Virtualization and Cloud

NIC (Network Interface Card), the server has three layers:1- H.W. components. 2- OS. 3- Apps as Email/Web server ,… etc, has a one to one relationship between hardware and OS.
Virtualization allows to break the one to one relationship allowing to run multiple OS on a single physical server, each instance is called a VM, a hypervisor (VMM→ Virtual Machine Monitor) is used to manage and allocate the hardware resources to each VM.
A hypervisor running directly on top of hardware is called type 1/ bare metal/ native hypervisor, usually used in data center environments.
Type 2 / hosted hypervisor: runs as a program on the OS as a regular computer program, the OS running directly on hardware is called Host OS, OS running in VM is called Guest OS, usually used for personal use.
Virtualization: provides partitioning where multiple OS can run on the same physical machine, dividing system resources among them so there is no hardware under use
Provides fault and security isolation at hardware level, provides encapsulation as the entire state of VM can be saved to files, moved and copied easily.
Provides hardware independence as long as the hardware can run the hypervisor.
VMs are connected to each other and the external network via a virtual switch running on the hypervisor, that can operate access or trunk ports and use VLANs to separate VMS at layer 2, interfaces on vSwitch connect to the physical NIC(s) of the server to communicate with the external network, a vPC (virtual port channel) can be used in the connection between the NIC and switches to form a port channel to two separate physical switched for redundancy.

Traditional IT infrastructure deployments were combination of: 1- On-Premises: all servers,network devices and other infrastructure are located on company property.2- Colocation: data centers that rent out space for customers to put their infrastructure.
Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.
Five essential characteristics of cloud computing:
1- On demand self service: a consumer can unilaterally provision computing capabilities, such as server time, network storage as needed automatically without requiring human interaction with each service provider.
2-Broad network access. Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations), the service could be accesses through the internet or private WAN connections.
3- Resource pooling. The provider’s computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, and network bandwidth.
4- Rapid elasticity. Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
5-Measured service. Cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.
In cloud computing everything is provided as a service model.

Service Models: 1- Software as a Service (SaaS). The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure2. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user- specific application configuration settings.

2- Platform as a Service (PaaS). The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

Infrastructure as a Service (IaaS). The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls).

Deployment Models: 1- Private cloud. The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

2- Community cloud. The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of them, and it may exist on or off premises, the least common cloud deployment.

3- Public cloud. The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider, the most common cloud deployment.

4- Hybrid cloud. The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

Benefits: cost – glocal scale – speed/agility – productivity – reliability.

Enterprise network can connect to cloud resources via a private WAN service provider, the internet, a VPN… etc, connections are preferred to be redundant.

Containers are software packages containing an App (Multiple Apps) and all dependencies for contained app to run, run on a container engine that runs on a host OS usually Linux, lightweight including only the dependencies required to run the specific App.

A container orchestrator: a software platform for automating the deployment, management, scaling,.. etc for containers, for large-scale systems with micro services and thousands of container.

Microservice Architecture: an approach to software architecture that divides a larger solution into smaller parts (micro services).

VMS can take minutes to boot up while containers take milliseconds, Vms take more disk space, resources, containers are more portable for example Docker containers can be run on any container service, VMS are more isolated because each run its own OS, while containers are less isolated as they share the same OS.

Virtual Routing and Forwarding (VRF): to divide a physical router into multiple virtual routers, each with their own routing table.

It allows the router to build multiple separate routing tables, Layer 3 interfaces only add router interfaces , SVIs and routed ports can be configured in a VRF, interfaces and routes are configured to be in a specific VRF (VRF instance).
Traffic in one VRF can't be forwarded out of an interface in another VRF, as an exception VRF leaking can allow traffic to pass between VRFs.
VRF is commonly used in MLPS, yet VRF lite is used without MLPS.
VRF is commonly used by service providers to allow one device to carry traffic from multiple customers, where each customer's traffic is isolated from the others and customer IP addresses can overlap without issues.
Without using VRF, two interfaces in the same router can't be in the same subnet.
Command: ip vrf <name> to create vrf. **Command: show ip vrf.**
Command: ip vrf forwarding <vrf name> to assign vrf to an interface in interface config mode (the ip address of interface will be removed on assigning a vrf).
Show ip route displays the global routing table.
Command: show ip route vrf <vrf name>.
If an interface is not in a vrf its routes will be in a global routing table and will be isolated from other vrfs.
Command: ping vrf <vrf name> ip.
Router's interfaces are in separate broadcast domains by default.