**19SFC162- CLOUD SECURITY NOTES**

**1.Describe in relation with information age the Evolution of cloud computing**

**The Evolution of Cloud Computing**

To understand what cloud computing is and is not, it is important to understand how this model of computing has evolved. As Alvin Toffler notes in his famous book, The Third Wave (Bantam, 1980), civilization has progressed in waves (three of them to date: the first wave was agricultural societies, the second was the industrial age, and the third is the information age). Within each wave, there have been several important subwaves. In this post-industrial information age, we are now at the beginning of what many people feel will be an era of cloud computing.

In his book The Big Switch (W.W. Norton & Co., 2008), Nicholas Carr discusses an information revolution very similar to an important change within the industrial era. Specifically, Carr equates the rise of cloud computing in the information age to electrification in the industrial age. It used to be that organizations had to provide their own power (water wheels, windmills). With electrification, however, organizations no longer provide their own power; they just plug in to the electrical grid. Carr argues that cloud computing is really the beginning of the same change for information technology. Now organizations provide their own computing resources (power). The emerging future, however, is one in which organizations will simply plug in to the cloud (computing grid) for the computing resources they need. As he puts it, “In the end the savings offered by utilities become too compelling to resist, even for the largest enterprises. The grid wins.” In fact, Part 2 of his book is about “living in the cloud” and the benefits it provides. (Carr also discusses at length some of the perceived negative consequences to society of this big switch, specifically some of the darker aspects this change brings to society.)

Carr is not alone in arguing for the benefits of cloud computing, but he has put forth what is arguably the most articulate statement of those benefits thus far. And although he focuses specifically on the economic benefits of cloud computing, he does not discuss information security problems associated with “the big switch.” We do, and that is the purpose of this book: to articulate security and privacy issues associated with “the big switch” to cloud computing.

As we noted earlier, within each wave there are subwaves, and there have already been several within the information age, as [Figure 1-1](#_3znysh7) shows. We started with mainframe computers and progressed to minicomputers, personal computers, and so forth, and we are now entering cloud computing.

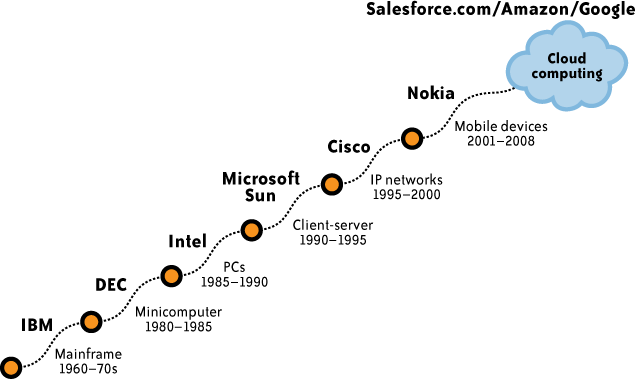


FIGURE 1-1. Subwaves within the information age

Another view illustrates that cloud computing itself is a logical evolution of computing.[Figure 1-2](#_3dy6vkm) displays cloud computing and cloud service providers (CSPs) as extensions of the Internet service provider (ISP) model.

In the beginning (ISP 1.0), ISPs quickly proliferated to provide access to the Internet for organizations and individuals. These early ISPs merely provided Internet connectivity for users and small businesses, often over dial-up telephone service. As access to the Internet became a commodity, ISPs consolidated and searched for other value-added services, such as providing access to email and to servers at their facilities (ISP 2.0). This version quickly led to specialized facilities for hosting organizations’ (customers’) servers, along with the infrastructure to support them and the applications running on them. These specialized facilities are known as collocation facilities (ISP 3.0). Those facilities are “a type of data center where multiple customers locate network, server, and storage gear and interconnect to a variety of telecommunications and other network service provider(s) with a minimum of cost and complexity.”\* As collocation facilities proliferated and became commoditized, the next step in the evolution was the formation of application service providers (ASPs), which focused on a higher value-added service of providing specialized applications for organizations, and not just the computing infrastructure (ISP 4.0). ASPs typically owned and operated the software application(s) they provided, as well as the necessary infrastructure.

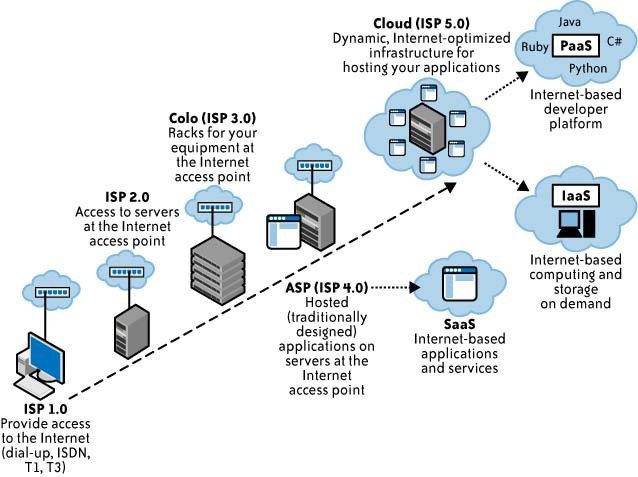


FIGURE 1-2. Evolution of cloud computing

Although ASPs might appear similar to a service delivery model of cloud computing that is referred to as software-as-a-service (SaaS), there is an important difference in how these services are provided, and in the business model. Although ASPs usually provided services to multiple customers (just as SaaS providers do today), they did so through dedicated infrastructures. That is, each customer had its own dedicated instance of an application, and that instance usually ran on a dedicated host or server. The important difference between SaaS providers and ASPs is that SaaS providers offer access to applications on a shared, not dedicated, infrastructure.

With increasing attention, some would say hype, now being paid to cloud computing, companies are increasingly claiming to be “cloudy.” Suddenly, many companies are claiming to operate “in the cloud.” Serious cloud washing is underway. Similarly, a number of computing groups have announced their efforts to promote some facet of cloud computing. Some of these groups are established (e.g., the National Institute of Standards and Technology efforts to promote standardization in cloud computing), and some of them are brand new, having emerged only with the appearance of this new computing model (e.g., the Cloud Security Alliance’s promotion of security in cloud computing, or the Open Cloud Manifesto’s promotion of cloud interoperability). Many other groups have also announced efforts dedicated to cloud computing, such as the Distributed Management Task Force (DMTF); the Information Technology Association of America, a high-technology industry association; and the Jericho Forum, an international information security thought leadership association, among many others.

**2.Draw the SPI Framework for Cloud Computing and explain relevant technologies in cloud computing**

**The SPI Framework for Cloud Computing**

A commonly agreed upon framework for describing cloud computing services goes by the acronym “SPI.” This acronym stands for the three major services provided through the cloud: software-as-a-service (SaaS), platform-as-a-service (PaaS), and infrastructure-as-a-service (IaaS). [Figure 2-3](#_17dp8vu) illustrates the relationship between services, uses, and types of clouds.

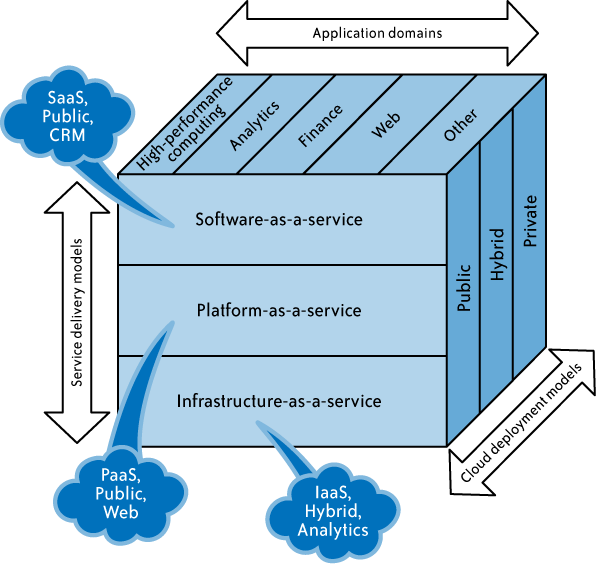


FIGURE 2-3. SPI service model

We will now explore each of these components in detail.

**Relevant Technologies in Cloud Computing**

Cloud computing isn’t so much a technology as it is the combination of many preexisting technologies. These technologies have matured at different rates and in different contexts, and were not designed as a coherent whole; however, they have come together to create a technical ecosystem for cloud computing. New advances in processors, virtualization technology, disk storage, broadband Internet connection, and fast, inexpensive servers have combined to make the cloud a more compelling solution.

[Figure 2-4](#_lnxbz9) illustrates the relevant technologies.

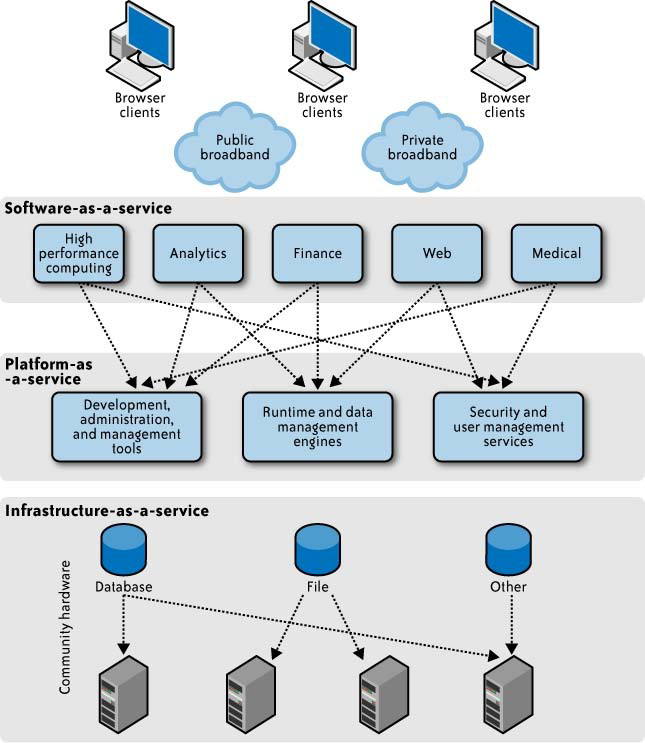


FIGURE 2-4. Architecture for relevant technologies

**Cloud access devices**

The range of access devices for the cloud has expanded in recent years. Home PCs, enterprise PCs, network computers, mobile phone devices, custom handheld devices, and custom static devices (including refrigerators) are all online. Interestingly, the growth of the iPhone and the proliferation of applications available from its App Store illustrate an improvement in terms of access to the cloud. This greater access is resulting in greater use and growth of services within the cloud. For example, you can now use Skype through the iPhone, thus bringing thispeer-to-peer network much closer to users, and Salesforce.com has introduced an application that allows users to access its services from the iPhone, as well as many other vendors.

**Browsers and thin clients**

Users of multiple device types can now access applications and information from wherever they can load a browser. Indeed, browsers are becoming increasingly sophisticated. Enterprise applications, such as SAP and Oracle, can be accessed through a browser interface—a change from when a client (a so-called “fat”) application needed to be loaded onto the desktop. The general population has become more familiar with the browser function and can use a discrete application, where the context is intuitive, without requiring training or user guides.

**High-speed broadband access**

A critical component of the cloud is the broadband network, which offers the means to connect components and provides one of the substantial differences from the utility computing concept of 30 years ago. Broadband access is now widely available, especially in global metropolitan areas. Nearly pervasive wireless access (e.g., WiFi, cellular, emerging WiMAX) is available, which has established mobile devices as entry points to the IT resources of the enterprise and the cloud.

**Data centers and server farms**

Cloud-based services require large computing capacity and are hosted in data centers and server farms. These distributed data centers and server farms span multiple locations and can be linked via internetworks providing distributed computing and service delivery capabilities.

A number of examples today illustrate the flexibility and scalability of cloud computing power. For instance, Google has linked a very large number of inexpensive servers to provide tremendous flexibility and power. Amazon’s Elastic Compute Cloud (EC2) provides virtualization in the data center to create huge numbers of virtual instances for services being requested. Salesforce.com provides SaaS to its large customer base by grouping its customers into clusters to enable scalability and flexibility.

**Storage devices**

Decreasing storage costs and the flexibility with which storage can be deployed have changed the storage landscape. The fixed direct access storage device (DASD) has been replaced with storage area networks (SANs), which have reduced costs and allowed a great deal more flexibility in enterprise storage. SAN software manages integration of storage devices and can independently allocate storage space on demand across a number of devices.

