Virtualization

Virtualization is the creation of a virtual resource. Now, I know that's pretty broad, but that's because virtualization itself is a broad category. We can virtualize anything. This includes servers, desktops, file systems, hard drives, and even an entire network.
A virtual machine is a container that contains an emulated computer that can run an entire operating system inside of it. This includes emulation of all of the hardware that's required to run the system. This means you have the hard drives, the optical drives, video cards, processors, and even the BIOS being emulated.

Two main types of virtual machines:

1. System Virtual Machines
2. Processor Virtual Machines

A system virtual machine is a complete platform that's designed to take the place of an entire computer. That means you can run the entire operating system virtually.

A processor virtual machine, on the other hand, is designed to run a single application. Often, this is used to run something like a web browser or possibly even a simple web server.

Hypervisors

A hypervisor may adjust the distribution of the physical resources of the server to the virtual machines. This includes the processor, the memory, and the hard disk space.

Hypervisors come in two distinct flavors, Type 1 and Type 2.

A Type 1 hypervisor is known as bare metal, or native, since it runs directly on the host hardware and functions as a type of operating system. Microsoft's Hyper-V, Citrix's XenServer, and VMWare's ESXi and vSphere are all considered Type 1 hypervisors.

A Type 2 hypervisor runs from within a normal operating system, something like Windows, Mac, or Linux.

But there is a third type of virtualization that's becoming popular in our networks today.
This is called Application Container-Based Virtualization. With this type of virtualization, the operating system kernel is shared across multiple virtual machines, but the user space for each of these virtual machines is uniquely created and managed. Often called Application Containerization, this allows an organization to deploy and run distributed applications without launching a resource-heavy, full virtual machine with a full operating system.

Container Virtualization is commonly used with Linux servers, and some examples of Container-Based Virtualization software include things like Docker, Parallels Virtuozzo, and the OpenVZ project.

Threats to VMs

VM escape

Virtual machine escape, or VM escape, occurs when an attacker is able to break out of one of these normally isolated virtual machines and they can begin to interact directly with the underlying hypervisor. From this position, the attacker could migrate themselves out, and into another virtual machine being hosted on the same physical server. Now, VM escape techniques are extremely difficult to conduct. They rely on exploiting the physical resources that are shared between the VMs.

Data Remnants

When a server is scaled up, a new virtual instance is created on a physical server. This instance takes up some hard drive space for all those files that represent the virtual hard disk and the configurations. When this is no longer needed because the load has decreased, the virtual machine can be deprovisioned, which means it's shut down and the files are deleted. When this occurs, the confidential files from that virtual machine are left on the physical server. This is known as a data remnant.

These data remnants could be recovered by an attacker, and therefore, it could breach the confidentiality of that data.

Privilege elevation

Privilege elevation occurs when a user is able to grant themselves the ability to run functions as a higher-level user, such as the root or the administrator. While this can be bad on a single server, it can be catastrophic on a physical server if the attacker is able to perform this on the hypervisor itself.