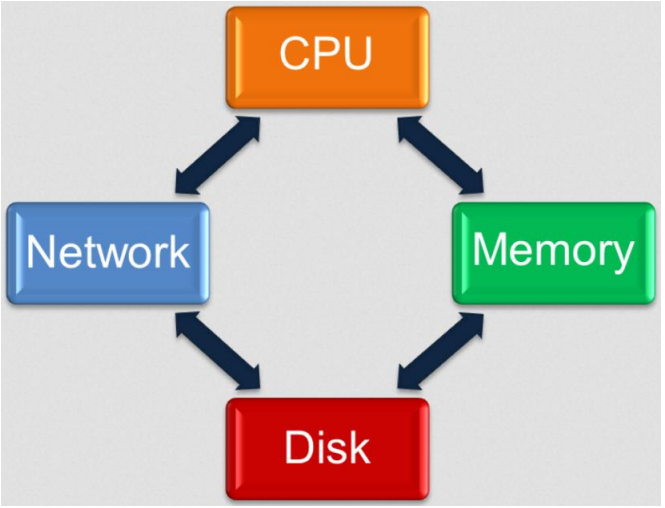
# Performance Tuning

Ed Crowley

*Ch8*

# Building a virtualization host

- Proper host computer resources planning ensures that the host can deliver the performance needed for virtualization support .
- Compute resources include:
  - Disk
  - Processor
  - Memory
  - Networking



# Quotas and Limits

- Cloud providers must protect limited compute resources and make certain that their customers only have access to the amount for which they contracted .

Two methods:

1. Quotas
2. Limits.

- Limits are a defined floor, or ceiling, on the amount of resources that can be used.
- Quotas are limits that are defined for a system on the total amount of resources that can be utilized.

# Hard or Soft Limits

- A hard limit is the maximum amount of resources that can be utilized.
  - For example, a hard limit of 100 Gigabytes (GB) for a storage partition will not allow anything to be added to that partition once it reaches 100 GB.
    - Will log an event or notify the user.
- A soft limit allows the user to save a file even if the drive reaches 100 GB.
  - Will still log an alert and notify the user.

# Quotas

- Have to do with allocation of the host compute resources to its guest machines.
- Are established according to service level agreements (SLAs).
  - Created between the provider and their customers to indicate a specific level of capacity.
- Capacity management is essentially the practice of allocating the correct amount of resources required to deliver a business service.
- The resources that these quotas enforce limits upon may be physical disks, disk arrays, host bus adapters, RAM chips, physical processors, and network adapters.
  - Allocated from the total resources available to individual guests based on their SLA.

# Licensing

- Organization needs to identify their virtualization software vendor has own licensing.
- Some vendors have a free version of their product.
  - Only require a license for advanced feature sets that enable functionality.,
- Others offer a completely free virtualization platform.
  - But might not offer some of the more advanced features with their product.

# Licenses

- Before deploying a virtualization host and choosing a virtualization vendor, the organization should understand the license agreements and determine:
  - Needed features
  - How features are licensed.
- Both the virtualization host and virtual machine require software licenses.

# Reservations

- Work similarly to quotas.
- Ensure that a lower limit is enforced for the amount of resources guaranteed to a cloud consumer for their virtual machine or machines.
- Reservations ensure certain virtual machines always have a defined baseline level of resources available to them regardless of the demands placed on them by other virtual machines.

# Resource Pools

- Slices or portions of compute resources, CPU, memory, and storage.
  - Can be partitioned order to provide different levels of resources to specific groups or organization.
  - Can be nested within a hierarchy for organizational alignment.
- Provide a flexible mechanism with which to organize the sum total of the compute resources in a virtual environment and link them back to their underlying physical resources.

# Virtual Machine Resource Allocation

- Guest virtual machines should be configured for its intended application or task.
  - In addition to CPUs and memory, a virtual machine may require higher-priority access to certain storage or disk types.
- Consider:
  - Virtual machine role
  - Machine load
  - Number of clients
  - Ongoing monitoring and assessment.
- Amount of disk space the virtual machine is using also needs monitoring.

# Compute Resources

- Virtual machine compute resources are still made up of disk, network, processor, and memory components.
  - But these components are made available to virtual machines as abstractions of physical components presented by a hypervisor that emulates those physical resources for the virtual machine.
- With virtual machines, BIOS is emulated by the hypervisor.
- When the BIOS is emulated and these physical resources are abstracted, administrators have the ability to divide the virtual compute resources from their physical providers and distribute those subdivided resources across multiple virtual machines.
  - Ability to subdivide physical resources is one of the key elements that make cloud computing and virtualization so powerful.