**Virtualization technologies**

Virtualization is a foundational technology platform fostering cloud computing, and it is transforming the face of the modern data center. The term virtualization refers to the abstraction of compute resources (CPU, storage, network, memory, application stack, and database) from applications and end users consuming the service. The abstraction of infrastructure yields the notion of resource democratization—whether infrastructure, applications, or information— and provides the capability for pooled resources to be made available and accessible to anyone or anything authorized to utilize them via standardized methods.

Virtualization technologies enable multitenancy cloud business models by providing a scalable, shared resource platform for all tenants. More importantly, they provide a dedicated resource view for the platform’s consumers. From an enterprise perspective, virtualization offers data center consolidation and improved IT operational efficiency. Today, enterprises have deployed virtualization technologies within data centers in various forms, including OS virtualization (VMware, Xen), storage virtualization (NAS, SAN), database virtualization, and application or software virtualization (Apache Tomcat, JBoss, Oracle App Server, WebSphere).

From a public cloud perspective, depending on the cloud services delivery model (SPI) and architecture, virtualization appears as a shared resource at various layers of the virtualized service (e.g., OS, storage, database, application).

**3.Draw and explain the core concepts of The Cloud Services Delivery Model**

**The Traditional Software Model**

Traditional software applications are based on a model with large, upfront licensing costs and annual support costs. Increasing the number of users can raise the base cost of the package due to the need for additional hardware server deployments and IT support. Licensing costs are often based on metrics that are not directly aligned with usage (server type, number of CPUs, etc., or some physical characteristic) and are not virtual. A typical enterprise software package requires hardware deployment, servers, and backup and network provisioning to accommodate the number of users on- and off-campus. Security architecture is also taxed in an effort to protect this valuable resource from unauthorized access. Traditional software applications tend to be highly customizable, which comes at a cost—in both dollars and manpower.

**The Cloud Services Delivery Model**

As we noted earlier, a cloud services delivery model is commonly referred to as an SPI and falls into three generally accepted services (see [Figure 2-7](#_23ckvvd)).

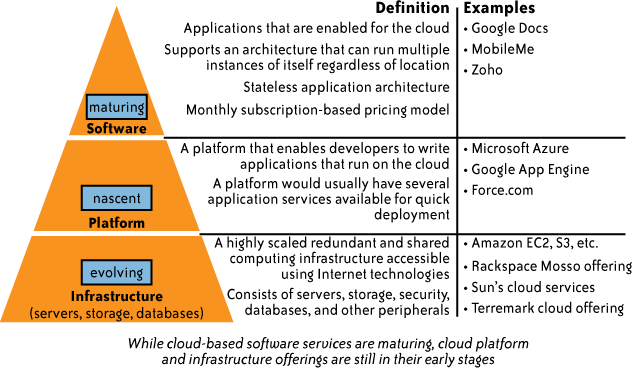


FIGURE 2-7. Cloud services delivery model

**4.Summarize the core concepts in accordance with The Software-As-a-Service Model**

#### **The Software-As-a-ServiceModel**

Traditionalmethodsofpurchasingsoftwareinvolvedthecustomerloadingthesoftwareonto hisownhardwareinreturnforalicensefee(acapitalexpense,knownas*CapEx*).Thecustomer could also purchase a maintenance agreement to receive patches to the software or other supportservices.Thecustomerwasconcernedwiththecompatibilityofoperationalsystems, patch installations, and compliance with licenseagreements.

In a SaaS model, the customer does not purchase software, but rather rents it for use on a subscription or pay-per-use model (an operational expense, known as *OpEx*). In some cases, theserviceisfreeforlimiteduse.Typically,thepurchasedserviceiscompletefromahardware, software,andsupportperspective.Theuseraccessestheservicethroughanyauthorizeddevice. Insomecases,preparatoryworkisrequiredtoestablishcompany-specificdatafortheservice tobefullyusedandpotentiallyintegratedwithotherapplicationsthatarenotpartoftheSaaS platform.

**Key benefits of a SaaS model include the following:**

* SaaS enables the organization to outsource the hosting and management of applications to a third party (software vendor and service provider) as a means of reducing the cost of application software licensing, servers, and other infrastructure and personnel required to host the applicationinternally.
* SaaSenablessoftwarevendorstocontrolandlimituse,prohibitscopyinganddistribution, and facilitates the control of all derivative versions of their software. SaaS centralized control often allows the vendor or supplier to establish an ongoing revenue stream with multiple businesses and users without preloading software in each device in an organization.
* Applications delivery using the SaaS model typically uses the one-to-many delivery approach, with the Web as the infrastructure. An end user can access a SaaS application via a web browser; some SaaS vendors provide their own interface that is designed to support features that are unique to theirapplications.
* A typical SaaS deployment does not require any hardware and can run over the existing Internet access infrastructure. Sometimes changes to firewall rules and settings may be requiredtoallowtheSaaSapplicationtorunsmoothly.
* Management of a SaaS application is supported by the vendor from the end user perspective, whereby a SaaS application can be configured using an API, but SaaS applications cannot be completelycustomized.

A typical SaaS offering is SaaS over a public network, in which a SaaS-based application is delivered via the Internet to the organization’s firewall.

The single most important architectural difference between the traditional software model and the SaaS model is the number of tenants the application supports. The traditional software modelisanisolated,single-tenantmodel,whichmeansacustomerbuysasoftwareapplicationand installs it on a server. The server runs only that specific application and only for that single customer’s end user group. The SaaS model is a multitenant architecture model, which means the physical backend hardware infrastructure is shared among many different customers, but logically is unique for eachcustomer.

Multitenant architecture design maximizes the sharing of resources across tenants, but is still able to securely differentiate data belonging to each tenant. For example, when a user at one company accesses customer information by using a SaaS Customer Relationship Management (CRM) application, the application instance that the user connects to can accommodate users from dozens, or even hundreds, of other companies—all completely unbeknownst to any of the otherusers.

SaaS solutions are very different from application service provider (ASP) solutions. There are two main explanations for this:

* + ASP applications are traditional, single-tenant applications, but are hosted by a third party. They are client/server applications with HTML frontends added to allow remote access to theapplication.
  + ASPapplicationsarenotwrittenasNet-nativeapplications.Asaresult,theirperformance may be poor, and application updates are no better than self-managed premise-based applications.

Bycomparison,SaaSapplicationsaremultitenantapplicationsthatarehostedbyavendorwith expertise in the applications and that have been designed as Net-native applications and are updated on an ongoingbasis.

**5.Discuss The Platform-As-a-Service Model from your perspective**

**The Platform-As-a-Service Model**

In a platform-as-a-service (PaaS) model, the vendor offers a development environment to application developers, who develop applications and offer those services through the provider’s platform. The provider typically develops toolkits and standards for development, and channels for distribution and payment. The provider typically receives a payment for providing the platform and the sales and distribution services. This enables rapid propagation of software applications, given the low cost of entry and the leveraging of established channels for customer acquisition.

PaaS is a variation of SaaS whereby the development environment is offered as a service. The developers use the building blocks (e.g., predefined blocks of code) of the vendor’s development environment to create their own applications.

PaaS solutions are development platforms for which the development tool itself is hosted in the cloud and accessed through a browser. With PaaS, developers can often build web applications without installing any tools on their computer, and can then deploy those applications without any specialized system administration skills.

PaaS systems are useful because they enable lone developers and start-up companies to deploy web-based applications without the cost and complexity of buying servers and setting them up. The benefits of PaaS lie in greatly increasing the number of people who can develop, maintain, and deploy web applications. In short, PaaS offers to democratize the development of web applications in much the same way that Microsoft Access democratized the development of the client/server application.

Today, building web applications requires expert developers with three highly specialized skill sets:

Backend server development (e.g., Java/J2EE)

Frontend client development (e.g., JavaScript/Dojo)

**Website administration**

PaaS offers the potential for general developers to build web applications without needing specialized expertise, which allows an entire generation of Microsoft Access, Lotus Notes, and PowerBuilder developers to build web applications without too steep a learning curve.

The alternative to PaaS is to develop web applications using desktop development tools, such as Eclipse or Microsoft Access, and then manually deploy those applications to a cloud-hosting provider, such as Amazon Web Services (AWS).

At a minimum, a PaaS solution should include the following elements:

A PaaS development studio solution should be browser-based.

An end-to-end PaaS solution should provide a high-productivity integrated development environment (IDE) running on the actual target delivery platform so that debugging and test scenarios run in the same environment as production deployment.

A PaaS solution should provide integration with external web services and databases.

A PaaS solution must provide comprehensive monitoring of application and user activity, to help developers understand their applications and effect improvements.

Scalability, reliability, and security should be built into a PaaS solution without requiring additional development, configuration, or other costs. Multitenancy (the ability for an application to automatically partition state and data to service an arbitrary number of users) must be assumed without additional work of any sort.

A PaaS solution must support both formal and on-demand collaboration throughout the entire software life cycle (development, testing, documentation, and operations), while maintaining the security of source code and associated intellectual property.

A PaaS solution should support pay-as-you-go metered billing.[Table 2-1](#_19c6y18) illustrates the different components of a typical PaaS offering.

TABLE 2-1. PaaS components

| Client capabilities | Browser-based development tools: Google Web Toolkit, Google Gears, Mashup Editor, Google Gadgets, etc. |
| --- | --- |
| Cloud computing services | Cloud-based runtime: EC2, Google App Engine, etc. |
| General purpose support services | Web services tools: Simple Storage Service, Simple DB, MTurk, GAE Datastore, GDate, Google Accounts, Social Graph API, etc. |

PaaS platforms also have functional differences from traditional development platforms, including:

**Multitenant development tools**

Traditional development tools are intended for a single user; a cloud-based studio must support multiple users, each with multiple active projects.

**Multitenant deployment architecture**

Scalability is often not a concern of the initial development effort and is left instead for the system administrators to handle when the project deploys. In PaaS, scalability of the application and data tiers must be built-in (e.g., load balancing and failover should be basic elements of the developing platform).

**Integrated management**

Traditional development solutions (usually) are not associated with runtime monitoring, but in PaaS the monitoring ability should be built into the development platform.

**Integrated billing**

PaaS offerings require mechanisms for billing based on usage that are unique to the SaaS world.

[Table 2-2](#_nmf14n) compares the flexibility offered by in-house development platforms and PaaS.

TABLE 2-2. Comparison of in-house and PaaS development platforms

| Supported area | In-house development platform | PaaS |
| --- | --- | --- |
| Endpoints: desktops, browsers, mobile devices | Most endpoints and clients are supported | Mostly browser-based |
| Business logic | Multiple vendors are supported | Restricted by PaaS model |
| Application development framework | Java Platform, Enterprise Edition (Java EE), .NET, etc. | Restricted by PaaS model |
| Application servers | Multiple vendors are supported | Provided by PaaS |
| Databases | Multiple vendors are supported | Provided by PaaS |
| Servers and VMs | Multiple vendors are supported | Provided by PaaS |
| Storage | Multiple vendors are supported | Provided by PaaS |

**6.Compare the various features involved in The Infrastructure-As-a-Service Model**

**The Infrastructure-As-a-Service Model**

In the traditional hosted application model, the vendor provides the entire infrastructure for a customer to run his applications. Often, this entails housing dedicated hardware that is purchased or leased for that specific application. The IaaS model also provides the infrastructure to run the applications, but the cloud computing approach makes it possible to offer a pay-per- use model and to scale the service depending on demand. From the IaaS provider’s perspective, it can build an infrastructure that handles the peaks and troughs of its customers’ demands and add new capacity as the overall demand increases. Similarly, in a hosted application model, the IaaS vendor can cover application hosting only, or can extend to other services (such as application support, application development, and enhancements) and can support the more comprehensive outsourcing of IT.

The IaaS model is similar to utility computing, in which the basic idea is to offer computing services in the same way as utilities. That is, you pay for the amount of processing power, disk space, and so on that you actually consume. IaaS is typically a service associated with cloud computing and refers to online services that abstract the user from the details of infrastructure, including physical computing resources, location, data partitioning, scaling, security, backup, and so on. In cloud computing, the provider is in complete control of the infrastructure. Utility computing users, conversely, seek a service that allows them to deploy, manage, and scale online services using the provider’s resources and pay for resources the customer consumes. However, the customer wants to be in control of the geographic location of the infrastructure and what runs on each server.

