," probably leading to undesired results. Network security mechanisms are required to control the QoS policy.
- Scale: In an enterprise with several offices that are connected over an untrusted network (for example, the Internet), it may be impractical today to roll out full end-to-end security across the entire enterprise. The currently used approach in most enterprises is to connect the offices with IPsec gateways, and leave traffic within an office in the clear. This scenario increases manageability and scalability of the network. Again, this solution is network-based security solution.
- Although PCs can theoretically be equipped with IPsec (for example) for all communications, many end devices in an enterprise do not support the security mechanisms required. Printers, faxes, and scanners are examples. Full end-to-end security, however, would require all endpoints to support a common mechanism, such as IPsec or TLS. Until all such devices have this support, network-based mechanisms are required to secure communications with them.

Summary

End-to-end security protocols and solutions are an essential cornerstone in network security. We cannot live without them. However, it is unrealistic in today's networks to assume that end-to-end security solutions alone will suffice. The fundamental underlying problem is that typically the network operator, where a PC is attached, has a need and often an obligation to monitor the behavior of the endpoint, and to control malicious activities emerging from that PC. All solutions to control endpoints, however, are by definition network-based. Therefore, network-based security mechanisms are also an essential component of overall network security: Overall security requires both endpoint security and network-based security.

References

[1] T. Dierks, et al., "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246, August 2008.

[2] S. Kent, et al., "Security Architecture for the Internet Protocol," RFC 4301, December 2005.

[3] Joan Daemen and Vincent Rijmen, *The Design of Rijndael: AES—The Advanced Encryption Standard,* Springer-Verlag, 2002. ISBN 3-540-42580-2.

[4] ANSI X9.52:1998, "Triple Data Encryption Algorithm Modes of Operation," July 1998.

[5] FIPS 180-2, "Secure Hash Standard (SHS)," February 2004.

[6] F. Ali, "IP Spoofing," *The Internet Protocol Journal,* Volume 10, No. 4, December 2007.

[7] W. Eddy, "Defenses Against TCP SYN Flooding Attacks," *The Internet Protocol Journal,* Volume 9, No. 4, December 2006.,

[8] C. Patrikakis, et al., "Distributed Denial of Service Attacks," *The Internet Protocol Journal,* Volume 7, No. 4, December 2004.

[9] K. Trivedi and D. Holloway, "Secure Multivendor Networks," *The Internet Protocol Journal,* Volume 10, No. 3, September 2007.

[10] P. Sangster, et al., "Network Endpoint Assessment (NEA): Overview and Requirements," RFC 5209, June 2008.

[11] IEEE 802.1X "Port-Based Network Access Control," `http://www.ieee802.org/1/pages/802.1x.html`

[12] According to Wikipedia: "Click fraud is a type of Internet crime that occurs in pay per click online advertising when a person, automated script or computer program imitates a legitimate user of a web browser clicking on an ad, for the purpose of generating a charge per click without having actual interest in the target of the ad's link. Click fraud is the subject of some controversy and increasing litigation due to the advertising networks being a key beneficiary of the fraud.

Use of a computer to commit this type of Internet fraud is a felony in many jurisdictions, for example, as covered by *Penal Code 502* in California, USA, and the *Computer Misuse Act 1990* in the United Kingdom. There have been arrests relating to click fraud with regard to malicious clicking in order to deplete a competitor's advertising budget."
http://en.wikipedia.org/wiki/Click_fraud

MICHAEL H. BEHRINGER works at Cisco Systems as a distinguished engineer, where he focuses on core security problems, such as MPLS security, multicast security, and Denial-of-Service attack prevention. Michael holds a diploma in computer science from the Technical University of Munich. He is an active member of the IETF, and has published several papers, RFCs, and a book about MPLS VPN security. E-mail: **mbehring@cisco.com**

