Cloud Computing

*Resources used:*

*DionTraining UDemy Course - “Cloud Computing and Virtualization”*

*Professor Messer on Youtube - “Cloud Models” and “Cloud Characteristics”*

*Comptia A+ Exam Textbook*

*ChatGPT*

The technology at the heart of these innovative services is virtualization. We use the servers and networks of the cloud through layers of software that add great value to the underlying hardware by making it simple to perform complex tasks or manage powerful hardware. As end users we generally interact with just the sweet software icing of the service-layer cake.

Vocabulary

1. High Availability: services experience very little downtime when using the cloud.
   1. Most services built on the cloud were built to be highly reliable and very fault tolerant = ensure high availability
   2. Usually measured in percentages; the gold standard in the industry is 99.99% - only 5 minutes and 15 seconds of downtime in a year! Wow!
   3. Organizations can also design their cloud deployments for high availability—for example, they can set up redundant servers all around the world and automatically reroute traffic whenever a server is down.
2. Scalability: ability to increase the number of items in a system at a linear rate or less than a linear rate
   1. Vertical Scaling (scaling up)
      1. Increasing the power of the existing resources in the working environment
         1. Add more resources to a particular server or node
            1. Ex: cloud server with 2 virtual CPU’s, you can scale up to 4 CPU’s (could also be more RAM, more storage, etc).
   2. Horizontal Scaling (scaling out)
      1. Adding additional resources to help handle the extra load being experienced.
         1. Basically, instead of providing a status boost to 1 server, like vertical scaling, you would simply add more characters to your party! 1 server? No, more like 3 servers now! Power of friendship!
3. Rapid Elasticity: ability to quickly scale up or down; the system’s ability to handle changes to demand in real time (huge plus/pro of cloud services!)
   1. Etc; being able to handle 100 users, and then 1000 users within minutes with ease
4. Metered Utilization: Being charged for a serve on a pay per use basis (kind of like how our water bill or electricity bill works). Benefit of using the cloud is that most things are done on a metered basis.
5. Shared Resources: Ability to minimize the costs by putting VMs on other servers.
   1. Examples: AWS, Microsoft Azure, Google Cloud
   2. Instead of renting out each physical unit separately, cloud providers aggregate them into a pool of shared resources that they make available on-demand. The provider’s customers draw from this pool as they need additional resources and release them back into the pool when they are done.
6. File Synchronization: Ability to store data which can then spread to other places depending on configuration
   1. Synchronization apps make it easy to access the same set of files across multiple devices.
   2. File synchronization makes it easier for users to collaborate or work on the same files from multiple devices without having to shuffle files around.

Cloud Deployment Models

1. Public: Systems and users interact with devices on public networks, such as the Internet and other clouds
   1. Examples: Google Drive; public cloud services can be offered via pay per use model or for free (like Google Drive!)
2. Private: Systems and users only have access with other devices inside the same private cloud or system
   1. Company makes its own cloud environment that only it can utilize as an internal enterprise resource; usually done for security reasons (we don’t want the public to get their grubby fingers on our work stuff, y’know?)
   2. Example: AWS GovCloud (US)
3. Hybrid: Combination of private and public clouds
   1. Enterprise will utilize their own cloud environment and systems, but also utilize public services or outsource services to another service provider (like how Fairlead outsources their printer servers).
   2. Strict rules need to be implemented d/t security reasons
   3. Sometimes we can have our cake and eat it too. Not all data is crucial, and not every document is a secret. Needs that an organization can only meet in-house might be less important than keeping an application running when demand exceeds what it can handle onsite. We can build a hybrid cloud by connecting some combination of public, private, and community clouds, allowing communication between them. Using a hybrid cloud model can mean not having to maintain a private cloud powerful enough to meet peak demand—an application can grow into a public cloud instead of grinding to a halt, a technique called cloud bursting.
4. Community: Collaborative effort where infrastructure is shared between several different organizations with common service needs
5. Multitenancy: Allows customers to share computing resources in a public or private cloud.
   1. Literally in the name: multiple tenants within a single server or a set server; most companies or entities won’t utilize the full capacities of a single server so why not divide up the server and offer it to several users?
      1. However, the downside of this is that if one of the other customers gets attacked, then y’all getting attacked. One of y’all effed, all y’all effed.
      2. Data in multi tenancy solutions may be exposed to other servers
6. Single Tenancy: Assigns a particular resource to a single organization
   1. More expensive because it requires more hardware to run it properly, but more secure.