Features available for a typical IaaS system include:

**Scalability**

The ability to scale infrastructure requirements, such as computing resources, memory, and storage (in near-real-time speeds) based on usage requirements

**Pay as you go**

The ability to purchase the exact amount of infrastructure required at any specific time

**Best-of-breed technology and resources**

Access to best-of-breed technology solutions and superior IT talent for a fraction of the cost

**7.Distinguish the essential characteristics in 4 different Cloud Deployment Models**

**Cloud Deployment Models**

The term cloud is a metaphor for the Internet and is a simplified representation of the complex, internetworked devices and connections that form the Internet. Private and public clouds are subsets of the Internet and are defined based on their relationship to the enterprise. Private and public clouds may also be referred to as internal or external clouds; the differentiation is based on the relationship of the cloud to the enterprise.

The public and private cloud concepts are important because they support cloud computing, which enables the provisioning of dynamic, scalable, virtualized resources over Internet connections by a vendor or an enterprise IT organization to customers for a fee. The end users who use the services offered via cloud computing may not have knowledge of, expertise in, or control over the technology infrastructure that supports them.

The majority of cloud computing infrastructure consists of reliable services delivered through data centers and built on servers with different levels of virtualization technologies. The services are accessible anywhere that access to networking infrastructure is available. The cloud appears as a single point of access for all consumer computing needs. Commercial offerings should meet the quality of service requirements of customers and typically offer service-level agreements (SLAs). Open standards are critical to the growth of cloud computing, and open source software has provided the foundation for many cloud computing implementations (e.g., the use of Xen in AWS).

**Public Clouds**

Public clouds (or external clouds) describe cloud computing in the traditional mainstream sense, whereby resources are dynamically provisioned on a fine-grained, self-service basis over the Internet, via web applications or web services, from an off-site, third-party provider who shares resources and bills on a fine-grained, utility-computing basis.

A public cloud is hosted, operated, and managed by a third-party vendor from one or more data centers. The service is offered to multiple customers (the cloud is offered to multiple tenants) over a common infrastructure; see [Figure 2-8](#_3ygebqi).

In a public cloud, security management and day-to-day operations are relegated to the third- party vendor, who is responsible for the public cloud service offering. Hence, the customer of the public cloud service offering has a low degree of control and oversight of the physical and logical security aspects of a private cloud.

**Private Clouds**

Private clouds and internal clouds are terms used to describe offerings that emulate cloud computing on private networks. These (typically virtualization automation) products claim to deliver some benefits of cloud computing without the pitfalls, capitalizing on data security, corporate governance, and reliability concerns. Organizations must buy, build, and manage them and, as such, do not benefit from lower upfront capital costs and less hands-on management. The organizational customer for a private cloud is responsible for the operation of his private cloud.

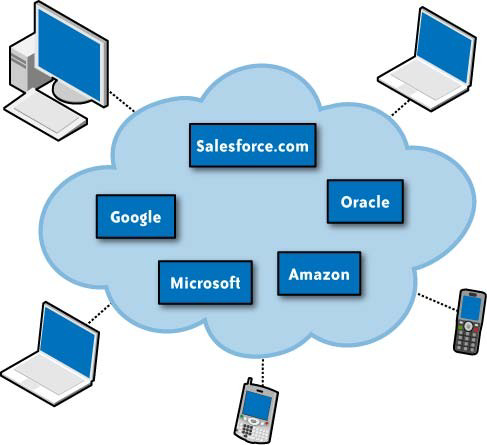


FIGURE 2-8. Public cloud

Private clouds differ from public clouds in that the network, computing, and storage infrastructure associated with private clouds is dedicated to a single organization and is not shared with any other organizations (i.e., the cloud is dedicated to a single organizational tenant). As such, a variety of private cloud patterns have emerged:

**Dedicated**

Private clouds hosted within a customer-owned data center or at a collocation facility, and operated by internal IT departments

**Community**

Private clouds located at the premises of a third party; owned, managed, and operated by a vendor who is bound by custom SLAs and contractual clauses with security and compliance requirements

**Managed**

Private cloud infrastructure owned by a customer and managed by a vendor

In general, in a private cloud operating model, the security management and day-to-day operation of hosts are relegated to internal IT or to a third party with contractual SLAs. By virtue of this direct governance model, a customer of a private cloud should have a high degree of control and oversight of the physical and logical security aspects of the private cloud infrastructure—both the hypervisor and the hosted virtualized OSs. With that high degree of control and transparency, it is easier for a customer to comply with established corporate security standards, policies, and regulatory compliance.

**Hybrid Clouds**

A hybrid cloud environment consisting of multiple internal and/or external providers is a possible deployment for organizations. With a hybrid cloud, organizations might run non-core applications in a public cloud, while maintaining core applications and sensitive data in-house in a private cloud (see [Figure 2-9](#_kgcv8k)).

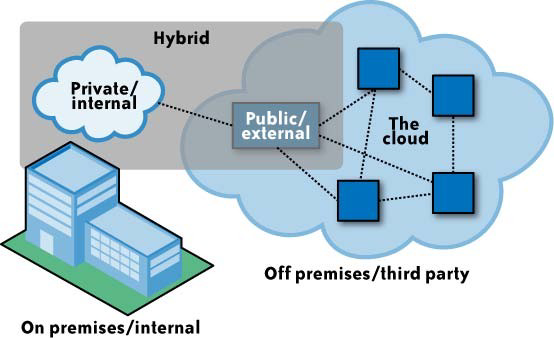


FIGURE 2-9. Hybrid cloud

[Figure 2-10](#_34g0dwd) lists some examples of CSPs.

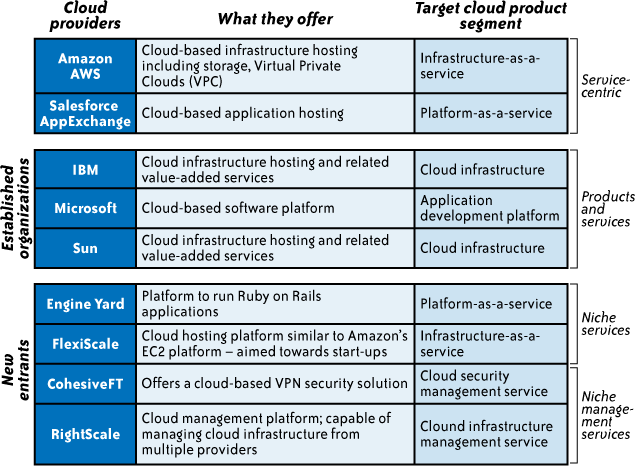


FIGURE 2-10. CSP examples and their respective offerings

Services provided through the integration of cloud components are evolving, barriers are being overcome, and enablers are being developed. A major concern is to trust that a company’s or an individual’s information is both secure and private. Establishing this trust is a major milestone in the adoption of the full range of cloud computing; see the next section for more details.

**8.Summarize the role of Infrastructure Security at The Network Level**

**Infrastructure Security: The Network Level**

When looking at the network level of infrastructure security, it is important to distinguish between public clouds and private clouds, as we explained in [Chapter 2](#_2iq8gzs). With private clouds, there are no new attacks, vulnerabilities, or changes in risk specific to this topology that information security personnel need to consider. Although your organization’s IT architecture may change with the implementation of a private cloud, your current network topology will probably not change significantly. If you have a private extranet in place (e.g., for premium customers or strategic partners), for practical purposes you probably have the network topology for a private cloud in place already. The security considerations you have today apply to a private cloud infrastructure, too. And the security tools you have in place (or should have in place) are also necessary for a private cloud and operate in the same way. [Figure 3-1](#_xvir7l) shows the topological similarities between a secure extranet and a private cloud.

However, if you choose to use public cloud services, changing security requirements will require changes to your network topology. You must address how your existing network topology interacts with your cloud provider’s network topology. There are four significant risk factors in this use case:

Ensuring the confidentiality and integrity of your organization’s data-in-transit to and from your public cloud provider

Ensuring proper access control (authentication, authorization, and auditing) to whatever resources you are using at your public cloud provider

Ensuring the availability of the Internet-facing resources in a public cloud that are being used by your organization, or have been assigned to your organization by your public cloud providers

Replacing the established model of network zones and tiers with domains We will discuss each of these risk factors in the sections that follow.

**Ensuring Data Confidentiality and Integrity**

Some resources and data previously confined to a private network are now exposed to the Internet, and to a shared public network belonging to a third-party cloud provider.

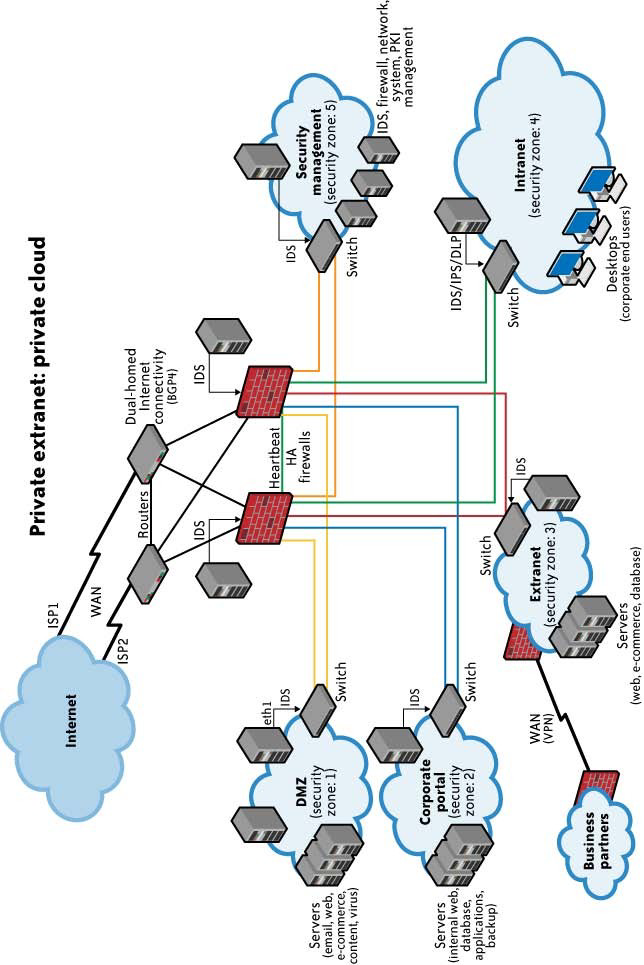


FIGURE 3-1. Generic network topology for private cloud computing

An example of problems associated with this first risk factor is an Amazon Web Services (AWS) security vulnerability reported in December 2008.\* In a blog post, the author detailed a flaw in the digital signature algorithm used when “... making Query (aka REST) requests to Amazon SimpleDB, to Amazon Elastic Compute Cloud (EC2), or to Amazon Simple Queue Service (SQS) over HTTP.” Although use of HTTPS (instead of HTTP) would have mitigated the integrity risk, users not using HTTPS (but using HTTP) did face an increased risk that their data could have been altered in transit without their knowledge.

**Ensuring Proper Access Control**

Since some subset of these resources (or maybe even all of them) is now exposed to the Internet, an organization using a public cloud faces a significant increase in risk to its data. The ability to audit the operations of your cloud provider’s network (let alone to conduct any real- time monitoring, such as on your own network), even after the fact, is probably non-existent. You will have decreased access to relevant network-level logs and data, and a limited ability to thoroughly conduct investigations and gather forensic data.

An example of the problems associated with this second risk factor is the issue of reused (reassigned) IP addresses. Generally speaking, cloud providers do not sufficiently “age” IP addresses when they are no longer needed for one customer. Addresses are usually reassigned and reused by other customers as they become available. From a cloud provider’s perspective this makes sense. IP addresses are a finite quantity and a billable asset. However, from a customer’s security perspective, the persistence of IP addresses that are no longer in use can present a problem. A customer can’t assume that network access to its resources is terminated upon release of its IP address. There is necessarily a lag time between the change of an IP address in DNS and the clearing of that address in DNS caches. There is a similar lag time between when physical (i.e., MAC) addresses are changed in ARP tables and when old ARP addresses are cleared from cache; an old address persists in ARP caches until they are cleared. This means that even though addresses might have been changed, the (now) old addresses are still available in cache, and therefore they still allow users to reach these supposedly non-existent resources. Recently, there were many reports of problems with “non-aged” IP addresses at one of the largest cloud providers; this was likely an impetus for an AWS announcement of the Amazon Elastic IP capabilities in March 2008.† (With Elastic IP addresses, customers are given a block of five routable IP addresses over which they control assignment.) Additionally, according to Simson Garfinkel:

A separate ongoing problem with the load balancers causes them to terminate any TCP/IP connection that contains more than 231 bytes. This means that objects larger than 2GB must be stored to S3 in several individual transactions, with each of those transactions referring to different byte ranges of the same object.‡

However, the issue of “non-aged” IP addresses and unauthorized network access to resources does not apply only to routable IP addresses (i.e., resources intended to be reachable directly from the Internet). The issue also applies to cloud providers’ internal networks for customer use and the assignment of non-routable IP addresses.§ Although your resources may not be directly reachable from the Internet, for management purposes your resources must be accessible within the cloud provider’s network via private addressing. (Every public/Internet- facing resource also has a private address.) Other customers of your cloud provider may not be well intentioned and might be able to reach your resources internally via the cloud provider’s networks.‖ As reported in The Washington Post, AWS has had problems with abuses of its resources affecting the public and other customers.#

Some products emerging onto the market\* will help alleviate the problem of IP address reuse, but unless cloud providers offer these products as managed services, customers are paying for yet another third-party product to solve a problem that their cloud provider’s practices created for them.