Letter to the Editor

End of Eternity

Dear Ole,

In their "The End of Eternity" articles, (IPJ Volume 11, No. 4 and Volume 12, No. 1) Niall Murphy and David Wilson provide a detailed and compelling description of the lasting harm that could result from the exhaustion of unallocated IPv4 addresses—harm to Internet users and aspiring new entrants, to technical-coordination and fault-management mechanisms, and to the likely irreplaceable cooperative decision-making and consensus-development mechanisms that distinguish the Internet from every other important transnational sphere of activity in human history. Thankfully, the authors foresee a potential happy ending—or at least yet another chapter in the story—in "an IPv6 Internet, or at least enough of one to keep off address scarcity for a workable subset of the industry."

However, having foreshadowed how they expect the IP addressing cliffhanger to be resolved, the authors go on to detail a variety of interesting but considerably less persuasive assumptions and predictions, all based on the *stipulation* that establishing IPv4 address markets would represent the best means to "shorten the gap" between the end of IPv4 and the return to a "normal" state of Internet growth and development, that is, one that is unconstrained by IP address-related scarcity (or at least no more constrained than it has been over the last decade-plus of CIDR and hierarchical interdomain routing).

I believe that it is worth highlighting here the logic that binds these two engaging and well-written articles together into something that is, unfortunately, substantially less than the sum of its parts. If the authors are to be taken at their word that "an IPv6 Internet" represents the only currently feasible and also *satisfactory* conclusion to "the IPv4 end game," then that conclusion does not by itself entail that IPv4 markets are the only, or most obvious or effective—or even *workable*—candidate mechanisms for coordinating the distribution of IP addressing in the run-up to more widespread IPv6 adoption. And yet, that postulate is offered, without explanation or defense, as the grounding justification for an investigation of various optional features and collateral effects that the foretold IPv4 address market might have.

Many observers have committed untold pages and pixels to the exploration of hypothetical IPv4 address markets, both in IPJ and elsewhere, going back as far as RFC 1744 (1994). The two articles by Murphy and Wilson represent valuable additions to that growing corpus. However, to my knowledge, no other writings in this area have built on the proposition that IPv6 is indispensable; therefore, IPv4 addresses should be privately traded. To put it in the most generous possible terms, this claim is highly contestable. As separate and independent analyses, IPJ readers may derive many useful insights from these two articles, but attributing any special relevance to those insights based on any presumptive connection between IPv4 markets and the future necessity or viability of IPv6 would be a mistake.

—Tom Vest, Consultant
tvest@eyeconomics.com

Fragments

CSNET Receives 2009 Postel Service Award

The *Internet Society* (ISOC) has awarded the *Jonathan B. Postel Service Award* for 2009 to CSNET, the *Computer Science Network,* a research networking effort that during the early 1980s provided the critical bridge from the original research undertaken through the ARPANET to the modern Internet.

The award recognizes the pioneering work of the four principal investigators that conceived and later led the building of CSNET—Peter J. Denning, David Farber, Anthony C. Hearn and Lawrence Landweber—and the U.S. National Science Foundation program officer and visionary responsible for encouraging and funding CSNET—Kent Curtis.

Stephen Wolff, a past recipient of the Postel Award, said, "CSNET was a critical link in the transition from the research-oriented ARPANET to today's global Internet. CSNET also helped lead the way by sharing technologies, fostering connections, and nurturing the worldwide community that provided a foundation for the global expansion of the Internet."

ISOC presented the award, including a US$20,000 honorarium and a crystal engraved globe, during the 75th meeting of the *Internet Engineering Task Force* (IETF) in Stockholm, Sweden. The awardees have requested that the ISOC present the honorarium to non-profit organizations they believe support the spirit of the award.

Lynn St. Amour, President and CEO of the ISOC, said "In many ways, CSNET helped set the stage for the Internet that today reaches more than 1 billion people. CSNET's community-driven, self-sustaining governance structure was an early example of the model that helps ensure that even as today's Internet grows and evolves, it remains an open platform for innovation around the world."

