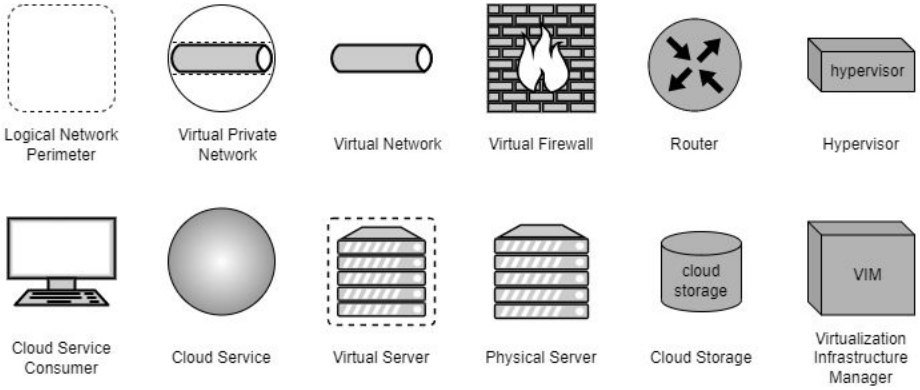
CS4650 Topic 23:
Cloud Infrastructure Mechanisms

Cloud Infrastructure Mechanisms

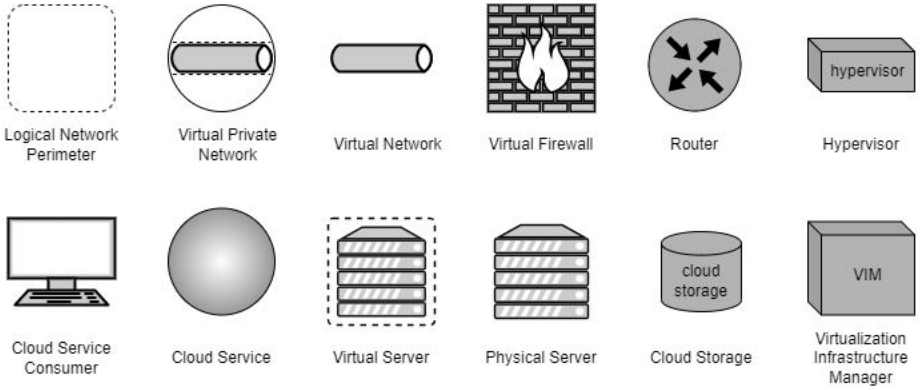
- Cloud Infrastructure Mechanisms are building blocks of cloud environments.
- These are common to many cloud platforms.

Symboligy



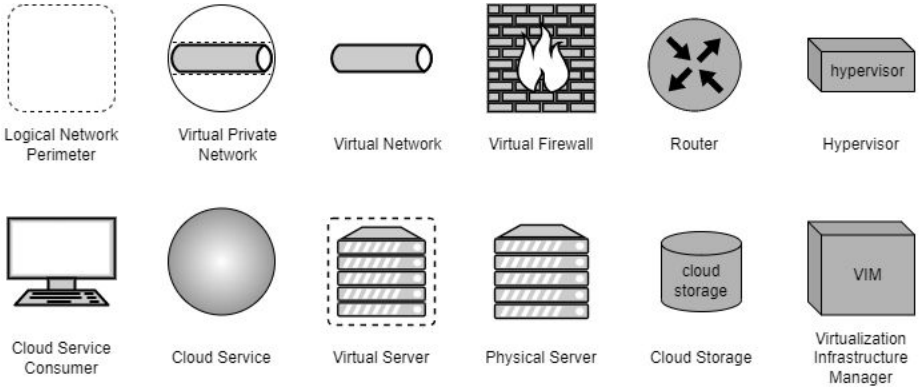
- *Logical Network Perimeter:* Isolating a portion of a larger network. Communication between IT resources within this perimeter is more open; communicating with resources outside the perimeter is more restricted.
- *Virtual Private Network*: Provides a secure link between physically isolated logical network perimeters, essentially expanding the perimeter to include distant resources.