# Resource Requirements

- When splitting resources among multiple virtual machines, vendor-specific algorithms can help the hypervisor make resource decisions..
  - Host resources required for these activities, include small amounts of processor, memory, and disk.
- Determinations factors include:
  - Defined quotas and limits,
  - Which resource is requested by which virtual machine,
  - Business logic that may be applied by a management system for either a virtual machine or a pool of virtual machines, and
  - Resources available at the time of the request.

# Possibilities

- Possible for the processing power required to make these decisions to outweigh the benefits.
  - In those situations, administrators can configure their systems to allocate specific resources or blocks of resources to specific hosts.
- CPU affinity is one such application, in which processes or threads from a specific virtual machine are tied to a specific processor or core, and all subsequent requests from that process or thread are executed by that same processor or core.
- Reservations can be used to guarantee an amount of compute resources for that virtual machine.

# Quotas and Limits

- Virtual machines utilize quotas and limits to constrain user's access to compute resources.
  - Prevent users from either completely depleting or monopolizing those resources.
- Quotas can be  either hard or soft.
- Hard quotas set limits that users and applications cannot exceed.
  - If an attempt to use resources beyond the set limit is registered, the request is rejected, and an alert is logged.
- The soft quota difference is that the request is granted instead of rejected, and the resources are made available to service the request.
  - Same alert is still logged.

# Licensing

- Successfully managing software license agreements in a virtual environment is tricky.
  - Software application must support licensing a virtual instance of the application.
  - Some software vendors still require the use of a dongle or a hardware key when licensing their software.
- Others have adopted their licensing agreements to coexist with a virtual environment.
- Some vendors have moved to a per-CPU-core type of license agreement to adapt to virtualization.

# Physical Resource Redirection

- There are times when it is useful to have a virtual machine connect its virtual serial port to a host physical serial port.
  - For example, a user might want to install an external modem or another form of a handheld device on the virtual machine, and this would require the virtual machine to use a physical serial port on the host computer.
- It might also be useful to connect a virtual serial port to a file on a host computer and then have the virtual machine send output to a file on the host computer.
  - An example of this would be to send data that was captured from a program running on the virtual machine via the virtual serial port and transfer the information from the guest to the host computer.

# Virtual Ports

- In addition to using a virtual serial port, it is also helpful in certain instances to connect to a virtual parallel port.
- In addition to supporting serial and parallel port emulation for virtual machines, some virtualization vendors support USB device pass-through from a host computer to a virtual machine.
  - USB pass-through allows a USB device plugged directly into a host computer to be passed through to a virtual machine.
  - USB pass-through allows for multiple USB devices (such as security dongles and storage devices) that are physically attached to a host computer to be added to a virtual machine.
- When a USB device is attached to a host computer, that device is available only to the virtual machines that are running on that host computer and only to one virtual machine at a time.

# Resource Pools

- A hierarchical abstraction of compute resources that can give relative importance, or weight, to a defined set of virtualized resources.
  - Pools at the higher level in the hierarchy are called parent pools;
  - Parent pools can contain either child pools or individual virtual machines.
- Each pool can have a defined weight assigned to it.
  - Allows administrators to define a flexible hierarchy that can be adapted at each pool level as required by the business.
- Hierarchical structure makes it possible to:
  - Maintain access control and delegation of the administration of each pool and its resources;
  - Ensure isolation between the pools, as well as sharing within the pools;
  - Separate the compute resources from discrete host hardware.

# Dynamic Resource Allocation

- Instead of relying on administrators to evaluate resource utilization and apply changes to the environment that result in the best performance, availability, and capacity arrangements, a computer can do it for them.
  - Based on business logic that has been predefined by either the management software's default values or the administrator's modification to those values.
- Management platforms have the ability to manage compute resources not only for performance, availability, and capacity reasons but also to realize more cost-effective implementation of those resources in a data center.
  - Employing only the hosts required at the given time and shutting down any resources that are not needed.
- Enables providers to both reduce power costs and go greener by shrinking their power footprint and waste.

# Configuration Best Practices

- To best understand their use cases and potential impact, we'll look at common configuration options for:
  - Memory
  - Processor
  - Disk.