**Ensuring the Availability of Internet-Facing Resources**

Reliance on network security has increased because an increased amount of data or an increased number of organizational personnel now depend on externally hosted devices to ensure the availability of cloud-provided resources. Consequently, the three risk factors enumerated in the preceding section must be acceptable to your organization.

BGP† prefix hijacking (i.e., the falsification of Network Layer Reachability Information) provides a good example of this third risk factor. Prefix hijacking involves announcing an

‡ See Section 3.3, “An Evaluation of Amazon’s Grid Computing Services: EC2, S3 and SQS,” by Simson L. Garfinkel; TR-08-07, Computer Science Group, Harvard University, Cambridge, Massachusetts.

§ See RFC 1918, “Address Allocation for Private Internets,” for further information.

‖ For example, see “Instance Addressing and Network Security” in the Amazon Elastic Compute Cloud Developer Guide (API Version 2008-12-01).

# “Amazon: Hey Spammers, Get Off My Cloud!” reported in The Washington Post, July 1, 2008.

\* An example is CohesiveFT’s VPN-Cubed, but this product is not available as a cloud provider service from most cloud providers—which would mean yet another third-party solution to integrate into your cloud environment. However, cloud provider AWS does offer this product as a service.

† Border Gateway Protocol is an interdomain routing protocol used in the core of the Internet. You can find more information about BGP in RFC 4271, “A Border Gateway Protocol 4 (BGP-4).”autonomous system‡ address space that belongs to someone else without her permission. Such announcements often occur because of a configuration mistake, but that misconfiguration may still affect the availability of your cloud-based resources. According to a study presented to the North American Network Operators Group (NANOG) in February 2006, several hundred such misconfigurations occur per month.§ Probably the best known example of such a misconfiguration mistake occurred in February 2008 when Pakistan Telecom made an error by announcing a dummy route for YouTube to its own telecommunications partner, PCCW, based in Hong Kong. The intent was to block YouTube within Pakistan because of some supposedly blasphemous videos hosted on the site. The result was that YouTube was globally unavailable for two hours.‖

In addition to misconfigurations, there are deliberate attacks as well. Although prefix hijacking due to deliberate attacks is far less common than misconfigurations, it still occurs and can block access to data. According to the same study presented to NANOG, attacks occur fewer than 100 times per month. Although prefix hijackings are not new, that attack figure will certainly rise, and probably significantly, along with a rise in cloud computing. As the use of cloud computing increases, the availability of cloud-based resources increases in value to customers. That increased value to customers translates to an increased risk of malicious activity to threaten that availability.

DNS# attacks are another example of problems associated with this third risk factor. In fact, there are several forms of DNS attacks to worry about with regard to cloud computing.

Although DNS attacks are not new and are not directly related to the use of cloud computing, the issue with DNS and cloud computing is an increase in an organization’s risk at the network level because of increased external DNS querying (reducing the effectiveness of “split horizon” DNS configurations\*) along with some increased number of organizational personnel being more dependent on network security to ensure the availability of cloud-provided resources being used.

That is not to say that internal DNS systems are entirely free of attacks—just that they are safer than external DNS systems and queries using them. For example, see the paper “Corrupted DNS Resolution Paths: The Rise of a Malicious Resolution Authority,” written by members of the faculty of the Georgia Institute of Technology.

Although the “Kaminsky Bug”† (CVE-2008-1447, “DNS Insufficient Socket Entropy Vulnerability”) garnered most of the network security attention in 2008, other DNS problems impact cloud computing as well. Not only are there vulnerabilities in the DNS protocol and in implementations of DNS,‡ but also there are fairly widespread DNS cache poisoning attacks whereby a DNS server is tricked into accepting incorrect information. Although many people thought DNS cache poisoning attacks had been quashed several years ago, that is not true, and these attacks are still very much a problem—especially in the context of cloud computing.

Variants of this basic cache poisoning attack include redirecting the target domain’s name server (NS), redirecting the NS record to another target domain, and responding before the real NS (called DNS forgery).

A final example of problems associated with this third risk factor is denial of service (DoS) and distributed denial of service (DDoS) attacks. Again, although DoS/DDoS attacks are not new and are not directly related to the use of cloud computing, the issue with these attacks and cloud computing is an increase in an organization’s risk at the network level because of some increased use of resources external to your organization’s network. For example, there continue to be rumors of continued DDoS attacks on AWS, making the services unavailable for hours at a time to AWS users.§ (Amazon has not acknowledged that service interruptions are in fact due to DDoS attacks.)

However, when using IaaS, the risk of a DDoS attack is not only external (i.e., Internet-facing). There is also the risk of an internal DDoS attack through the portion of the IaaS provider’s network used by customers (separate from the IaaS provider’s corporate network). That internal (non-routable) network is a shared resource, used by customers for access to their non-public instances (e.g., Amazon Machine Images or AMIs) as well as by the provider for management of its network and resources (such as physical servers). If I were a rogue customer, there would be nothing to prevent me from using my customer access to this internal network to find and attack other customers, or the IaaS provider’s infrastructure—and the provider would probably not have any detective controls in place to even notify it of such an attack. The only preventive controls other customers would have would be how hardened their instances (e.g., AMIs) are, and whether they are taking advantage of a provider’s capabilities to firewall off groups of instances (e.g., AWS).

**Replacing the Established Model of Network Zones and Tiers with Domains**

The established isolation model of network zones and tiers no longer exists in the public IaaS and PaaS clouds. For years, network security has relied on zones, such as intranet versus extranet and development versus production, to segregate network traffic for improved security. This model was based on exclusion—only individuals and systems in specific roles have access to specific zones. Similarly, systems within a specific tier often have only specific access within or across a specific tier. For example, systems within a presentation tier are not allowed to communicate directly with systems in the database tier, but can communicate only with an authorized system within the application zone. SaaS clouds built on public IaaS or PaaS clouds have similar characteristics. However, a public SaaS built on a private IaaS (e.g., Salesforce.com) may follow the traditional isolation model, but that topology information is not typically shared with customers.

The traditional model of network zones and tiers has been replaced in public cloud computing with “security groups,” “security domains,” or “virtual data centers” that have logical separation between tiers but are less precise and afford less protection than the formerly established model. For example, the security groups feature in AWS allows your virtual machines (VMs) to access each other using a virtual firewall that has the ability to filter traffic based on IP address (a specific address or a subnet), packet types (TCP, UDP, or ICMP), and ports (or a range of ports). Domain names are used in various networking contexts and application-specific naming and addressing purposes, based on DNS. For example, Google’s App Engine provides a logical grouping of applications based on domain names such as mytestapp.test.mydomain.com and myprodapp.prod.mydomain.com.

In the established model of network zones and tiers, not only were development systems logically separated from production systems at the network level, but these two groups of systems were also physically separated at the host level (i.e., they ran on physically separated servers in logically separated network zones). With cloud computing, however, this separation no longer exists. The cloud computing model of separation by domains provides logical separation for addressing purposes only. There is no longer any “required” physical separation, as a test domain and a production domain may very well be on the same physical server.

Furthermore, the former logical network separation no longer exists; logical separation now is at the host level with both domains running on the same physical server and being separated only logically by VM monitors (hypervisors).

**Network-Level Mitigation**

Given the factors discussed in the preceding sections, what can you do to mitigate these increased risk factors? First, note that network-level risks exist regardless of what aspects of “cloud computing” services are being used (e.g., software-as-a-service, platform-as-a-service, or infrastructure-as-a-service). The primary determination of risk level is therefore not which

\*aaS is being used, but rather whether your organization intends to use or is using a public,

private, or hybrid cloud. Although some IaaS clouds offer virtual network zoning, they may not match an internal private cloud environment that performs stateful inspection and other network security measures.

If your organization is large enough to afford the resources of a private cloud, your risks will decrease—assuming you have a true private cloud that is internal to your network. In some cases, a private cloud located at a cloud provider’s facility can help meet your security requirements but will depend on the provider capabilities and maturity.

You can reduce your confidentiality risks by using encryption; specifically by using validated implementations of cryptography for data-in-transit. Secure digital signatures make it much more difficult, if not impossible, for someone to tamper with your data, and this ensures data integrity.

Availability problems at the network level are far more difficult to mitigate with cloud computing—unless your organization is using a private cloud that is internal to your network topology. Even if your private cloud is a private (i.e., non-shared) external network at a cloud provider’s facility, you will face increased risk at the network level. A public cloud faces even greater risk. But let’s keep some perspective here—greater than what?

Even large enterprises with significant resources face considerable challenges at the network level of infrastructure security. Are the risks associated with cloud computing actually higher than the risks enterprises are facing today? Consider existing private and public extranets, and take into account partner connections when making such a comparison. For large enterprises without significant resources, or for small to medium-size businesses (SMBs), is the risk of using public clouds (assuming that such enterprises lack the resources necessary for private clouds) really higher than the risks inherent in their current infrastructures? In many cases, the answer is probably no—there is not a higher level of risk.

[Table 3-1](#_36ei31r) lists security controls at the network level.

TABLE 3-1. Security controls at the network level

| Threat outlook | Low (with the exception of DoS attacks) |
| --- | --- |
| Preventive controls | Network access control supplied by provider (e.g., firewall), encryption of data in transit (e.g., SSL, IPSec) |
| Detective controls | Provider-managed aggregation of security event logs (security incident and event management, or SIEM), network-based intrusion detection system/intrusion prevention system (IDS/IPS) |

**9.Infrastructure Security: The Host Level**

**Infrastructure Security: The Host Level**

When reviewing host security and assessing risks, you should consider the context of cloud services delivery models (SaaS, PaaS, and IaaS) and deployment models (public, private, and hybrid). Although there are no known new threats to hosts that are specific to cloud computing, some virtualization security threats—such as VM escape, system configuration drift, and insider threats by way of weak access control to the hypervisor—carry into the public cloud computing environment. The dynamic nature (elasticity) of cloud computing can bring new operational challenges from a security management perspective. The operational model motivates rapid provisioning and fleeting instances of VMs. Managing vulnerabilities and patches is therefore much harder than just running a scan, as the rate of change is much higher than in a traditional data center.

In addition, the fact that the clouds harness the power of thousands of compute nodes, combined with the homogeneity of the operating system employed by hosts, means the threats can be amplified quickly and easily—call it the “velocity of attack” factor in the cloud. More importantly, you should understand the trust boundary and the responsibilities that fall on your shoulders to secure the host infrastructure that you manage. And you should compare the same with providers’ responsibilities in securing the part of the host infrastructure the CSP manages.

**SaaS and PaaS Host Security**

In general, CSPs do not publicly share information related to their host platforms, host operating systems, and the processes that are in place to secure the hosts, since hackers can exploit that information when they are trying to intrude into the cloud service. Hence, in the context of SaaS (e.g., Salesforce.com, Workday.com) or PaaS (e.g., Google App Engine, Salesforce.com’s Force.com) cloud services, host security is opaque to customers and the responsibility of securing the hosts is relegated to the CSP. To get assurance from the CSP on the security hygiene of its hosts, you should ask the vendor to share information under a non- disclosure agreement (NDA) or simply demand that the CSP share the information via a controls assessment framework such as SysTrust or ISO 27002. From a controls assurance perspective, the CSP has to ensure that appropriate preventive and detective controls are in place and will have to ensure the same via a third-party assessment or ISO 27002 type assessment framework.

Since virtualization is a key enabling technology that improves host hardware utilization, among other benefits, it is common for CSPs to employ virtualization platforms, including Xen and VMware hypervisors, in their host computing platform architecture. You should understand how the provider is using virtualization technology and the provider’s process for securing the virtualization layer.