Cloud Service Models

1. On-Premise Solution: Needs to procure hardware, software, and personnel necessary to run the organization’s cloud
   1. Allows ability to control all the physical and logical access to servers, which is good when you want to keep your data SECURE and CONFIDENTIAL
   2. BUT IT'S ALL ON YOU DAWG
2. Hosted Solution: Third-party service provider that provides all the hardware and facilities needed to maintain a cloud solution
   1. Amazon, Microsoft and Google all provide hosted solutions for organizations to utilize
3. SaaS, Software as a Service: Provides your organization a “complete solution”
   1. HARDWARE, NETWORKING, STORAGE, SERVERS, VIRTUALIZATION, OS, MIDDLEWARE, RUNTIME, DATA, APPLICATION
   2. Examples: Microsoft 365, google docs, turbotax, etc
   3. No development work required, no on-going maintenance required on your end.
4. PaaS, Platform as a Service: up to you to handle to application code and data processing between your client and your severs
   1. HARDWARE, NETWORKING, STORAGE, SERVERS, VIRTUALIZATION, OS, MIDDLEWARE, RUNTIME
   2. Gives programmers all the tools they need to deploy, administer, and maintain a Web application. The PaaS provider starts with some form of infrastructure, which could be provided by an IaaS, and on top of that infrastructure the provider builds a platform: a complete deployment and management system to handle every aspect of a Web application.
   3. As far as the programmer is concerned, the PaaS is just a place to deploy and run his or her application.
   4. Organization only needs the infrastructure, hardware and network to deploy their software instances, then IaaS is the way to go. You will be responsible for implementation and management and security.
5. IaaS, Infrastructure as a Service: Large-scale global Infrastructure as a Service (IaaS) providers use hardware virtualization to minimize idle hardware, protect against data loss and downtime, and respond to spikes in demand.
   1. The beauty of IaaS is that you no longer need to purchase expensive, heavy hardware.
   2. The hitch is that, while we’re no longer responsible for the hardware, we are still responsible for configuring and maintaining the operating system and software of any virtual machines we create. This can mean we have a lot of flexibility to tune it for our needs, but it also requires knowledge of the underlying OS and time to manage it.

Virtual Desktop Infrastructure (VDI): Host desktop OSs within a virtualized environment hosted by a centralized server or server farm

* Separates the personal computing environment from a user’s physical computer - the end user is then able to access the virtual desktop from a thin client (a computer that uses resources housed inside a central server as opposed to a hard drive) or through a web browser. Then, they can interact with that virtualized desktop as if they were sitting right there.
  + “Everytime I try to run a command, it doesn’t process it on my local machine, it processes it on that cloud server.”

1. Centralized Model: hosts all the desktop instances on a single server or server farm
2. Hosted Model/Desktop as a Service (DaaS): Maintained by a service provider and provided to the end user as a service
   1. Desktops are maintained by provided and given to the end user as a service
      1. Examples: Amazon Workspaces, VMware Horizon Air, and Citrix XenDesktop

Cloud Storage Services

1. Cloud Storage Applications: amount of space on a cloud-based server as file storage
   1. DropBox, google drive, etc…
2. File Synchronization: Ability to synchronize from different devices using a single account
   1. A Lot of cloud storage apps have the ability to synchronize from your different devices and be able to send data to and from all of your different connected devices using a single account
3. Content Delivery Networks: Network or servers that locates the nearest server to minimize delay or download time
   1. ESSENTIALLY, there are several different servers located around the world,s o that whenever that user is going from, it will find the closest server near them and this helps minimize the delay or download time to get those files. It’s like server ping pong.
      1. Less latency! Minimized downtime! Things CDN helps with

Software-Defined Networking (SDN): Enables the network to be intelligently and centrally controlled, or programmed using software applications

1. Application Layer
   1. Focuses on the communication resource requests or information about the network as a whole
2. Control Layer
   1. Uses the information from the applications and decides how to route a data packet on the network
3. Infrastructure Layer
   1. Contains the network devices that receive information about where to move the data
4. Management Plane
   1. Used to monitor traffic conditions and the status of the network

Questions to ask a Host Provider…

* Authentication and authorization mechanisms
* Redundancy and fault tolerance measures
* Storage location and location-based laws

**Infrastructure as a Service (IaaS):**

IaaS provides virtualized computing resources over the internet. Users typically rent virtual machines, storage, and networking infrastructure from a cloud provider.

Users have control over the operating systems, applications, and middleware running on the provided infrastructure.

Examples of IaaS providers include Amazon Web Services (AWS) EC2, Microsoft Azure Virtual Machines, and Google Compute Engine.

**Platform as a Service (PaaS):**

PaaS offers a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining the underlying infrastructure.

PaaS typically includes development tools, middleware, databases, and other services necessary for application development and deployment.

Users focus on writing and deploying code, while the PaaS provider handles scalability, security, and maintenance of the underlying infrastructure.

Examples of PaaS offerings include Google App Engine, Microsoft Azure App Service, and Heroku.

**Software as a Service (SaaS):**

SaaS delivers applications over the internet on a subscription basis, eliminating the need for users to install, maintain, and update software locally.

Users access SaaS applications through a web browser or API, usually paying a recurring fee based on usage or number of users.

The provider manages all aspects of software delivery, including infrastructure, maintenance, updates, and security.

Examples of SaaS products include Salesforce CRM, Google Workspace (formerly G Suite), and Microsoft Office 365.

In summary, IaaS provides infrastructure resources, PaaS offers a platform for developing and deploying applications, and SaaS delivers fully functional software applications over the internet. The level of abstraction and management responsibilities increase from IaaS to SaaS, with users having more control over infrastructure in IaaS and less control in SaaS.