Carnegie Endowment for International Peace

Report Part Title: What Is the Cloud?

Report Title: Cloud Security:
Report Subtitle: A Primer for Policymakers
Report Author(s): Tim Maurer and Garrett Hinck
Published by: Carnegie Endowment for International Peace (2020)
Stable URL: https://www.jstor.org/stable/resrep25787.7

This paper is a first step to building that understanding. As a primer for policymakers on the cloud, this study outlines how to conceptualize the cloud and describes the evolution of the cloud market. It then discusses cloud security in detail, using a timeline of past incidents together with in-depth case studies of the most significant incidents that are publicly known. Together, these serve as a foundation for developing a comprehensive framework for mapping the various risks and a severity schema to prioritize them. The paper then briefly outlines additional public policy issues to take into account while considering cloud security. Finally, it sums up and discusses the implications for public policy, while listing promising areas for future security-related research.

## Chapter 1: What Is the Cloud?

At its most basic level, the cloud is simply someone else's more powerful computer that does work for others. There is no one single cloud—so while it might be accurate to say that data crosses the internet, it is not correct to say that such data is stored in an ephemeral form, hovering somewhere in the sky. In fact, the cloud stores and transports data across a global infrastructure of data centers and networks. A more accurate description of the cloud is that cloud services are an abstraction of a parallel system of computers, data centers, cables, infrastructure, and networks that provides the power to run modern enterprises' and organizations' digital operations and to store their data. Building the necessary infrastructure for cloud services on a truly global scale has been one of the most significant architectural achievements of the past decade—and it mostly exists behind the scenes, out of common knowledge. With that said, as chapter 2 highlights, the cloud marketplace has evolved significantly over the years, as has the cloud itself.

To make sense of the transformative impact of cloud services, first consider how computing, for example, worked prior to widespread cloud adoption. In the past few decades, for every computational task that a company or individual needed to do, they had to have their own computers, servers, and even data centers. For instance, Capital One, a major company in the financial services sector, announced in 2015 that it would move all of its apps to the AWS cloud, meaning that it subsequently did not have to build and buy data center storage as it rolled out new apps.[16] For smaller businesses, the costs of information technology (IT) procurement—that is, buying all the necessary computers and setting up the necessary networking for inhouse data storage and processing capabilities—were prohibitive to rapid growth.

When companies like Amazon, Google, and Microsoft began to offer storage and computing power as services in the late 2000s, they changed this paradigm. These massive IT giants could manage networks of data centers, servers, and networking at global scales—meaning they could take advantage of economies of scale to offer computing as a service—at prices that would beat internal costs for most companies and still make them a profit, especially after significant price drops starting in 2014.

Amazon, Google, and Microsoft particularly focus on providing the basic elements of IT infrastructure—server space and computing power—that are highly scalable, custom-configurable, and capable of being rapidly deployed and shifted. However, cloud computing encompasses a wide range of service types in which different firms predominate (see chapter 2), and the services provided include the basic infrastructure to build digital platforms on top of ready-made applications delivered over the internet. These various services can be grouped into the three principal types of cloud services. In practice, the major CSPs offer different services spanning all three of these categories.[17]

- Infrastructure as a service (IaaS): CSPs provide basic access to storage, networking, servers, or other computing resources.

- Platform as a service (PaaS): CSPs provide an environment—a platform—for customers to build and deliver applications.

- Software as a service (SaaS): CSPs build, run, and host applications delivered over the internet, which customers pay to access.[18]

## Defining Cloud Computing

Given the particular importance of cloud computing as a service, it is worth considering a 2011 definition of cloud computing by the U.S. Department of Commerce's National Institute of Standards and Technology (NIST)—the agency that sets technology standards. According to NIST, "cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (. . . networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."[19] Essentially, this definition touches on five key characteristics of cloud computing: 1) on-demand self-service, 2) rapid elasticity, 3) measured service, 4) broad network access, and 5) resource pooling.[20]

The first, on-demand self-service, means that customers can use capabilities only when needed and don't have to pay more. They are also siloed from each other even while making use of the same resources. Customers can select computing capabilities automatically—without needing any human support from the CSPs they use. And on the CSP side, on-demand self-service means that any customer request can be handled automatically. No technician has to go configure a server when a customer selects additional computing capacity. Automatic digital systems handle the allocation, provisioning, and deployment of the needed infrastructure, platforms, and services.

Second, rapid elasticity means that the amount of resources dedicated to any one customer can at any time quickly increase or decrease depending on the needs of the customer. Because the resource capacity of a CSP is exponentially larger than the likely needs of any one customer, customers can scale their operations rapidly without taxing the CSP.

Third, measured service captures how CSPs manage and price their services. CSPs, like AWS and Microsoft Azure, charge customers for the resources they use on a unit-per-time basis— these units are an abstraction of the resources used. For instance, AWS's Elastic Compute Cloud measures its service in units that AWS defines in terms of standard central processing unit (CPU) integer processing power.[21]

Fourth, broad network access means that customers access these services over the network, including potentially the public internet. This is a straightforward but highly important point from a security point of view. No longer are computing resources solely part of a firm's internal network—instead, in many cases, the core operating systems rely on connections that could be open to the entire global internet. In this respect, both the capacity and security of these network connections are critical. Even private cloud solutions, where the cloud servers are accessed over a private connection, would require significant bandwidth.