Both the PaaS and SaaS platforms abstract and hide the host operating system from end users with a host abstraction layer. One key difference between PaaS and SaaS is the accessibility of the abstraction layer that hides the operating system services the applications consume. In the case of SaaS, the abstraction layer is not visible to users and is available only to the developers and the CSP’s operations staff, where PaaS users are given indirect access to the host abstraction layer in the form of a PaaS application programming interface (API) that in turn interacts with the host abstraction layer. In short, if you are a SaaS or a PaaS customer, you are relying on the CSP to provide a secure host platform on which the SaaS or PaaS application is developed and deployed by the CSP and you, respectively.

In summary, host security responsibilities in SaaS and PaaS services are transferred to the CSP. The fact that you do not have to worry about protecting hosts from host-based security threats is a major benefit from a security management and cost standpoint. However, as a customer, you still own the risk of managing information hosted in the cloud services. It’s your responsibility to get the appropriate level of assurance regarding how the CSP manages host security hygiene.

**IaaS Host Security**

Unlike PaaS and SaaS, IaaS customers are primarily responsible for securing the hosts provisioned in the cloud. Given that almost all IaaS services available today employ virtualization at the host layer, host security in IaaS should be categorized as follows:

**Virtualization software security**

The software layer that sits on top of bare metal and provides customers the ability to create and destroy virtual instances. Virtualization at the host level can be accomplished using any of the virtualization models, including OS-level virtualization (Solaris containers, BSD jails, Linux-VServer), paravirtualization (a combination of the hardware version and versions of Xen and VMware), or hardware-based virtualization (Xen, VMware, Microsoft Hyper-V). It is important to secure this layer of software that sits between the hardware and the virtual servers. In a public IaaS service, customers do not have access to this software layer; it is managed by the CSP only.

**Customer guest OS or virtual server security**

The virtual instance of an operating system that is provisioned on top of the virtualization layer and is visible to customers from the Internet; e.g., various flavors of Linux, Microsoft, and Solaris. Customers have full access to virtual servers.

**Virtualization Software Security**

Since the CSP manages the virtualization software that sits on top of the hardware, customers will have neither visibility nor access to this software. Hardware or OS virtualization enables the sharing of hardware resources across multiple guest VMs without interfering with each other so that you can safely run several operating systems and applications at the same time on a single computer. For the purpose of simplicity, we made an assumption that IaaS services are using “bare metal hypervisor” technologies (also known as type 1 hypervisors), such as VMware ESX, Xen, Oracle VM, and Microsoft’s Hyper-V. These hypervisors support a variety of guest OSs, including Microsoft Windows, various Linux “flavors,” and Sun’s OpenSolaris.

Given that hypervisor virtualization is the essential ingredient that guarantees compartmentalization and isolation of customer VMs from each other in a multitenant environment, it is very important to protect the hypervisors from unauthorized users. A new arms race between hacker and defender (CSP) in the realm of virtualization security is already underway. Since virtualization is very critical to the IaaS cloud architecture, any attack that could compromise the integrity of the compartments will be catastrophic to the entire customer base on that cloud. A recent incident at a tiny UK-based company called Vaserv.com exemplifies the threat to hypervisor security. By exploiting a zero-day vulnerability in HyperVM, a virtualization application made by a company called Lxlabs, hackers destroyed 100,000 websites hosted by Vaserv.com. The zero-day vulnerability gave the attackers the ability to execute sensitive Unix commands on the system, including rm -rf, which forces a recursive delete of all files. Evidently, just days before the intrusion, an anonymous user posted on a hacker website called milw0rm a long list of yet-unpatched vulnerabilities in Kloxo, a hosting control panel that integrates into HyperVM. The situation was worse for approximately 50% of Vaserv’s customers who signed up for unmanaged service, which doesn’t include data backup. It remains unclear whether those website owners will ever be able to retrieve their lost data.

CSPs should institute the necessary security controls, including restricting physical and logical access to hypervisor and other forms of employed virtualization layers. IaaS customers should understand the technology and security process controls instituted by the CSP to protect the hypervisor. This will help you to understand the compliance and gaps with reference to your host security standard, policies, and regulatory compliances. However, in general, CSPs lack transparency in this area and you may have no option but to take a leap of faith and trust CSPs to provide an “isolated and secured virtualized guest OS.”

**Threats to the hypervisor**

The integrity and availability of the hypervisor are of utmost importance and are key to guaranteeing the integrity and availability of a public cloud built on a virtualized environment.

A vulnerable hypervisor could expose all user domains to malicious insiders. Furthermore, hypervisors are potentially susceptible to subversion attacks. To illustrate the vulnerability of the virtualization layer, some members of the security research community demonstrated a “Blue Pill” attack on a hypervisor. During Black Hat 2008 and Black Hat DC 2009‖ Joanna Rutkowska, Alexander Tereshkin, and Rafal Wojtczuk from Invisible Things Lab demonstrated a number of ways to compromise Xen’s virtualization.# Although Rutkowska and her team

Black Hat DC 2009.have identified problems with Xen implementations, generally they seem quite positive about the Xen approach. But their demonstration does illustrate the complexity of securing virtualized systems and the need for new approaches to protect hypervisors from such attacks.

Since virtualization layers within public clouds for the most part are proprietary and closed source (although some may employ a derivative of open source virtualization software such as Xen), the source code of software used by CSPs is not available for scrutiny by the security research community.

**Virtual Server Security**

Customers of IaaS have full access to the virtualized guest VMs that are hosted and isolated from each other by hypervisor technology. Hence customers are responsible for securing and ongoing security management of the guest VM.

A public IaaS, such as Amazon’s Elastic Compute Cloud (EC2), offers a web services API to perform management functions such as provisioning, decommissioning, and replication of virtual servers on the IaaS platform. These system management functions, when orchestrated appropriately, can provide elasticity for resources to grow or shrink in line with workload demand. The dynamic life cycle of virtual servers can result in complexity if the process to manage the virtual servers is not automated with proper procedures. From an attack surface perspective, the virtual server (Windows, Solaris, or Linux) may be accessible to anyone on the Internet, so sufficient network access mitigation steps should be taken to restrict access to virtual instances. Typically, the CSP blocks all port access to virtual servers and recommends that customers use port 22 (Secure Shell or SSH) to administer virtual server instances. The cloud management API adds another layer of attack surface and must be included in the scope of securing virtual servers in the public cloud. Some of the new host security threats in the public IaaS include:

Stealing keys used to access and manage hosts (e.g., SSH private keys)

Attacking unpatched, vulnerable services listening on standard ports (e.g., FTP, NetBIOS, SSH)

Hijacking accounts that are not properly secured (i.e., weak or no passwords for standard accounts) Attacking systems that are not properly secured by host firewalls Deploying Trojans embedded in the software component in the VM or within the VM image (the OS) itself

**Securing virtual servers**

The simplicity of self-provisioning new virtual servers on an IaaS platform creates a risk that insecure virtual servers will be created. Secure-by-default configuration needs to be ensured by following or exceeding available industry baselines.

Securing the virtual server in the cloud requires strong operational security procedures coupled with automation of procedures. Here are some recommendations:

Use a secure-by-default configuration. Harden your image and use a standard hardened image for instantiating VMs (the guest OS) in a public cloud. A best practice for cloud- based applications is to build custom VM images that have only the capabilities and services necessary to support the application stack. Limiting the capabilities of the underlying application stack not only limits the host’s overall attack surface, but also greatly reduces the number of patches needed to keep that application stack secure.

Track the inventory of VM images and OS versions that are prepared for cloud hosting. The IaaS provider provides some of these VM images. When a virtual image from the IaaS provider is used it should undergo the same level of security verification and hardening for hosts within the enterprise. The best alternative is to provide your own image that conforms to the same security standards as internal trusted hosts.

Protect the integrity of the hardened image from unauthorized access.

Safeguard the private keys required to access hosts in the public cloud.

In general, isolate the decryption keys from the cloud where the data is hosted—unless they are necessary for decryption, and then only for the duration of an actual decryption activity. If your application requires a key to encrypt and decrypt for continuous data processing, it may not be possible to protect the key since it will be collocated with the application.

Include no authentication credentials in your virtualized images except for a key to decrypt the filesystem key.

Do not allow password-based authentication for shell access.

Require passwords for sudo\* or role-based access (e.g., Solaris, SELinux).

Run a host firewall and open only the minimum ports necessary to support the services on an instance.

Run only the required services and turn off the unused services (e.g., turn off FTP, print services, network file services, and database services if they are not required).

Install a host-based IDS such as OSSEC or Samhain.

Enable system auditing and event logging, and log the security events to a dedicated log server. Isolate the log server with higher security protection, including accessing controls.

If you suspect a compromise, shut down the instance, snapshot your block volumes, and back up the root filesystem. You can perform forensics on an uncompromised system later.

Institute a process for patching the images in the cloud—both offline and instantiated images.

Periodically review logs for suspicious activities. [Table 3-2](#_1idq7dh) lists security controls at the host level.

TABLE 3-2. Security controls at the host level

| Threat outlook | High |
| --- | --- |
| Preventive controls | Host firewall, access control, patching, hardening of system, strong authentication |
| Detective controls | Security event logs, host-based IDS/IPS |

**11.Identify the key points in implementing Infrastructure Security at the Application Level**

**Infrastructure Security: The Application Level**

Application or software security should be a critical element of your security program. Most enterprises with information security programs have yet to institute an application security program to address this realm. Designing and implementing applications targeted for deployment on a cloud platform will require that existing application security programs reevaluate current practices and standards. The application security spectrum ranges from standalone single-user applications to sophisticated multiuser e-commerce applications used by millions of users. Web applications such as content management systems (CMSs), wikis, portals, bulletin boards, and discussion forums are used by small and large organizations. A large number of organizations also develop and maintain custom-built web applications for their businesses using various web frameworks (PHP,† .NET,‡ J2EE,§ Ruby on Rails, Python, etc.). According to SANS, until 2007 few criminals attacked vulnerable websites because other attack vectors were more likely to lead to an advantage in unauthorized economic or information access. Increasingly, however, advances in cross-site scripting (XSS) and other attacks have demonstrated that criminals looking for financial gain can exploit vulnerabilities resulting from web programming errors as new ways to penetrate important organizations. In this section, we will limit our discussion to web application security: web applications in the cloud accessed by users with standard Internet browsers, such as Firefox, Internet Explorer, or Safari, from any computer connected to the Internet.

Since the browser has emerged as the end user client for accessing in-cloud applications, it is important for application security programs to include browser security into the scope of application security. Together they determine the strength of end-to-end cloud security that helps protect the confidentiality, integrity, and availability of the information processed by cloud services.

**Application-Level Security Threats**

According to SANS,‖ web application vulnerabilities in open source as well as custom-built applications accounted for almost half the total number of vulnerabilities discovered between November 2006 and October 2007.# The existing threats exploit well-known application vulnerabilities (e.g., the OWASP Top 10; see [http://www.owasp.org/index.php/Top\_10\_2007),](http://www.owasp.org/index.php/Top_10_2007)) including cross-site scripting (XSS), SQL injection, malicious file execution, and other vulnerabilities resulting from programming errors and design flaws. Armed with knowledge and tools, hackers are constantly scanning web applications (accessible from the Internet) for application vulnerabilities. They are then exploiting the vulnerabilities they discover for various illegal activities including financial fraud, intellectual property theft, converting trusted websites into malicious servers serving client-side exploits, and phishing scams. All web frameworks and all types of web applications are at risk of web application security defects, ranging from insufficient validation to application logic errors.

It has been a common practice to use a combination of perimeter security controls and network- and host-based access controls to protect web applications deployed in a tightly controlled environment, including corporate intranets and private clouds, from external hackers. Web applications built and deployed in a public cloud platform will be subjected to a high threat level, attacked, and potentially exploited by hackers to support fraudulent and illegal activities. In that threat model, web applications deployed in a public cloud (the SPI model) must be designed for an Internet threat model, and security must be embedded into the Software Development Life Cycle (SDLC); see [Figure 3-2](#_4fsjm0b).

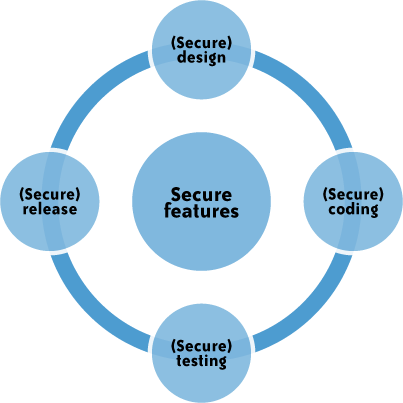


FIGURE 3-2. The SDLC

**DoS and EDoS**

Additionally, you should be cognizant of application-level DoS and DDoS attacks that can potentially disrupt cloud services for an extended time. These attacks typically originate from compromised computer systems attached to the Internet (routinely, hackers hijack and control computers infected by way of viruses/worms/malware and, in some cases, powerful unprotected servers). Application-level DoS attacks could manifest themselves as high-volume web page reloads, XML\* web services requests (over HTTP or HTTPS), or protocol-specific requests supported by a cloud service. Since these malicious requests blend with the legitimate traffic, it is extremely difficult to selectively filter the malicious traffic without impacting the service as a whole. For example, a DDoS attack on Twitter on August 6, 2009, brought the service down for several hours (see [Figure 3-3](#_3u2rp3q)).

