System Administration

Clustering and failover strategies

http://courses.cs.ut.ee/2012/syshald/
Outline

- Test!

- Clustering and failover strategies
Clusters

- High-availability (or failover) clusters are groups of computers that support server applications that can be reliably utilized with a minimum of down-time

- HA cluster implementations attempt to build redundancy into a cluster to eliminate single points of failure, including multiple network connections and data storage which is redundantly connected via storage area networks (wikipedia)
Cluster components

- Specially designed application software
  - Client side software (in clients/users computer):
    - Must verify every data modification with server
    - Must be able to redo every transaction when server does respond with confirmation
Cluster components (2)

- Specially designed application software
  - Server side software:
    - Must write as much of the state to hard disk as possible
    - Must have mechanisms to avoid data corruption in case of crash (some kind of versioning with data integrity control)
Cluster components (3)

- Clustering middleware
  - Software for node membership management (Heartbeat)
    - Provides messaging layer between nodes
    - Checks health of the nodes
    - Assigns node roles
Cluster components (4)

- Clustering middleware
  - Software for cluster resource management (Pacemaker)
    - Starts and stops services (with help of local resource manager), assigns ip-addresses to nodes, mounts file systems, etc.
  - Software for emergency actions with failing nodes, STONITH (Cluster Glue)
    - Shuts down or disables failed nodes
Cluster components (5)

- Cluster aware hard disk storage solution
  - Internally redundant architecture
  - Multiple paths to disk array
  - Internal disk redundancy (raid)
- Optionally: cluster aware file system
  - (gfs2, gpfs)
Failover cluster examples

- Application server cluster scenario
  - Two node servers, each connected to SAN with two fiberchannel interfaces
  - Cluster middleware negotiates and assigns one node as active
  - Resource management mounts shared file system, assigns ip-address, starts application
Failover cluster examples (2)

- Application server cluster scenario (2)
  - Node membership manager checks periodically health of the nodes
  - When active node fails
    - STONITH shuts it completely down or disables
    - Promotes failover node to active
Failover cluster examples (3)

...  

- Active node mounts shared file system and starts application  
- When failure is corrected by human, then node is re-added to cluster as passive node
Failover cluster examples (4)

• Application server cluster scenario (2)
  – When should active node be declared as “dead”?
    • When it does not respond to cluster “heartbeat” (similar to ping)?
      – Maybe heartbeat works but application is crashed?
    • When monitoring system receives incorrect answer from application?
      – What happens when user triggers a bug in application and it crashes?
Failover cluster examples (5)

• Application server cluster scenario (3)
  - When active node fails due hardware error, shared filesystem may left to inconsistent state. New active node usually succeeds on recovery but that is not guaranteed.
Active-Passive Clusters

- Described solution is called “active-passive cluster”
- It uses one currently active node and 1..n passive, standby nodes
- Implies usage of multipath-capable storage solution (which costs ~5000€ at entry level)
- Uses at least two servers to provide one-server-worth service
Active-Passive Clusters (2)

- Protects from number of possible errors but introduces whole family of new risks, which are very hard to assess
Active-Active Clusters

- Can all the nodes used simultaneously for load balancing?
  - Yes, Active-Active clustering
- Implies usage of cluster-aware file system
- Requires additional functionality from application
  - Fine-grained locking, special mechanisms to avoid dead-locks and race-conditions
Active-Active Clusters (2)

- Due complex requirements to application software, active-active clustering is rarely used
- Several commercial database providers offer active-active clustering solutions to their products
  - They use cluster-aware-filesystem-like solutions inside their data-files
Failover

- Shared storage is expensive. Can it be avoided in redundant system?
  - No, in general
  - Yes, in certain (yet common) scenarios
Failover (2)

- Redundant Web Server scenario
  - Two or more completely independent web servers
  - Configured to serve virtualhost with same name (www.google.com for example)
  - Web contents are replicated to both (all) servers
  - DNS server has two or more A-type records for virtualhost name
Failover (3)

• Redundant Web Server scenario (2)
  – When request for ip-address is made to DNS server and multiple records exists, DNS responds them in random order
  – When browser can not connect to server with first ip-address, it tries second and so on
  • This functionality is actually provided by operating system core libraries (libc), not by browser.
Failover (4)

- Redundant Web Server scenario (3)
  - Changes to content must be synchronized by some external means
  - This scenario also gives “free” load balancing and can easily scaled-up by adding additional servers
Failover (5)

- Redundant mail gateway (antispam) scenario
  - 2..n type-A records in DNS
  - 2..n independent servers
  - Transaction and verification support is all-ready natively built in to mail clients
Failover (6)

- Real world scenario
  - 2..n independent front-end servers
    - Web server with php
    - Java application server
      ... which connect to ...
  - Clustered database server
Questions?