Fifth, resource pooling means that a CSP combines its resources such that each customer shares the same infrastructure with other customers in a dynamic fashion, to be apportioned and reapportioned as necessary. This feature is what makes cloud computing a more efficient model than separate computing resources for each firm. CSPs can take advantage of economies of scale to build infrastructure and platforms at mass scale—and share these resources among multiple customers at the same time to save costs and unneeded capacity.

## Underlying Technologies

While modern CSPs rely on highly complex systems for allocating, managing, and deploying resources among millions of customers, three key technologies are essential for understanding how cloud services work at scale: virtualization, hypervisors, and containerization.

**Virtualization** allows for abstraction between physical hardware and individual computers. Essentially, virtualization allows for multiple computers, referred to as virtual machines, to exist on the same physical server. Beyond computational tasks and storage, entire networks can be built through virtualization.

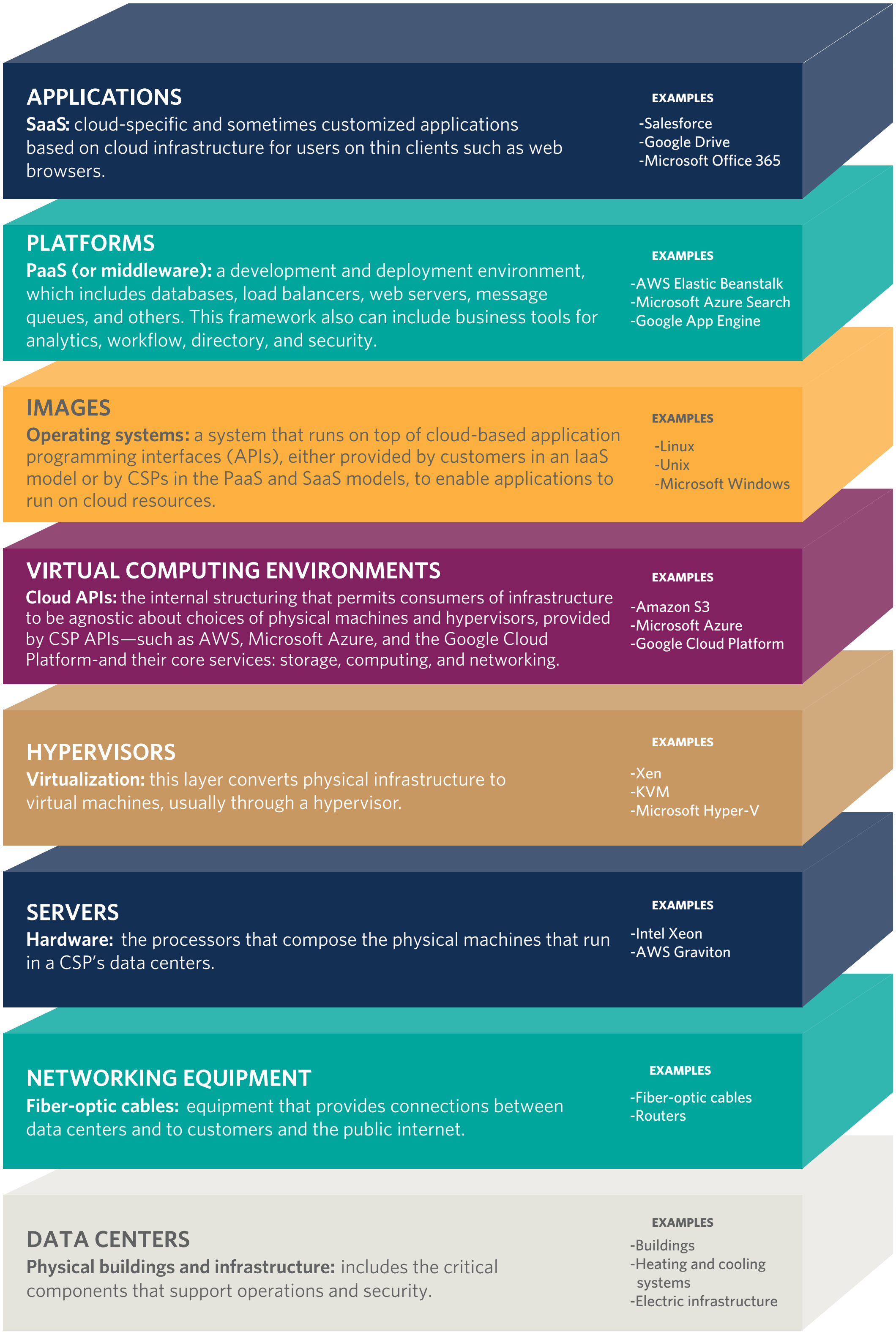
**Hypervisors** are programs that manage virtual machines, servers, the connection between those virtual machines and servers, and the allocation of resources to the virtual machines. Thus, it becomes possible to have a whole bank of physical servers, each running a hypervisor, and then create virtual machines across this bank of servers. CSPs refer to the virtual machines they create for their customers as instances, since they only last as long as needed and, once they are no longer needed, they are spun down to free up capacity.