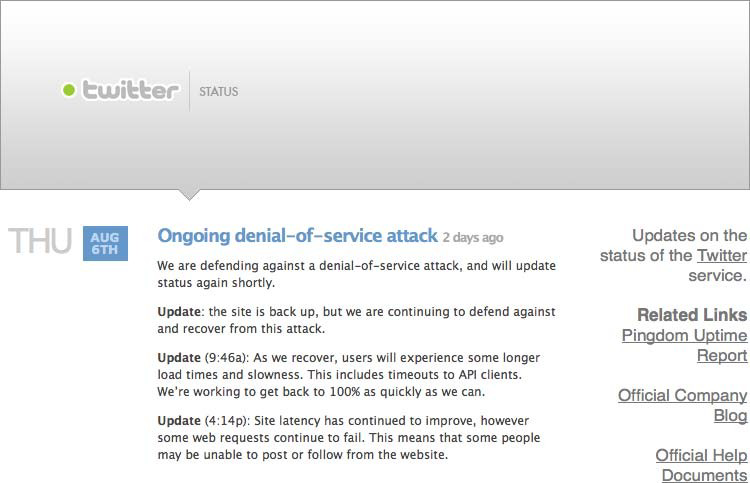


FIGURE 3-3. DDoS attack on Twitter

Apart from disrupting cloud services, resulting in poor user experience and service-level impacts, DoS attacks can quickly drain your company’s cloud services budget. DoS attacks on pay-as-you-go cloud applications will result in a dramatic increase in your cloud utility bill: you’ll see increased use of network bandwidth, CPU, and storage consumption. This type of attack is also being characterized as economic denial of sustainability (EDoS).†

The low barriers for small and medium-size enterprises to adopt cloud computing for legitimate use are also leveling the field for hackers. Using hijacked or exploited cloud accounts, hackers will be able to link together computing resources to achieve massive amounts of computing without any of the capital infrastructure costs. In the not-so-distant future, you might witness DoS attacks launched from IaaS or PaaS clouds against other cloud services (such hostile and offensive cloud models are being characterized as dark clouds).

**End User Security**

You, as a customer of a cloud service, are responsible for end user security tasks—security procedures to protect your Internet-connected PC—and for practicing “safe surfing.” Protection measures include use of security software, such as anti-malware, antivirus, personal firewalls, security patches, and IPS-type software on your Internet-connected computer. The new mantra of “the browser is your operating system” appropriately conveys the message thatbrowsers have become the ubiquitous “operating systems” for consuming cloud services. All Internet browsers routinely suffer from software vulnerabilities that make them vulnerable to end user security attacks. Hence, our recommendation is that cloud customers take appropriate steps to protect browsers from attacks. To achieve end-to-end security in a cloud, it is essential for customers to maintain good browser hygiene. The means keeping the browser (e.g., Internet Explorer, Firefox, Safari) patched and updated to mitigate threats related to browser vulnerabilities. Currently, although browser security add-ons are not commercially available, users are encouraged to frequently check their browser vendor’s website for security updates, use the auto-update feature, and install patches on a timely basis to maintain end user security.‡

**Who Is Responsible for Web Application Security in the Cloud?**

Depending on the cloud services delivery model (SPI) and service-level agreement (SLA), the scope of security responsibilities will fall on the shoulders of both the customer and the cloud provider. The key is to understand what your security responsibilities are versus those of the CSP. In that context, recent security surveys have highlighted the fact that lack of transparency in security controls and practices employed by CSPs is a barrier to cloud adoption.

To start with, cloud customers do not have the transparency required in the area of software vulnerabilities in cloud services. This prevents customers from managing the operational risk that might come with the vulnerabilities. Furthermore, by treating their software as proprietary, CSPs are impeding security researchers from analyzing the software for security flaws and bugs. (The exception is cloud providers that are operating on open source software.) Due to this lack of transparency, customers are left with no choice but to trust their CSPs to disclose any new vulnerability that may affect the confidentiality, integrity, or availability of their application. For example, as of March 2009, no prominent IaaS, PaaS, or SaaS vendors are participating in the Common Vulnerability and Exposures (CVE) project. Case in point: AWS took 7.5 months to fix a vulnerability that Colin Percival reported in May 2007.§ This vulnerability was a cryptographic weakness in Amazon’s request signing code that affected its database API (SimpleDB) and EC2 API services, and it was not made public until after it was fixed in December 2008. (Colin does acknowledge that Amazon took this issue seriously at all times, and the lengthy timeline was simply due to the large amount of work involved in rolling out a patch to the affected services.)

Enterprise customers should understand the vulnerability disclosure policy of cloud services and factor that into the CSP risk assessment. The following sections discuss the web application security in the context of the SPI cloud service delivery model.

**SaaS Application Security**

The SaaS model dictates that the provider manages the entire suite of applications delivered to users. Therefore, SaaS providers are largely responsible for securing the applications and components they offer to customers. Customers are usually responsible for operational security functions, including user and access management as supported by the provider. It is a common practice for prospective customers, usually under an NDA, to request information related to the provider’s security practices. This information should encompass design, architecture, development, black- and white-box application security testing, and release management.

Some customers go to the extent of hiring independent security vendors to perform penetration testing (black-box security testing) of SaaS applications (with consent from the provider) to gain assurance independently. However, penetration testing can be costly and not all providers agree to this type of verification.

Extra attention needs to be paid to the authentication and access control features offered by SaaS CSPs. Usually that is the only security control available to manage risk to information. Most services, including those from Salesforce.com and Google, offer a web-based administration user interface tool to manage authentication and access control of the application. Some SaaS applications, such as Google Apps, have built-in features that end users can invoke to assign read and write privileges to other users. However, the privilege management features may not be advanced, fine-grained access and could have weaknesses that may not conform to your organization’s access control standard. One example that captures this issue is the mechanism that Google Docs employs in handling images embedded in documents, as well as access privileges to older versions of a document. Evidently, embedded images stored in Google Docs are not protected in the same way that a document is protected with sharing controls. That means if you have shared a document containing embedded images, the other person will always be able to view those images even after you’ve stopped sharing the document. A blogger‖ discovered this access control quirk and brought it to Google’s attention. Although Google has acknowledged the issue, its response conveys that it believes# those concerns do not pose a significant security risk to its users.

Another incident related to Google Docs was a privacy glitch\* that inappropriately shared access to a small fraction (Google claims 0.05% of the documents were affected) of word processing and presentation documents stored on its Google Apps cloud service. Though the documents were shared only with people whom the Google Docs users had already shared documents, rather than with the world at large, the problem illustrates the need to evaluate and understand cloud-specific access control mechanisms.

Cloud customers should try to understand cloud-specific access control mechanisms— including support for strong authentication and privilege management based on user roles and functions—and take the steps necessary to protect information hosted in the cloud. Additional controls should be implemented to manage privileged access to the SaaS administration tool, and enforce segregation of duties to protect the application from insider threats. In line with security standard practices, customers should implement a strong password policy—one that forces users to choose strong passwords when authenticating to an application.†

It is a common practice for SaaS providers to commingle their customer data (structured and unstructured) in a single virtual data store and rely on data tagging to enforce isolation between customer data. In that multitenant data store model, where encryption may not be feasible due to key management and other design barriers, data is tagged and stored with a unique customer identifier. This unique data tag makes it possible for the business logic embedded in the application layer to enforce isolation between customers when the data is processed. It is conceivable that the application layer enforcing this isolation could become vulnerable during software upgrades by the CSP. Hence, customers should understand the virtual data store architecture and the preventive mechanisms the SaaS providers use to guarantee the compartmentalization and isolation required in a virtual multitenant environment.

Established SaaS providers, such as Salesforce.com, Microsoft, and Google, are known to invest in software security and practice security assurance as part of their SDLC. However, given that there is no industry standard to assess software security, it is almost impossible to benchmark providers against a baseline.‡

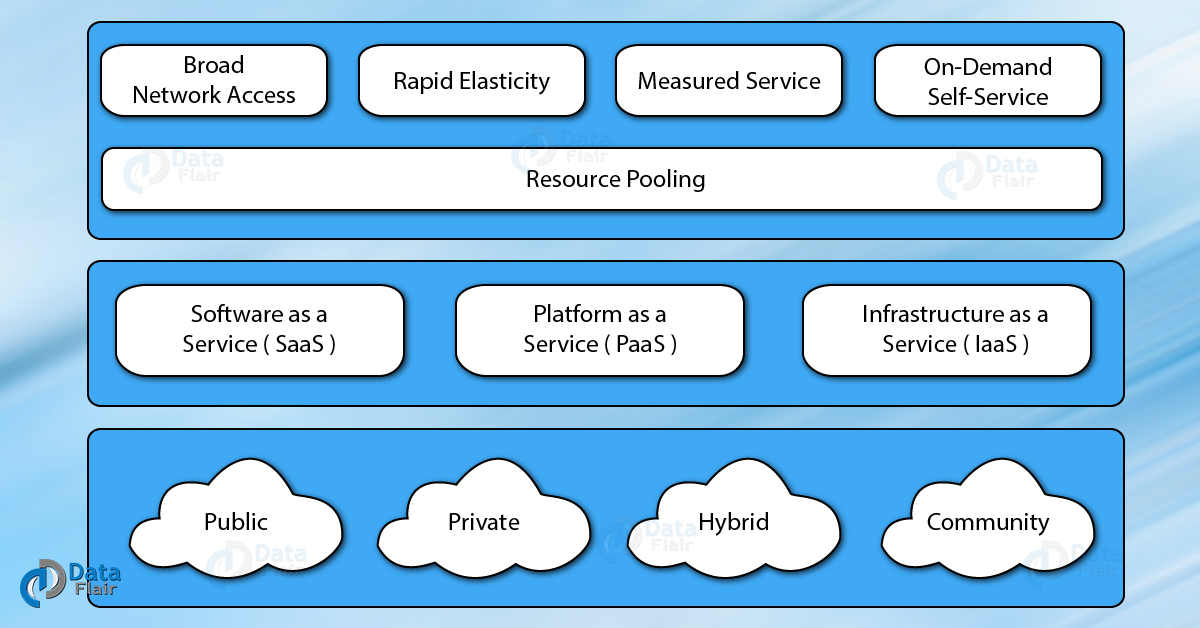
[Table 3-3](#_2eclud0) lists security controls at the application level.

TABLE 3-3. Security controls at the application level

| Threat outlook | Medium |
| --- | --- |
| Preventive controls | Identity management, access control assessment, browser hardened with latest patches, multifactor authentication via delegated authentication, endpoint security measures including antivirus and IPS |
| Detective controls | Login history and available reports from SaaS vendors |

**11.Explain various components of Cloud Cube Model with its architecture**

Cloud Cube Model, designed and developed by Jericho forum. Which helps to categorize the cloud network based on the four-dimensional factor: Internal/External, Proprietary/Open, De-Perimeterized/Perimeterized, and Insourced/Outsourced. Here, in this Cloud Cube Model tutorial, we will discuss a complete model with its dimensions and some questions.

So, let’s start the Cloud Cube Model.

**Cloud Cube Model – Introduction**

**1. What is Cloud Cube Model?**

Cloud Cube model, helps to categorize the cloud network based on the four-dimensional factor. Their main focus is to protect and secure the cloud network. This cloud cube model helps to select cloud formation for secure collaboration. This model helps IT managers, organizations, and business leaders by providing the secure and protected network.

Security is an important concern for cloud customers and most of the [cloud providers](https://data-flair.training/blogs/cloud-service-providers-companies/)understand it. The customer should also keep in mind, the selected cloud formation meets the regulatory and location requirements. They should also keep in mind that if cloud providers stop providing the services, where else they can move. There are three service models, which include:

* [Software as a Service (SaaS)](https://data-flair.training/blogs/software-as-a-service-saas/)
* Platform as a Service (PaaS)
* [Infrastructure as a Service (IaaS)](https://data-flair.training/blogs/infrastructure-as-a-service-iaas/)

In addition, there are four deployment models such as:

* [Public Cloud](https://data-flair.training/blogs/what-is-public-cloud/)
* Private Cloud
* [Community Cloud](https://data-flair.training/blogs/community-cloud-tutorial/)
* [Hybrid Cloud](https://data-flair.training/blogs/what-is-hybrid-cloud-computing/)

These models are very flexible Agile and responsible. They are user-friendly and provide many benefits to the customers.