# Memory

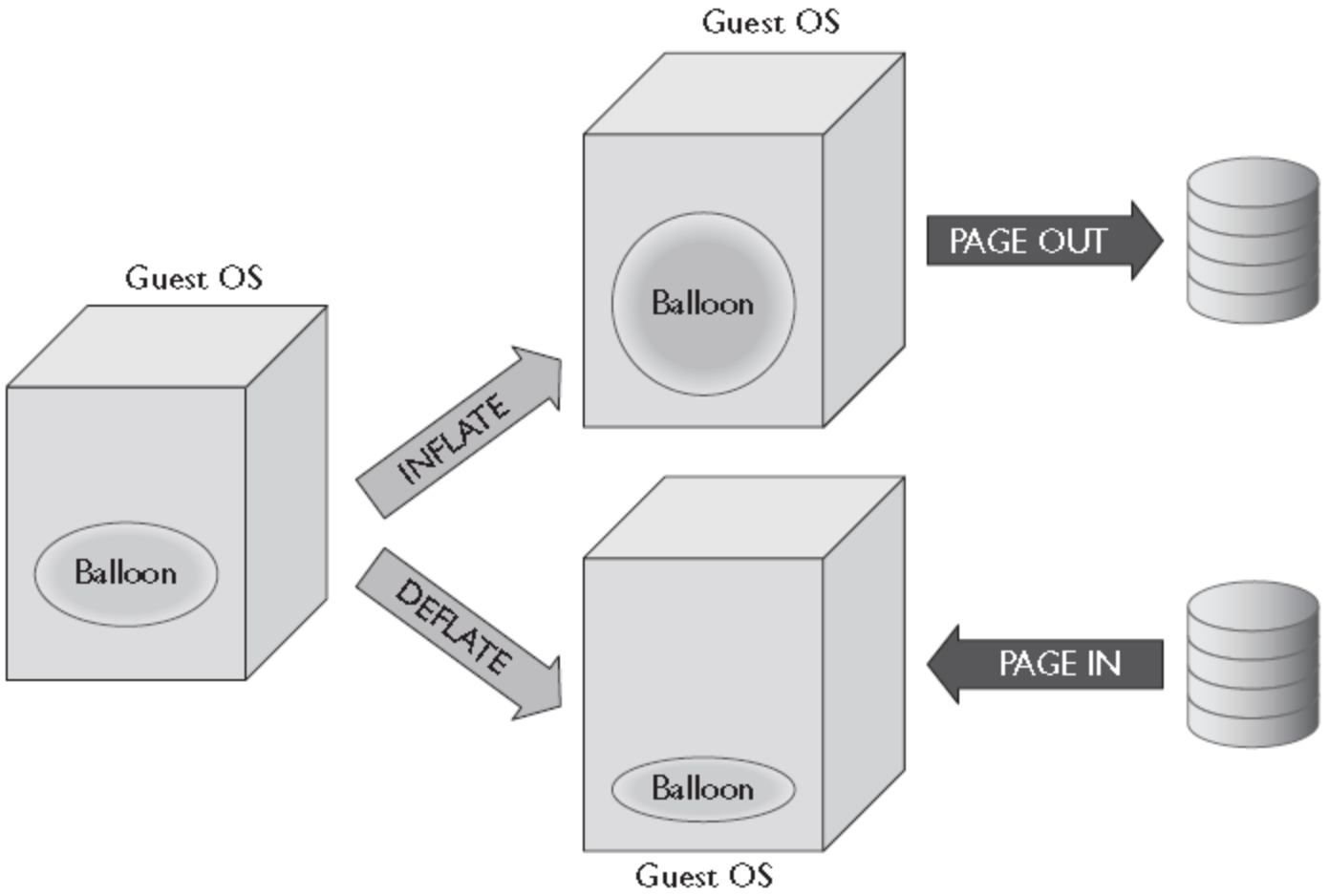
- Usually limiting factor on the number of guests that can be run on a given host.
  - Too many guests can cause performance issues.
- Two configuration options available for addressing shared memory concerns are

1. Memory ballooning
2. Swap disk space.

# Memory Ballooning

- Hypervisors have device drivers that they build into the host virtualization layer from within the guest operating system.
- Part of this installed tool set is a balloon driver.
  - Can be observed inside the guest.
  - The balloon driver communicates to the hypervisor to reclaim memory inside the guest when it is no longer valuable to the operating system.
  - If the host begins to run low on memory, it will grow the balloon driver to reclaim memory from the guest.
- Reduces the chance that the physical host will begin to utilize virtualized memory from a defined paging file on its available disk resource, which causes performance degradation.