**Containerization** is a refinement of virtualization that works by running discrete containers within the same operating system, basically moving up the abstraction provided by virtualization by one level. Containerization caught on around 2014 with the introduction of a new tool called Docker, which made it much more convenient and efficient to implement containerization for business uses. While a virtual machine includes an entirely virtual operating system, a container is an isolated environment within one single operating system. In terms of layers, while a hypervisor lies between the hardware and virtual machines, each with their own operating system inside them, a container sits on top of an operating system that is on top of the container engine, and then the hardware is below.

In table 1, the infrastructure for the cloud is visualized according to a layers model, similar to the Open Systems Interconnection model for the internet itself.[22] The table presents a hierarchy of layers from the physical data centers at the bottom to the entirely virtual application layer at the top, allowing the various parts of the cloud to be simplified. The descriptions for each layer present the key technologies operating at that level, along with several examples.

TABLE 1

**Visualizing Cloud Architecture**



**APPLICATIONS**

**SaaS:** cloud-specific and sometimes customized applications based on cloud infrastructure for users on thin clients such as web browsers.

EXAMPLES

-Salesforce
-Google Drive
-Microsoft Office 365

**PLATFORMS**

**PaaS (or middleware):** a development and deployment environment, which includes databases, load balancers, web servers, message queues, and others. This framework also can include business tools for analytics, workflow, directory, and security.

EXAMPLES

-AWS Elastic Beanstalk
-Microsoft Azure Search
-Google App Engine

**IMAGES**

**Operating systems:** a system that runs on top of cloud-based application programming interfaces (APIs), either provided by customers in an IaaS model or by CSPs in the PaaS and SaaS models, to enable applications to run on cloud resources.

EXAMPLES

-Linux
-Unix
-Microsoft Windows

**VIRTUAL COMPUTING ENVIRONMENTS**

**Cloud APIs:** the internal structuring that permits consumers of infrastructure to be agnostic about choices of physical machines and hypervisors, provided by CSP APIs—such as AWS, Microsoft Azure, and the Google Cloud Platform-and their core services: storage, computing, and networking.

EXAMPLES

-Amazon S3
-Microsoft Azure
-Google Cloud Platform

**HYPERVISORS**

**Virtualization:** this layer converts physical infrastructure to virtual machines, usually through a hypervisor.

EXAMPLES

-Xen
-KVM
-Microsoft Hyper-V

**SERVERS**

**Hardware:** the processors that compose the physical machines that run in a CSP's data centers.

EXAMPLES

-Intel Xeon
-AWS Graviton

**NETWORKING EQUIPMENT**

**Fiber-optic cables:** equipment that provides connections between data centers and to customers and the public internet.

EXAMPLES

-Fiber-optic cables
-Routers

**DATA CENTERS**

**Physical buildings and infrastructure:** includes the critical components that support operations and security.

EXAMPLES

-Buildings
-Heating and cooling systems
-Electric infrastructure

### Cloud Deployment Models

CSPs also offer their services in three main deployment models: the public cloud, a private cloud, or a hybrid cloud.

In the **public cloud**, customers share the same infrastructure available to be rented out to the public. A CSP manages the infrastructure and allows prospective customers to purchase resources. The customer uses the same CSP-allocated infrastructure as other customers.

A **private cloud** arrangement is designed for a single customer, essentially housing resources on premises or off premises but still isolated from other customers. An organization may set up its own private cloud, or it may contract with a CSP to do this.

Some organizations choose a **hybrid cloud**, in which they combine a public cloud service from a CSP with either a private cloud setup or a more traditional data center so they can communicate and share data and applications.[23] Some organizations opt for this arrangement because they have sensitive data that they consider too risky to store in a public-cloud-only environment, but they still want to take advantage of the public cloud's computing power to run applications.[24]

Customers may choose a deployment model based on their particular needs, including whether the data and services they are migrating to the cloud are especially critical or sensitive. For instance, a customer with sensitive personal data to manage (like a healthcare provider) might choose a private cloud to minimize the risks that their data may be exposed to their CSPs' other clients. Increasingly, hybrid cloud options are predominating, as customers realize that different types of data require different levels of security.

## Chapter 2: The Origins and Evolution of the Cloud and Its Market

The concept of cloud services predates the internet itself. The history of these services illustrates how computing has moved from the core to the edge of a network and back, as technology evolves. An early example is the mainframe computer model produced by IBM in the 1950s and 1960s, which were the most powerful machines at the time. Computing was organized around the core of these mainframe computers used by large government and industry leaders. Access to these computers was organized on a "time-share" model allowing multiple users to share the resources of one computer.[25]