CSNET began in 1981 with a five-year grant from the U.S. *National Science Foundation* (NSF). Five years later, CSNET connected more than 165 academic, government and industrial computer research groups comprised of more than 50,000 researchers, educators and students across the United States and around the world. It had concluded a seminal resource sharing agreement with the ARPANET and was self-governing and self-supporting. Open to all computer researchers, it demonstrated that researchers valued the kind of informal collaboration it made possible. CSNET's success was critical to the decision by NSF in 1986 to adopt the Internet technology for NSFNET, the network backbone to connect its supercomputing centers and their research communities. CSNET provided software, policies, and experienced alumni to the NSFNET teams. NSFNET became the first backbone of the modern Internet.

The CSNET architecture supported the Internet standards, SMTP and TCP/IP, and a variety of connection protocols including telephone dialup, X.25, and ARPANET. This architecture, along with strong technical support, enabled participants of differing means and skill levels to all join the community. CSNET pioneered the model of university, industry, government partnerships that were key to the pre-commercial Internet.

The CSNET proposal was assembled by a lengthy community consensus process that began in 1979. The four principal investigators, who led this effort and served as the project's management committee, were:

Peter Denning was head of the computer science department at Purdue University. His team included professor Douglas Comer, who was responsible for the software that ran TCP/IP over the GTE Telenet X.25 commercial packet network.

David Farber was a professor of electrical engineering at University of Delaware. His team included then graduate student David Crocker, who was responsible for Phonenet, dial-in telephone connections to relay servers for e-mail exchange.

Anthony Hearn was head of the information sciences department at RAND. His team included Michael O'Brien, who was responsible for the relays connecting CSNET and ARPANET.

Lawrence Landweber was a professor of computer science at the University of Wisconsin. His team included professor Marvin Solomon and Michael Litzkow who were responsible for the name server, a precursor of modern Directory Services.

At the NSF, the late *Kent Curtis* helped conceive the entire effort and, with assistance from Bill Kearn, saw it through its formative years. He was recognized for his pivotal role by the Computing Research Association's first distinguished service award in 1988.

The *Jonathan B. Postel Service Award* was established by the Internet Society to honor individuals or organizations that, like Jon Postel, have made outstanding contributions in service to the data communications community. The award is focused on sustained and substantial technical contributions, service to the community, and leadership. With respect to leadership, the nominating committee places particular emphasis on candidates who have supported and enabled others in addition to their own specific actions. Previous recipients of the Postel Award include Jon himself (posthumously and accepted by his mother), Scott Bradner, Daniel Karrenberg, Stephen Wolff, Peter Kirstein, Phill Gross, Jun Murai, Bob Braden and Joyce K. Reynolds (jointly), Nii Quaynor, and La Fundación Escuela Latinoamericana de Redes (EsLaRed). The award consists of an engraved crystal globe and a US$20,000 honorarium. For more information about the award, visit: **`http://www.isoc.org/postel`**

ISOC is a non-profit organization founded in 1992 to provide leadership in Internet related standards, education, and policy. ISOC is dedicated to ensuring the open development, evolution, and use of the Internet for the benefit of people throughout the world. More information is available at: `http://www.isoc.org`

NRO Declaration on RPKI

The *Number Resource Organization* (NRO) recently declared: "Over several years, a set of mechanisms has been under development for digital certification of Internet number resources, through a so-called *Resource Public Key Infrastructure,* or "RPKI." Like other PKIs, the RPKI requires one or more root authorities, to act as so-called *trust anchors* for one or more certification hierarchies.[1]

The RPKI architecture has been designed to allow a number of trust anchor configurations involving: either a single trust anchor located at the root of a single certification hierarchy; a set of independent trust anchors to be located at the roots of several independent hierarchies; or a hybrid of these. The alternative models may have advantages and disadvantages in various dimensions including: operational efficiency; alignment with resource allocation hierarchies; centralisation vs distribution of functions; recognised global or regional authority; and, operational capacity of the respective host organisations.