Symboligy



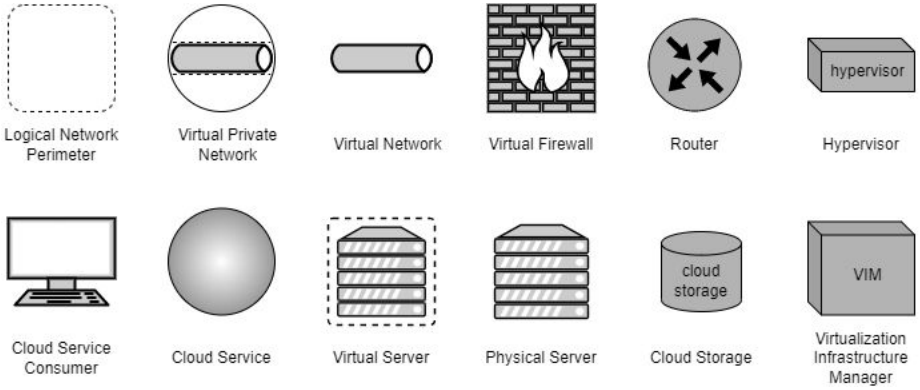
- *Virtual Network:* Represents the portion of the network inside the logical network perimeter.
- *Virtual Firewall:* Used to restrict (block) access from nodes outside the logical network perimeter.
- *Router*: These represent the routers of the Internet, which are the nodes that pass messages from one network to another.
- *Hypervisor:* The 'operating system' that hosts the virtual machines on a physical machine.

Symboligy



- *Cloud Service Consumer:* The application or user that is using a cloud service.
- *Cloud Service:* An application that runs in the cloud.
- *Virtual Server:* An environment (either operating system or applications) that run on top of a hypervisor on a physical server. The Virtual Server acts as if it were its own physical server.
- *Physical Server:* The actual hardware computer (or other IT device) that can host a hypervisor and one or more virtual servers.

Symboligy

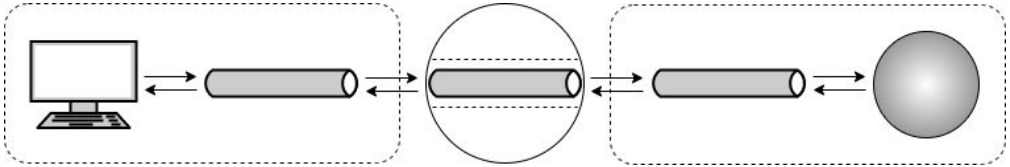


- *Cloud Storage:* A shared storage device on the cloud.
- *Virtual Image (not shown):* The data that implements a virtual server. Virtual Images are stored on a Cloud Storage device, and are copied to a Physical Server when an instance of a Virtual Server is deployed.
- *Virtualization Infrastructure Manager (VIM):* A software package that monitors a collection of hypervisors and virtual servers. The VIM decides when to deploy a virtual server, when to migrate (move) a virtual server, and verifies that the servers are running.

Logical Network Perimeter

- A *Logical Network Perimeter* isolates, to some degree, a portion of a network.
  - In many respects, the portion of the network that is inside the perimeter acts as a network separate from the outside network.
- A logical network perimeter might be used to:
  - Isolate IT resources in a cloud from unauthorized users.
  - Isolate IT resources in a cloud from non-users
  - Isolate IT resources in a cloud from cloud consumers
  - Control the bandwidth that is available to isolated IT resources

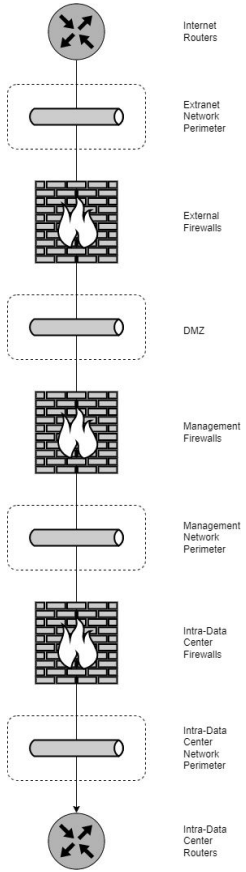
Logical Network Perimeter



- Here is one scenario:
  - The left shows a logical network perimeter for the cloud service consumer. This is a protected network, providing security and privacy for the users.
  - The right shows a logical network perimeter for the cloud service, providing security and privacy for the service.
  - Connecting these two is a virtual private network. In essence, the consumer has a special connection to the service. Its as if the two were on a single protected network.
  - The packets between the two are encrypted by the VPN.