**2. How to Secure Data in the Cloud Cube Model?**

There are some steps and points to keep in mind before securing your data in a cloud cube model:

**Step 1**

The classification of the data, the customer should know what rules must be applied to protect it.

**Step 2**

It should be ensured, the data exist only in specific trust levels.

**Step 3**

It should check that what regulatory compliance and restrictions are applicable. For example, the data should stay in a particular boundary and whether it has to stay in the safe harbour or not.

After the data is classified and is ready to put in the required zone, the assigned person is in a position to decide the following factors-

The data and processes, which are to be moved in the cloud.

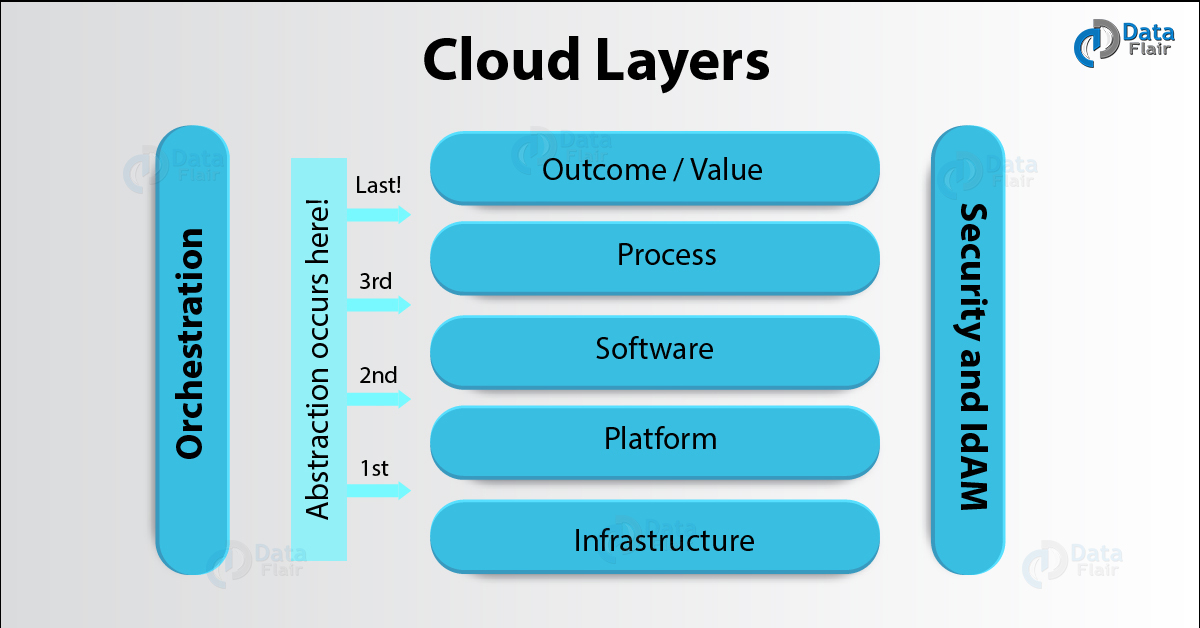
At what level the user wants to operate in the cloud. It can be infrastructure, platform, software, or [platform as a service](https://data-flair.training/blogs/platform-as-a-service-paas/).

The cloud formations, which are mostly compatible as per the requirement.

The level of [operation in the cloud](https://data-flair.training/blogs/cloud-operations-tutorial/) can be different as per the requirement.

Below is the chart which shows the Cloud layers, where the cloud operates.

After that, there are forms of cloud and the user can store the data which is mostly compatible with the company.



Cloud Cube Model – Cloud layers

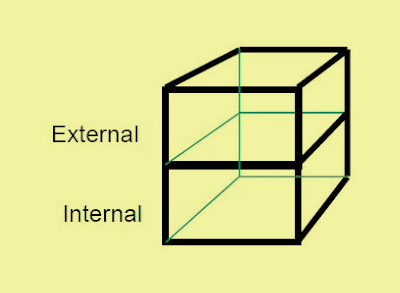
**3. Dimensions of Cloud Cube Model**

Cloud Cube model has four dimensions to categorized cloud formations:

* Internal/External
* Proprietary/Open
* De-Perimeterized/Perimeterized
* Insourced/Outsourced Dimension

**i. Internal/External**

The most basic cloud form is the external and internal cloud form. The external or internal dimension defines the physical location of the data. It acknowledges us whether the data exists inside or outside of your organization’s boundary. Here, the data which is stored using a [private cloud](https://data-flair.training/blogs/private-cloud-tutorial/) deployment will be considered internal and data outside the cloud will be considered external.



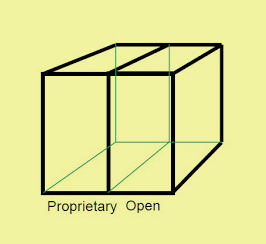
Cloud Cube Model – External/Internal

**ii. Proprietary/Open**

The second type of cloud formation is proprietary and open. The proprietary or open dimension states about the state of ownership of the cloud technology and interfaces. It also tells the degree of interoperability, while enabling data transportability between the system and other cloud forms.

The proprietary dimension means, that the organization providing the [service is securing](https://data-flair.training/blogs/cloud-security/) and protecting the data under their ownership.

The open dimension is using a technology in which there are more suppliers. Moreover, the user is not constrained in being able to share the data and collaborate with selected partners using the open technology.



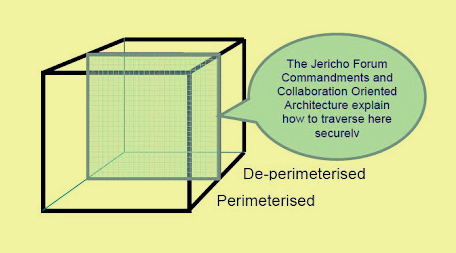
Cloud Cube Model – Proprietary/Open

**iii. De-Perimeterized/Perimeterized**

The third type of cloud formation is De-perimeterized and Perimeterized. To reach this form, the user needs collaboration oriented architecture and Jericho forum commandments.  
The Perimeterised and De-perimeterized dimension tells us whether you are operating inside your traditional it mindset or outside it.

Perimeterized dimension means, continuing to operate within the traditional it boundary, orphan signaled by network firewalls. With the help of VPN and operation of the virtual server in your own IP domain, the user can extend the organizations perimeter into external Cloud Computing domain. This means that the user is making use of the own services to control access.

De-perimeterized dimension means the system perimeter is architected on the principles outlined in the Jericho forums commandments. In De-perimeterized dimension, the data will be encapsulated with metadata and mechanisms, which will further help to protect the data and limit the inappropriate usage.

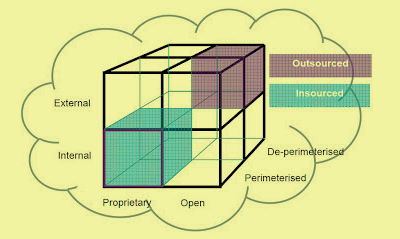


Cloud Cube Model – De-Perimeterized/Perimeterized

**iv. Insourced/Outsourced**

The Insourced and outsourced dimensions have two states in each of the eight cloud forms. In the outsourced dimension the services provided by the third party, whereas in the insourced dimension the services provided by the own staff under the control.

In this few organizations that are traditional bandwidth software or hardware, providers will run fluently on becoming cloud service providers. The organizations which are seeking to procedure cloud services must have the ability to set legally binding collaboration agreement. In this, an organization should ensure that data is deleted from the service provider’s Infrastructure.



Insourced/Outsourced Dimension

**12.Distinguish any 10 factors that leads to Cloud Computing threats**

**Analysis** I’ve seen different companies with operational models 90% based on cloud services, where the rest of the 10% is constituted of in-house servers. The basic response after asking about security issues related to cloud services was that the cloud service provider will take care of them and they don’t have to worry about it.

This isn’t necessarily the case with every cloud service provider, since some CSPs have a good security model in place, while others clearly do not. There are many advantages of cloud services, which is why the cloud service model is being used extensively, but they are out of scope of this article.

Before continuing, let’s quickly describe the difference between a threat and a vulnerability we’ll be using throughout the article:

**Vulnerability:** is a weakness that can be exploited by the attacker for his own personal gain. A weakness can be present in software, environments, systems, network, etc.

**Threat:** is an actor who wants to attack assets in the cloud at a particular time with a particular goal in mind, usually to inflict his own financial gain and consequentially financial loss of a customer.

Cloud computing vulnerabilities

When deciding to migrate to the cloud, we have to consider [the following cloud vulnerabilities:](http://www.academia.edu/4877213/Seven_Deadly_Threats_and_Vulnerabilities_in_Cloud_Computing)

**Session Riding:** Session riding happens when an attacker steals a user’s cookie to use the application in the name of the user. An attacker might also use CSRF attacks in order to trick the user into sending authenticated requests to arbitrary web sites to achieve various things.

**Virtual Machine Escape:** In virtualized environments, the physical servers run multiple virtual machines on top of hypervisors. An attacker can exploit a hypervisor remotely by using a vulnerability present in the hypervisor itself – such vulnerabilities are quite rare, but they do exist. Additionally, a virtual machine can escape from the virtualized sandbox environment and gain access to the hypervisor and consequentially all the virtual machines running on it.

**Reliability and Availability of Service:** We expect our cloud services and applications to always be available when we need them, which is one of the reasons for moving to the cloud. But this isn’t always the case, especially in bad weather with a lot of lightning where power outages are common. The CSPs have uninterrupted power supplies, but even those can sometimes fail, so we can’t rely on cloud services to be up and running 100% of the time. We have to take a little downtime into consideration, but that’s the same when running our own private cloud.

**Insecure Cryptography:** Cryptography algorithms usually require random number generators, which use unpredictable sources of information to generate actual random numbers, which is required to obtain a large entropy pool. If the random number generators are providing only a small entropy pool, the numbers can be brute forced. In client computers, the primary source of randomization is user mouse movement and key presses, but servers are mostly running without user interaction, which consequentially means lower number of randomization sources. Therefore the virtual machines must rely on the sources they have available, which could result in easily guessable numbers that don’t provide much entropy in cryptographic algorithms.

**Data Protection and Portability:** When choosing to switch the cloud service provider for a cheaper one, we have to address the problem of data movement and deletion. The old CSP has to delete all the data we stored in its data center to not leave the data lying around.

Alternatively, the CSP that goes out of the business needs to provide the data to the customers, so they can move to an alternate CSP after which the data needs to be deleted. What if the CSP goes out of business without providing the data? In such cases, it’s better to use a widely used CSP which has been around for a while, but in any case data backup is still in order.

**CSP Lock-in:** We have to choose a cloud provider that will allow us to easily move to another provider when needed. We don’t want to choose a CSP that will force us to use his own services, because sometimes we would like to use one CSP for one thing and the other CSP for something else.

**Internet Dependency:** By using the cloud services, we’re dependent upon the Internet connection, so if the Internet temporarily fails due to a lightning strike or ISP maintenance, the clients won’t be able to connect to the cloud services. Therefore, the business will slowly lose money, because the users won’t be able to use the service that’s required for the business operation. Not to mention the services that need to be available 24/7, like applications in a hospital, where human lives are at stake.

**Cloud computing threats**

Before deciding to migrate to the cloud, we have to look at the cloud security vulnerabilities and threats to determine whether the cloud service is worth the risk due to the many advantages it provides. The following are the top security threats in a cloud environment:

**Ease of Use:** The cloud services can easily be used by malicious attackers, since a registration process is very simple, because we only have to have a valid credit card. In some cases we can even pay for the cloud service by using PayPal, Western Union, Payza, Bitcoin, or Litecoin, in which cases we can stay totally anonymous. The cloud can be used maliciously for various purposes like spamming, malware distribution, botnet C&C servers, DDoS, password and hash cracking.

**Secure Data Transmission:** When transferring the data from clients to the cloud, the data needs to be transferred by using an encrypted secure communication channel like SSL/TLS. This prevents different attacks like MITM attacks, where the data could be stolen by an attacker intercepting our communication.