The *Regional Internet Registries* (RIRs) believe that the optimal eventual RPKI configuration involves a single authoritative trust anchor. That configuration may not be achievable in the short-term and the details and timelines for its implementation will depend among other things on discussions within the RIRs' communities and dialogues with others including the *Internet Architecture Board* (IAB) and the *Internet Engineering Task Force* (IETF).

In the meantime, the RIRs have agreed to undertake pragmatic implementations of RPKI services based on interim trust anchor models, such as, self-signed trust anchors. All such implementations will comply with the overall RPKI architecture. The implementations will also have the ability to evolve into a single trust anchor model and to provide robust and fully operational (and inter-operational) services for those who wish to use them. The objective is for all RIRs to be ready to start issuing certificates by no later than January 1, 2011.

The RIRs will continue working with and receiving feedback from their respective communities and industry partners to ensure effective ongoing evolution of the RPKI system."

For more information about the NRO, see `http://www.nro.net/`

ARIN Hosts 4-byte ASN Wiki

The *American Registry for Internet Numbers* (ARIN) has created a wiki to focus on issues related to 4-byte *Autonomous System Numbers* (ASNs)[2]. This wiki provides a central repository for ongoing discussion and information exchange associated with 4-byte ASN topics and issues. The wiki can be found at: **www.get4byteasn.info**

Ongoing Internet growth is rapidly depleting the existing pool of 2-byte ASNs (65,536 numbers in total). As a result, the IETF has approved the expansion of AS Numbers from 2-bytes to 4-bytes, to include over 4 billion ASNs. Following a globally coordinated policy, ARIN and the other RIRs began assigning 4-byte ASNs by request in January 2007 and by default in January 2009. However, some routers do not support the use of these 4-byte ASNs.

ARIN has set up this wiki to help educate the community about 4-byte ASN operational issues, to help vendors understand how to provide 4-byte ASN support in their products and to help network operators find those products. A wide range of community stakeholders will be able to share and benefit from information contributed to the wiki. ARIN looks forward to participation from everyone, including users, ISPs, and vendors, with interest in this topic.

Upcoming Events

The *North American Network Operators' Group* (NANOG) will meet in Dearborn, Michigan, October 18–21. Following the NANOG meeting, the *American Registry for Internet Numbers* (ARIN) will meet in the same venue October 21–23. For more information see: **http://nanog.org** and **http://arin.net**

The *Internet Engineering Task Force* (IETF) will meet in Hiroshima, Japan, November 8–13, 2009 and in Anaheim, California, March 21–26, 2010. For more information see:
http://www.ietf.org/meeting/

The *Internet Corporation for Assigned Names and Numbers* (ICANN) will meet in Seoul, Korea, October 25–30, 2009 and Nairobi, Kenya, March 7–12, 2010, and in Brussels, Belgium, June 21–25, 2010. For more information, see: **http://icann.org/**

The *Asia Pacific Regional Internet Conference on Operational Technologies* (APRICOT) will meet in Kuala Lumpur, Malaysia, February 23–March 5, 2010. For more information see:
http://www.apricot2010.net/

References

[1] Huston, Geoff, "Resource Certification," *The Internet Protocol Journal,* Volume 12, No. 1, March 2009.

[2] Huston, Geoff, "Exploring Autonomous Systems Numbers," *The Internet Protocol Journal,* Volume 9, No. 1, March 2006.

The Internet Protocol Journal, Cisco Systems
170 West Tasman Drive
San Jose, CA 95134-1706
USA

ADDRESS SERVICE REQUESTED

PRSRT STD
U.S. Postage
PAID
PERMIT No. 5187
SAN JOSE, CA

The Internet Protocol Journal is published quarterly by the Chief Technology Office, Cisco Systems, Inc.
www.cisco.com
Tel: +1 408 526-4000
E-mail: ipj@cisco.com