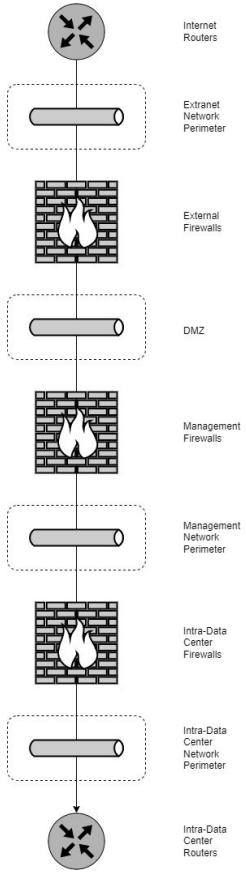
Case Study: DTGOV

- One of the case studies in the book is DTGOV, and this is a map of the segmentation and isolation of their data centers.
- The first step is that the external routers connect to a virtual network.
  - The only way into the extranet network perimeter is through these routers.
  - Being a virtual network, the external network is abstracted.
  - The devices connected to this network are loosely isolated and are protected from external users.
  - No cloud consumer IT resources are in this network.



Case Study: DTGOV

- The second step are the external firewalls, leading to the next virtual network, called the DMZ (demilitarized zone).
  - The DMZ is also a virtual network.
  - This network hosts the proxy servers, which are the intermediaries for commonly used network services, such as DNS, e-mail, and web portals and servers.
- One wonders why this uses a virtual network, rather than an actual physical network, especially because physical networks are faster and maybe more efficient?



Digression: Why a Virtual Network?

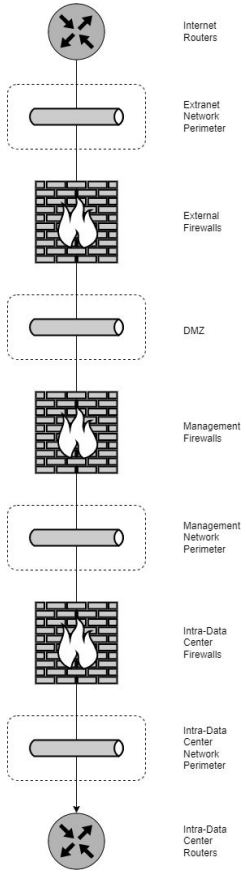
- Virtual networks provide the same resource sharing advantages of a physical network.
- When high bandwidth networking throughput is not required, then paying for the physical network is an unnecessary expense. Instead, the virtual network can be sharing the same hardware as other virtual systems, such as storage or processing.
- Virtual networks are easy to define, they reduce hardware investments, and eliminate dependencies on hardware. They are easy to control from a central point.

Digression: Why a Virtual Network?

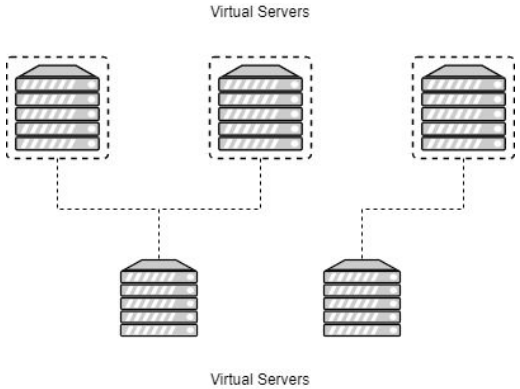
- Consolidating servers in a virtual network reduces the overhead of traditional networking components.
- By defining a virtual network within one processor, there is no network traffic outside the processor. Hence:
  - There is a high degree of network availability.
  - Security is improved.
  - Performance is improved.
- Not mentioned in the book: If a hacker compromises a physical system, it is hacked. For a virtual system, the system is repaired simply by restarting with a fresh virtual image. This restarting can be performed on a regular basis.

Case Study: DTGOV

- The network traffic leaving the proxy servers passes through management firewalls that isolate the management network perimeter.
  - The management network hosts the servers providing the bulk of the management services that the cloud consumers can externally access.
  - These services are provided in direct support of self-service and on-demand allocation of cloud-based IT resources.
- Traffic to the cloud=based IT resources flows through the management layer, then through the intra-data center firewalls, to finally reach the data center network, and the cloud based resources.