# Memory Balooning



# Swap Disk Space

- Swap space is disk space that is allocated to service memory requests when physical memory capacity limit is reached.
- When virtualizing and overcommitting memory resources to virtual machines, administrators must make certain to reserve enough swap space for the host to balloon memory in addition to reserving disk space within the guest operating system for it to perform its own swap operations.

# Processor

- PU time of the threads is additive.
  - The application CPU time is the sum of the CPU time of all the threads that run the application.
- Wait time is the amount of time that a given thread waits to be processed
  - It could be processed but must wait on other factors such as synchronization waits and I/O waits.
  - High CPU wait times signal too many requests for a given queue on a core to handle
    - Performance degradation will occur.
  - While high CPU wait time can be alleviated in some situations by adding processors, these additions sometimes hurt performance as well.
- Caution must be exercised when adding processors as there is a potential for causing even further performance degradation if the applications using them are not designed to be run on multiple CPUs.
- Another solution for alleviating CPU wait times is to scale out instead of scaling up.

# Disk

- In a cloud model, disk performance issues can limit access to all organization resources because multiple virtualized servers in a networked storage environment might be competing for the same storage resources.

# Disk Performance

- Several different configuration options.
- Media type can affect performance
- Administrators can choose between the most standard types of traditional rotational media or chip-based solid state drives.
- Solid state drives much faster.
- Solid state drives still much more expensive.

# Rotational Media

- The next consideration for disk performance is the speed of the rotational media.
- Server-class disks start at 7,200 rpm and go up to 15,000 rpm.
- Seek times for the physical arm reading the platters being considerably lower on the high-end drives.
- In enterprise configurations, price point per gigabyte is largely driven on the rotation speed and only marginally by storage space per gigabyte.
- With enterprise storage, you pay for performance, not space.

# RAID

- Once the media type and speed have been determined, the next consideration is the RAID type.
- Different levels of RAID can be employed based on the deployment purpose.
- These RAID levels should be evaluated and configured based on the type of I/O and on the need to read, write, or a combination of both.

# Disk Tuning

- Activity of analyzing type of I/O traffic taking place across the defined disk resources and moving it to the most appropriate set of resources.
- Virtualization management platforms enable the movement of storage, without interrupting current operations, to other disk resources within their control.
- Allows either administrators or dynamic resource allocation programs to move applications, storage, databases, and even entire virtual machines among disk arrays with no downtime to make sure that those virtualized entities get the performance they require based on either business rules or SLAs.

# Disk Latency

- Disk latency is a counter that provides administrators with the best indicator of when a resource is experiencing degradation due to a disk bottleneck and needs to have action taken.
- If high latency counters are experienced, a move to either another disk array with quicker response times or a different configuration, such as higher rotational speeds or a different array configuration, is warranted.
- Another option is to configure I/O throttling.

# I/O Throttling

- I/O throttling does not eliminate disk I/O as a bottleneck, but it can alleviate performance problems for specific virtual machines.
- I/O throttling defines limits that can be utilized specifically for disk resources assigned to virtual machines to ensure that they are not performance or availability constrained when working in an environment that has more demand than availability of disk resources.
  - May be a valuable option when an environment contains both development and production resources.
  - The production I/O can be given a higher priority than the development resources, allowing the production environment to perform better for its users.
- Does not eliminate the bottleneck; it just passes it on to the development environment.
  - Which becomes even further degraded in performance as it waits for all production I/O requests when the disk is over allocated.

# I/O Tuning

- When designing systems, administrators need to analyze input and output (I/O) needs from the top down.
- In order to perform this top-down evaluation, first the application I/O requirements need to be evaluated to understand how many reads and writes are required by each transaction and how many transactions take place each second.
- Once those application requirements understood, the disk configuration (specifically, which types of media, what array configuration, the number of disks, and the access methods) can be built.

# Common Issues

- A number of failures that can occur within a cloud environment, and the system must be configured to be tolerant of those failures and provide availability in line with the organization's SLA.
- Any mechanical environment will experience failures; it is just a matter of when and the quality of the equipment the company has purchased.
- Failures occur mainly on each of the four primary compute resources:

1. Disk
2. Memory
3. Network
4. Processor

# Common Disk Failures

- Disk are only compute resource with mechanical components.
  - Due to the moving parts, failure rates are relatively high.

Physical Hard Disk Failures

- Physical hard disks fail frequently because they are mechanical devices.
- In enterprise configurations they are deployed as components of drive arrays, and single failures do not affect array availability.