**Insecure APIs:** Various cloud services on the Internet are exposed by application programming interfaces. Since the APIs are accessible from anywhere on the Internet, malicious attackers can use them to compromise the [confidentiality and integrity](http://searchcloudsecurity.techtarget.com/tip/Develop-a-secure-API-design-in-a-cloud-environment) of the enterprise customers. An attacker gaining a token used by a customer to access the service through service API can use the same token to manipulate the customer’s data. Therefore it’s imperative that cloud services provide a secure API, rendering such attacks worthless.

**Malicious Insiders:** Employees working at cloud service provider could have complete access to the company resources. Therefore cloud service providers must have proper security measures in place to track employee actions like viewing a customer’s data. Since cloud service provides often don’t follow the best security guidelines and don’t implement a security policy, employees can gather confidential information from arbitrary customers without being detected.

**Shared Technology Issues:** The cloud service SaaS/PasS/IaaS providers use scalable infrastructure to support multiple tenants which share the underlying infrastructure. Directly on the hardware layer, there are hypervisors running multiple virtual machines, themselves running multiple applications.

On the highest layer, there are various attacks on the SaaS where an attacker is able to get access to the data of another application running in the same virtual machine. The same is true for the lowest layers, where hypervisors can be exploited from virtual machines to gain access to all VMs on the same server (example of such an attack is Red/Blue Pill). All layers of shared technology can be attacked to gain unauthorized access to data, like: CPU, RAM, hypervisors, applications, etc.

**Data Loss:** The data stored in the cloud could be lost due to the hard drive failure. A CSP could accidentally delete the data, an attacker might modify the data, etc. Therefore, the best way to protect against data loss is by having a proper data backup, which solves the data loss problems. Data loss can have catastrophic consequences to the business, which may result in a business bankruptcy, which is why keeping the data backed-up is always the best option.

**Data Breach:** When a virtual machine is able to access the data from another virtual machine on the same physical host, a data breach occurs – the problem is much more prevalent when the tenants of the two virtual machines are different customers. The side-channel attacks are valid attack vectors and need to be addressed in everyday situations. A side-channel attack occurs when a virtual machine can use a shared component like processor’s cache to access the data of another virtual machine running on the same physical host.

**Account/Service Hijacking:** It’s often the case that only a password is required to access our account in the cloud and manipulate the data, which is why the usage of two-factor authentication is preferred. Nevertheless, an attacker gaining access to our account can manipulate and change the data and therefore make the data untrustworthy. An attacker having access to the cloud virtual machine hosting our business website can include a malicious code into the web page to attack users visiting our web page – this is known as the watering hole attack. An attacker can also disrupt the service by turning off the web server serving our website, rendering it inaccessible.

**Unknown Risk Profile:** We have to take all security implications into account when moving to the cloud, including constant software security updates, monitoring networks with IDS/IPS systems, log monitoring, integrating SIEM into the network, etc. There might be multiple attacks that haven’t even been discovered yet, but they might prove to be highly threatening in the years to come.

**Denial of Service:** An attacker can issue a denial of service attack against the cloud service to render it inaccessible, therefore disrupting the service. There are a number of ways an attacker can disrupt the service in a virtualized cloud environment: by using all its CPU, RAM, disk space or network bandwidth.

**Lack of Understanding:** Enterprises are adopting the cloud services in every day operations, but it’s often the case they don’t really understand what they are getting into. When moving to the cloud there are different aspects we need to address, like understanding how the CSP operates, how the application is working, how to debug the application when something goes wrong, whether the data backups are already in place in case the hard drive dies, etc. If the CSP doesn’t provide additional backup of the data, but the customer expects it, who will be responsible when the hard drive fails? The customer will blame the CSP, but in reality it’s the customer’s fault, since they didn’t familiarize themselves enough with the cloud service operations – the result of which will be lost data.

**User Awareness:** The users of the cloud services should be educated regarding different attacks, because the weakest link is often the user itself. There are multiple social engineering attack vectors that an attacker might use to lure the victim into visiting a malicious web site, after which he can get access to the user’s computer. From there, he can observe user actions and view the same data the user is viewing, not to mention that he can steal user’s credentials to authenticate to the cloud service itself. [Security awareness](http://www.infosecinstitute.com/courses/security_awareness_training.html) is an often overlooked security concern.

**13.Summarize the core aspects that leads to Multi- Tenancy in cloud security**

**Multi Tenancy and Physical Security**

Multi-tenancy in cloud means sharing of resources and services to run software instances serving multiple consumers and client organizations (tenants). It means physical resources (such as computing, networking, storage) and services are shared, also the administrative functionality and support may also be shared. One of the big driver for providers is to reduce cost by sharing and reusing resources among tenants.

In a multi-tenant environment a lot more security dependence on the logical segregation (at multiple layers) rather than the physical separation of resources. Some of the cloud providers due to mult-tenancy may not allow audit and assessment by a particular tenant within their shared infrastructure.

**Security Risks**

Inadequate Logical Security Controls: Physical resources (CPU, networking, storage/databases, application stack) are shared between multiple tenants. That means dependence on logical segregation and other controls to ensure that one tenant deliberately or inadvertently can not interfere with the security ( confidentiality, integrity, availability) of the other tenants.

**Malicious or Ignorant Tenants:** If the provider has weaker logical controls between tenants, a malicious or an ignorant tenant may reduce the security posture of other tenants.

**Shared Services can become single point of failure:** If the provider has not architected well the common services, they can easily become single point of failure due to misuse or abuse by a tenant.

**Uncoordinated Change Controls and Mis configurations:** When multiple tenants are sharing the underlying infrastructure all changes needs to be well coordinated and tested .

**Co-mingled Tenant Data :** To reduce cost providers may be storing the data from multiple tenants in same database table-spaces and backup tapes. Data destruction can become a challenge in multi-tenancy especially if data is stored in the shared media (databases, backups, archives).

**Performance Risks :**One tenant’s heavy use of the service may impact the quality of service provided to other tenants.

**XaaS Specific Risks**

**SaaS:** Multiple clients (tenants) may be sharing the same application stack ( database, app/web servers, networking). That means the data from multiple tenants may get stored in the same database, may get backed up and archived together, may be moving on common networking devices (unencrypted), and managed by common application processes. This puts a heavy emphasis on logical security built within the application to separate one tenant's users from others.

**PaaS:** Platform stack is shared among the tenants. Vulnerability in the platform stack can allow bleeding between tenants. Shared data backups and archives.

**IaaS:** Cross Virtual machine attacks. Cross network traffic listening. Co-residents with lower security posture, where they are less concerned about keeping their hosts hardened and patched [5]. Especially when these hosts gets compromised and owned by the attackers.

**Countermeasures**

**Architecting for Multi-Tenancy :** Providers need to architect for multi-tenants, rather than making services that are not designed for multi-tenancy to work. Multi-tenancy architecture should take into account logical segregation, strengthen common services and single point of failures. Also provide more transparency to consumers/tenants [12].

**Data Encryption:** For encrypting the data and keeping it separate from other tenants , strong encryption complemented by tenant-owned key management is required. In a virtualized environment, this means that encryption can be done granularly on a per-VM basis, with key management owned by the tenant and not the service provider [1].

**Controlled Change Management :** eployment of the changes (especially to common shared services) should be well planned . Usage of feathered release cycles to migrate tenants to new infrastructure. For SaaS providers tenants should be progressively migrated to newer underlying infrastructure. (Providers need to plan extra resources for these activities). Providers should have a dependency mapping of tenants to underlying resources and services. So that any change in the underlying resources can be well planned.

**Transparency/Audit-ability of Administrative Access:** Tenants should have knowledge on administrative access to all their resources/services. One of the way is to have audit-ability of admin access enabled at all the layers of stack (OS, networking, applications , databases) that can be auditable by the tenants. Provider may still be doing the administration but under strong auditable environment.

**Virtual Private Cloud (VPC) :** It is a private cloud existing within a shared or public cloud. A VPC is a way to partition a public cloud (multi-tenancy) into quarantined virtual infrastructure [3] and link it back to the tenants internal resources by encrypted network links.

**Third Party Assessments:** Alternate assessment options or contractual exceptions if the auditing is required as per the consumer's security posture but provider does not allow it [6].

**Tenant Isolation :** Tenants can always negotiate or demand from the cloud provider to have their own separate physical infrastructure, databases, storage, networking gears,.. . Isolation does play a great role in the security field. However, it does increase the cost for tenants/clients.

**14.Identify the key features which leads to Cloud Benefits and assist in Business scenarios**

**Nine scenarios where cloud-based systems can benefit your company's security**



In our previous post, we shared our overview of cloud security, and the benefits and limitations of this emerging technology. You may be wondering whether now is the right time to consider adopting cloud-based security systems. Cloud security isn't necessarily the best choice for every business, and many security managers may be reluctant to adopt less proven security technologies. However, if your business is affected by any of the following scenarios, your security operations may benefit from a cloud-based system, either as a limited extension or a full upgrade.

**Your company has many small branches in areas where IT infrastructure is poorly developed**

Many growing enterprises encounter the problem of setting up branch offices in areas where IT infrastructure is not of high enough quality to meet your business and security needs. In this scenario, power failures may cause servers to become damaged and lead to loss of data. One of the greatest advantages of cloud-based systems is that servers are located in world class data centers in stable environments that are geographically dispersed, which isolates systems from natural disasters in on particular location.

**Your security system that doesn't need to be active 24/7 or doesn't require real time monitoring**

There are many companies for which active security systems are only required at certain times of day or week. In this scenario, users may be paying for systems that they barely use. Cloud-based systems solve this problem by offering "pay as you go" payment models. Many cloud-based security services also allow users to put servers online by themselves when they are needed.

**You have multiple sub-branches but an internal enterprise network has not been set up**

In this situation, end users may be required to by independent servers and network video recorders for multiple small sites. This is an expensive prospect, and means that regional security operations can not be easily managed by a central security. Cloud-based security systems can help end users avoid these challenges by providing public hosts which can be shared by different sub-branches. These hosts can be expanded and remotely managed as organizations grow and establish new sites. To ensure data security, firewalls and VPN services should also be implemented.

**Your security system shares a server with other programs or systems**

For many organizations, a dedicated server for a security system is not a cost effective option. Yet sharing a server with other systems often increases the probability of server crashes that can be highly disruptive to security and other business operations. Cloud computing allows security users to enjoy the benefits of a dedicated offsite server for security without the prohibitive costs incurred by an onsite server. With cloud-based security, companies pay according to their level of use and enjoy dedicated computing resources for security.

**Security maintenance for small and remote sites is difficult**

For organizations with a number of small, geographically dispersed sites, security maintenance can be a big challenge. Cloud-based security systems can make this easier for security teams by allowing them to configure systems remotely, perform system diagnostics quickly and to back up data regularly via the cloud platform.

**System servers and network video recorders don't have backup machines**

Many large enterprises rely on servers and network video recorders to manage security operations. But even for large companies, the cost of purchasing devices to backup data can be daunting. Cloud computing provides security users with a temporary backup host which is especially useful for disaster recovery and ensuring normal operations during site system maintenance.

**You have a temporary office that doesn't justify a large investment in security**

Global companies increasingly rely on temporary offices to give them greater business flexibility. Yet it is difficult to secure temporary sites using traditional server-based systems without considerable initial costs. Because cloud-based systems are pay as you go they are especially well-suited to temporary office sites. Using cloud-based systems, owners can eliminate the initial costs required for physical servers, operating system software, racks, network switches, UPS, air conditioners and other devices.

**Servers are not located near to end users and retrieving data takes a long time**

During a security incident, data needs to be instantly retrievable by security teams, but is often inconvenient to access when servers are in a remote location. Cloud computing systems allow users to retrieve data more quickly through "content delivery networks (CDN)". CDNs are largely distributed collections of servers found in multiple data centers across the internet that are designed to make content instantly available to users by providing increased bandwidth access. CDNs can therefore help security end users to view data streams like security video feeds much more easily than through remote servers.

**You have a system that requires complex configurations and a high maintenance workload**

The time costs, manpower and IT proficiency required to install, configure and maintain complex security systems are a drain on the resources of many organizations. With cloud-based systems, users can avoid this by  by creating what is known as a "cloud server image backup". This means that users can copy all the data in their server and create another identical server stored on the cloud. This new copied server can be allocated to a new site quickly and easily, without the need to install, configure and maintain an onsite server for each of your sites. What is more, all system upgrades are the responsibility of the cloud service provider, which makes security software upgrades one less issue for your IT team to worry about.

**PREPARED BY**

**SRINIVASAN.L**

**SAP/ISE/NHCE**