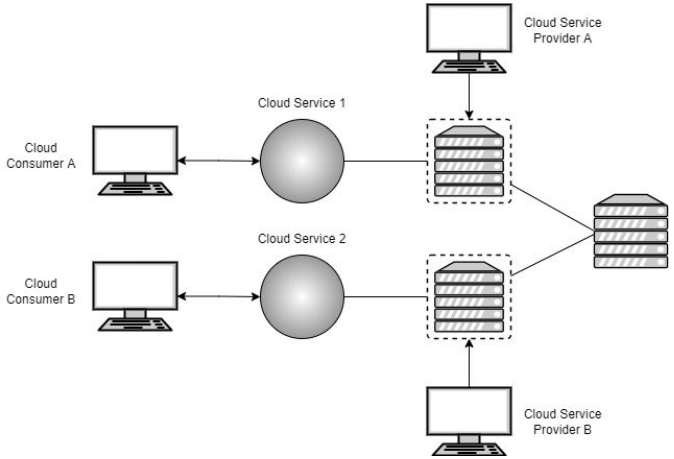


Virtual Servers



- A *virtual server* is a virtual image that emulates a physical server.
- A virtual server runs as a virtual machine on a hypervisor, which runs on a physical machine.
- A hypervisor can run multiple virtual servers at the same time.
- Virtual servers can be added on-demand.
- 'Spinning up' a new instance of a virtual server can be quite fast, usually much faster than starting a physical machine.

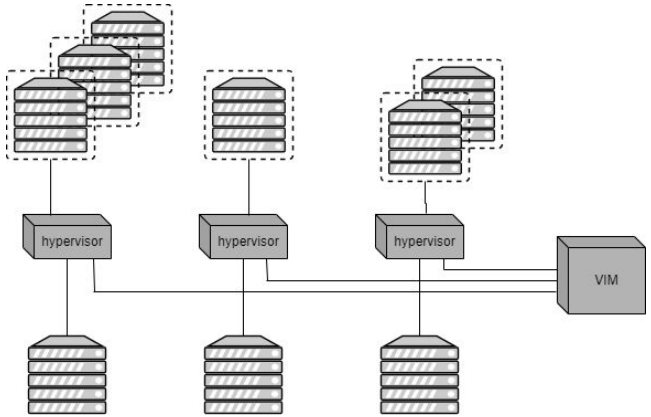
Virtual Servers



- A virtual server is the most fundamental building block of a cloud environment.
- Each virtual server can host numerous IT resources, cloud-based solutions, and various other cloud computing mechanisms.
- Each cloud consumer can customize their environments independently from other cloud consumers that may be sharing the same physical machine.
- If they were customizing the actual physical hardware, then the settings used by one consumer may conflict with the settings of other consumers.

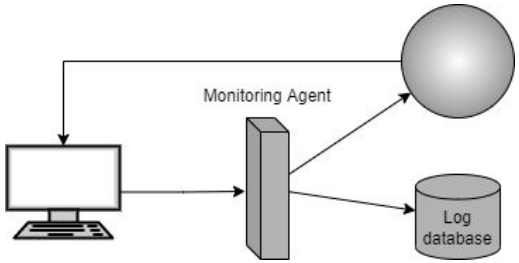
Virtualization Infrastructure Management (VIM)

- Usually, a cluster of physical servers are part of the cloud's implementation.
- Each of these physical servers has a hypervisor (the 'operating system' sort of).
- Each of the virtual servers is hosted by a hypervisor.
- To manage this ensemble, there is a VIM.
- The VIM monitors and controls the operation of the hypervisors.
    - If new virtual servers are required, the VIM chooses which physical machine, then directs that hypervisor to add a new virtual server.
    - The VIM can also direct that a virtual server be migrated from one hypervisor to another.



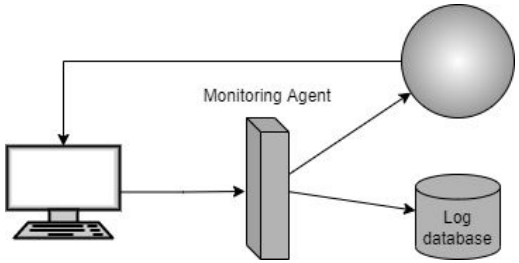
Cloud Usage Monitor

- A *cloud usage monitoring* mechanism is a lightweight and autonomous software program.
- It collects and processes IT resource usage data.
- There are several types of monitors.
- In the example shown, the monitor intercepts all messages to the cloud service.
- The monitor records relevant usage information to the log database, then passes the message on to the actual cloud service.
- The reply message goes directly to the cloud consumer.