# Controller Card Failures

- Redundant controllers are very expensive to run in parallel as they require double the amount of drives to become operational.
- Therefore, an organization should do a return-on-investment analysis to determine the feasibility of making such devices redundant.

# Disk Corruption

- Occurs when the structured data on disk is no longer accessible.
- Can happen as a result of malicious acts or programs, skewing of the mechanics of the drive, or even a lack of proper maintenance.
- Difficult to repair.
- Backups can also be unreliable for these failures if the corruption began prior to its identification, as the available backup sets may also be corrupted.

# Host Bus Adapter

- (HBA) failures, while not as common as physical disk failures, need to be expected and storage solutions need to be designed with them in mind.
- HBAs have the option of being multipathed, which prevents a loss of availability in the event of a failure.

# Fabric/Network Failures

- Similar to arrays, fabric or network failures can be fairly expensive to design around, as they happen when a storage networking switch or switch port fails.
- Design principles to protect against such a failure are similar to those for HBAs, as multipathing needs to be in place to make certain all hosts that depend on the fabric or network have access to their disk resources through another channel.

# Common Memory Failures

- Good system design in cloud environments will take RAM failure into account as a risk and ensure that there is always RAM available to run mission-critical systems. Some types of memory failurefollow.
- RAM Failures
  - Memory chip failures happen less frequently than physical device failures.
  - They will, however, break from time to time and need to be replaced.

# Motherboard Failures

- Motherboards have no moving parts and fail less frequently than mechanical devices.
- When they do fail, however, virtual machines are unable to operate as they have no processor, memory, or networking resources that they can access.
- In this situation, they must be moved immediately to another host or go offline.

# Swap Files Out of Space

- Swap space failures often occur in conjunction with a disk failure, when disks run out of available space to allocate to swap files for memory overallocation.
- They do, however, result in out-of-memory errors for virtual machines and hosts alike.

# Network Failures

- Unlike memory, network resources are highly configurable and prone to errors based on human mistakes during implementation.
- Some common types of network failures follow.

Physical NIC Failures

- Because they fail from time to time, redundancy needs to be built into the host through multiple physical NICs and into the virtualization through designing multiple network paths using virtual NICs for the virtual machines.

# Speed/Duplex Mismatches

- Mismatch failures happen only on physical NICs and switches, as virtual networks negotiate these automatically.
- Speed and duplex mismatches result in dropped packets between the two connected devices, and can be identified through getting many cyclical redundancy check (CRC) errors on the devices.

# Switch Failures

- Similar to fabric and network failures, network switch failures are expensive to plan for as they require duplicate hardware and cabling.
- Switches fail wholesale only a small percentage of the time.
  - More frequently have individual ports fail.
- When these individual ports fail, the resources that are connected to them need to have another path available or their service will be interrupted.

# Physical Transmission Media Failures

- Cables break from time to time when their wires inside are crimped or cut.
  - This can happen either when they are moved, when they are stretched too far, or when they become old and the connector breaks loose from its associated wires.
- As with other types of network failures, multiple paths to the resource using that cable is the way to prevent a failure from interrupting operations.

# Physical Processor Failures

Processors fail for one of three main reasons, they :

1. Get broken while getting installed
2. Are damaged by voltage spike
3. Are damaged due to overheating from failed or ineffective fans.

- Damaged processors either take hosts completely off-line or degrade performance.

# Performance Concepts

- A number of performance concepts underlie each failure type.
- Let's look at each according to their associated compute resources.

Disk

- Configuration of disk resources important.
- Based on the user and application requirements and usage patterns, there are numerous design choices that need to be made to implement a storage system that meets an organization's needs in a cost-effective fashion.

Considerations for disk performance follow.

IOPS

- IOPS, or input/output operations per second, are the standard measurement for disk performance.
- Usually gathered as read IOPS, write IOPS, and total IOPS.

# Read Versus Write

- Reads take place when a resource requests data from a disk resource.
- Writes take place when a resource requests new data be recorded on a disk resource.
- Based on type of operation, different configuration options exist both for troubleshooting and performance tuning.

# File System Performance

- File systems can be formatted and cataloged differently based on the proprietary technologies of their associated vendors.
- Little to do in configuration of file systems performance outside of evaluating the properties of each that is planned for operation in the environment.

# Metadata Performance

- Metadata performance refers to how quickly files and directories can be created, removed, or checked.
- Applications exist now that produce millions of files in a single directory and create very deep and wide directory structures, and this rapid growth of items within a file system can have a huge impact on performance.
- Ability to create, remove, and check their status efficiently grows in direct proportion to the number of items in use on any file system.

