9.12.9 Clustering Facts

Clustering is connecting a group of independent computers to increase the availability to applications and services. Each clustered server is called a *node*. The nodes are connected physically by cables and use software to monitor and maintain the connections.

- Clusters typically use a storage area network (SAN) to provide access to the shared storage. Cluster members have a network connection to the regular network to respond to client requests and a network connection to the SAN to access the shared storage.
- The cluster is identified by a shared IP address. Client requests are directed to the shared IP address, not the IP address of an individual cluster member.
- Cluster members send out periodic *heartbeat* signals to communicate with each other to maintain consistent information about cluster membership.
- Failover clustering provides redundancy for services or applications. If a service, application, or an entire server in the cluster fails, failover redirects client requests to other servers in the cluster. *Failback* (also called fallback) is the process of returning client requests to the failed service or server when it comes back online.
- *Convergence* is the process that cluster members use to reach a consistent state, meaning that all cluster members are aware of all other members and the client load has been distributed between cluster members according to the load balancing rules.
- Using clustering to ensure a service is accessible most of the time is a form of *high availability*.
- *Elasticity* is the level of difficulty involved when removing nodes from the data store.
- *Scalability* is a system's ability to handle a growing level of work.

A *high availability cluster* (HA) is a group of computers that are configured with the same service. In HA clusters:

- One node is configured as the master node, and other nodes are configured as slave nodes.
- The master node provides requested information to network users; the slave nodes are inactive.
- Master and slave nodes continually communicate via heartbeats and are connected to the same shared storage.
- When the master node fails, a slave node takes over.
- A single point of failure is eliminated through the use of redundant nodes.

A load balancing cluster disperses a workload between two or more computers or resources to achieve optimal resource utilization, throughput, or response time. Load balancing improves performance by distributing the workload between multiple servers. Load balancing also provides fault tolerance; if one server is unavailable, additional servers are available to fulfill the request.

- All of the nodes in a load balancing cluster are active participants at all times.
- All of the processing tasks to be completed are distributed between all of the nodes in the cluster.
- Depending on the implementation, nodes can share processing capabilities, storage, and system RAM.
- Nodes in a load balancing cluster can be tightly or loosely linked. The tighter the link, the more the nodes function as one system.
- A tightly linked load balancing cluster is known as a supercomputing cluster.

The more tightly linked the nodes in the cluster are, the more identical the nodes need to be.

The intent of load balancing is to virtualize a service, such as a web or a database service, offered by multiple servers. If the servers are not clustered with load balancing capabilities, a separate load balancer can be used. The load balancer forwards the service request from a client to a single member of the cluster. It chooses or schedules the member based on an algorithm.

- **Round robin**: There is no priority for selecting a member. Each member receives an equal share of requests portioned out in a circular order.
- **Affinity**: A member is selected based on an affinity. When it is desirable to send all service requests from a user to the same cluster member, an affinity can be established based on the IP address of the client or the class C address space of the client IP address.
- **Least connections**: The member with the least number of connections is chosen.
- **Least response time**: The member who responds most quickly to a request is chosen.

For higher availability, two load balancers can be used in either an active-passive mode, or active-active mode.

- **Active-passive**: One load balancer is active and handles all the service request. The passive load balancer is in listening mode and monitors the performance of the active load balancer. If the active load balancer fails, the passive load balancer become active and takes over the load balancing duties.
- **Active-active**: Both load balancers work as a team to distribute the service requests.

For high availability, multiple load balancers can be clustered in the same way as other server clusters.

In regards to load balancing, a *Virtual IP* (VIP) is an address presented to the outside world, but doesn't correspond to an actual physical network interface. To the client, the VIP responds like any normal IP address. The load balancing environment is responsible for forwarding service request from the client to a physical server who responds to the request.