Cloud Usage Monitor

- Other forms of monitor:
  - Directly receive messagesfrom the cloud service, then record to the database.
  - Occasionally poll the cloud service, asking what's up, then recording this to the database.
  - Rather than tap into the cloud service, tap into the VIM.
    - This monitor might record data into the database.
    - This monitor might report back to the VIM, resulting in a change to the configuration.
    - The monitor might do both.

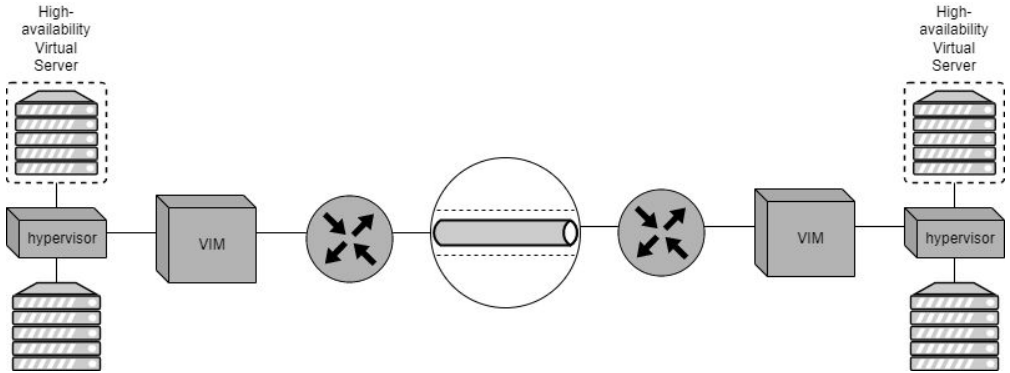


Resource Replication

- One monitor that taps into the VIM is a Resource Replicator.
- This monitor checks the loads on the Virtual Machines.
- If a virtual machine is being overloaded, and if replication would fit within the parameters of the SLA, then the VIM would be instructed to replicate that virtual machine.
- Resource 'unreplication' is a similar process: when the load reduces, the steps are reversed, and a virtual machine is released.

High-Availability Virtual Server

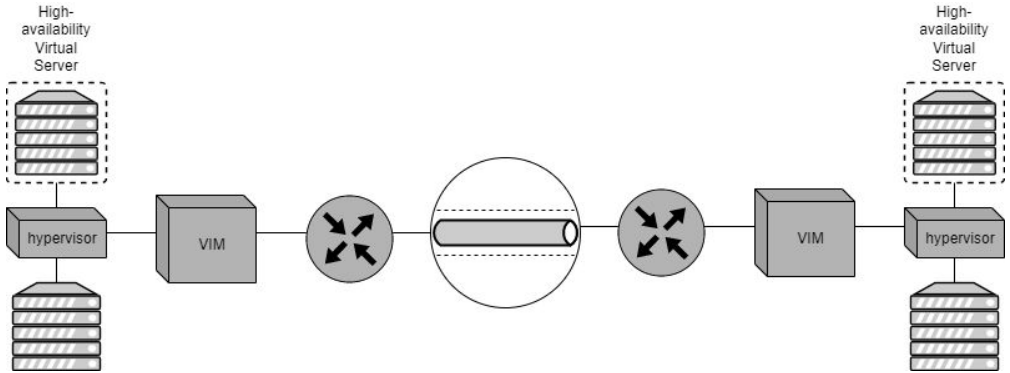
- A high-availability virtual server may be running in one data center.
- The VIM in that data center is working in coordination with a VIM in another data center.



- If the physical machine in the first data center becomes unavailable, the server, the hypervisor and the VIM cease to run.

High-Availability Virtual Server

- When the first VIM ceases, the second VIM notices.
- The second VIM then starts another instance of the high-availability server in the second data center.



- At some point the hardware in the first data center is repaired. At this point, there are two options:
  - Use the first data center as the fail-over backup for the second data center.
  - Immediately migrate the Virtual Server to the first data center, then release the copy from the second.

Load Balancer

- A *Load Balancer* is a monitor that seeks to increase performance and capacity by shifting work from one virtual server to another, or by shifting work from one instance of a cloud service to another.
- There are several algorithms or plans:
  - Simple division of labor.
  - *Asymmetric Distribution*, where larger workloads are issued to resources with higher capacity.
  - *Workload Prioritization*, where workloads are distributed according to their priority level.
  - *Content-Aware Distribution*, where requests are distributed to different resources as dictated by the request content.
  - A *round robin* approach to distribute traffic evenly (although the *load* might not be even!)