# Caching

- In order to improve performance, hard drives are architected with a disk cache that reduces both read and write times.
- On a physical hard disk, the disk cache is usually a RAM chip that is built in and holds data that is likely to be accessed again soon.
- On virtual hard disks, same caching mechanism can be employed by using a specified portion of a memory resource.

# Network

- Configuration of network resources is critical.

Bandwidth

- Bandwidth is the measurement of available or consumed data communication resources on a network.
- Performance of all networks is dependent on having available bandwidth.

Throughput

- Throughput is the amount of data that can be realized between two network resources.
  - Throughput can be greatly increased through the use of bonding or teaming of network adapters, which allows resources to see multiple interfaces as one single interface with aggregated resources.

# Jumbo Frames

- Jumbo frames are Ethernet frames with more than 1500 bytes of payload.
  - Can carry up to 9000 bytes of payload.
  - Utilized because they are much less processor intensive than a large number of smaller frames.

Network Latency

- Network latency refers to any performance delays experienced during the processing of any network data.
- A low-latency network connection is one that generally experiences small delay times, such as a dedicated T-1, while a high-latency connection generally suffers from long delays, like DSL or a cable modem.

Hop Counts

- A hop count represents the total number of devices a packet passes through in order to reach its intended network target.
- The more hops data must pass through to reach their destination, the greater the delay will be for the transmission.

# Network Utilities

- Ping can be used to determine the hop count.
  - Ping generates packets that include a field reserved for the hop count (typically referred to as a TTL, or time-to-live), and each time a capable device (typically a router) along the path to the target receives one of these packets, that device modifies the packet, decrementing the TTL by one.
  - Each packet is sent out with a particular time-to-live value, ranging from 1 to 254; for every router (hop) that it traverses, that TTL count is decremented.
- In addition, for every one second that the packet resides in the memory of the router, it is also decremented by one.
- The device then compares the hop count against a predetermined limit and discards the packet if its hop count is too high.
- If the TTL is decremented to zero at any point during its transmission, an ICMP port unreachable message is generated, with the IP of the source router or device included, and sent back to the originator.
- The finite TTL is used as it counts down to zero in order to prevent packets from endlessly bouncing around the network due to routing errors.

# Quality of Service (QoS)

- QoS is a set of technologies that can identify the type of data in data packets and divide those packets into specific traffic classes that can be prioritized according to defined service levels.
- Enable administrators to meet service requirements for a workload or an application by measuring network bandwidth, detecting changing network conditions, and prioritizing the network traffic accordingly.
- QoS can be targeted at a network interface, toward a given server or router's performance, or in terms of specific applications.
- A network monitoring system is typically deployed as part of a QoS solution to ensure that networks are performing at the desired level.

# Multipathing

- Multipathing is the practice of defining and controlling redundant physical paths to I/O devices, so that when an active path to a device becomes unavailable, the multipathing configuration can automatically switch to an alternate path in order to maintain service availability.
- Capability of performing this operation without intervention from an administrator is known as automatic failover.
- A prerequisite for taking advantage of multipathing capabilities is to design and configure the multipathed resource with redundant hardware, such as redundant network interfaces or host bus adapters.

# Load Balancing

- A load balancer is a networking solution that distributes incoming traffic among multiple servers hosting the same application content.
- Load balancers improve overall application availability and performance by preventing any application server from becoming a single point of failure.
- If deployed alone, however, the load balancer becomes a single point of failure by itself.
- Therefore, it is always recommended to deploy multiple load balancers in parallel.
- In addition to improving availability and performance, load balancers add to the security profile of a configuration by the typical usage of network address translation, which obfuscates the IP address of the back-end application servers.

# Scalability

- Scalability can be vertical or horizontal.
  - Commonly referred to as “scaling up” or “scaling out”.
- To scale vertically means to add resources to a single node, thereby making that node capable of handling more of a load within itself.
  - Most often seen in virtualization environments where individual hosts add more processors or more memory with the objective of adding more virtual machines to each host.
- To scale horizontally, more nodes are added to a configuration instead of increasing the resources for any one node.
  - Often used in application farms, where more web servers are added to a farm to better handle distributed application delivery.
- A third type of scaling, diagonal scaling, is a combination of both, increasing resources for individual nodes and adding more of those nodes to the system.
  - Diagonal scaling allows for the best configuration to be achieved for a quickly growing, elastic solution.

# Questions???