NAS and Clustering

Ashutosh Tripathi

Sun Microsystems

ashutosh.tripathi@sun.com

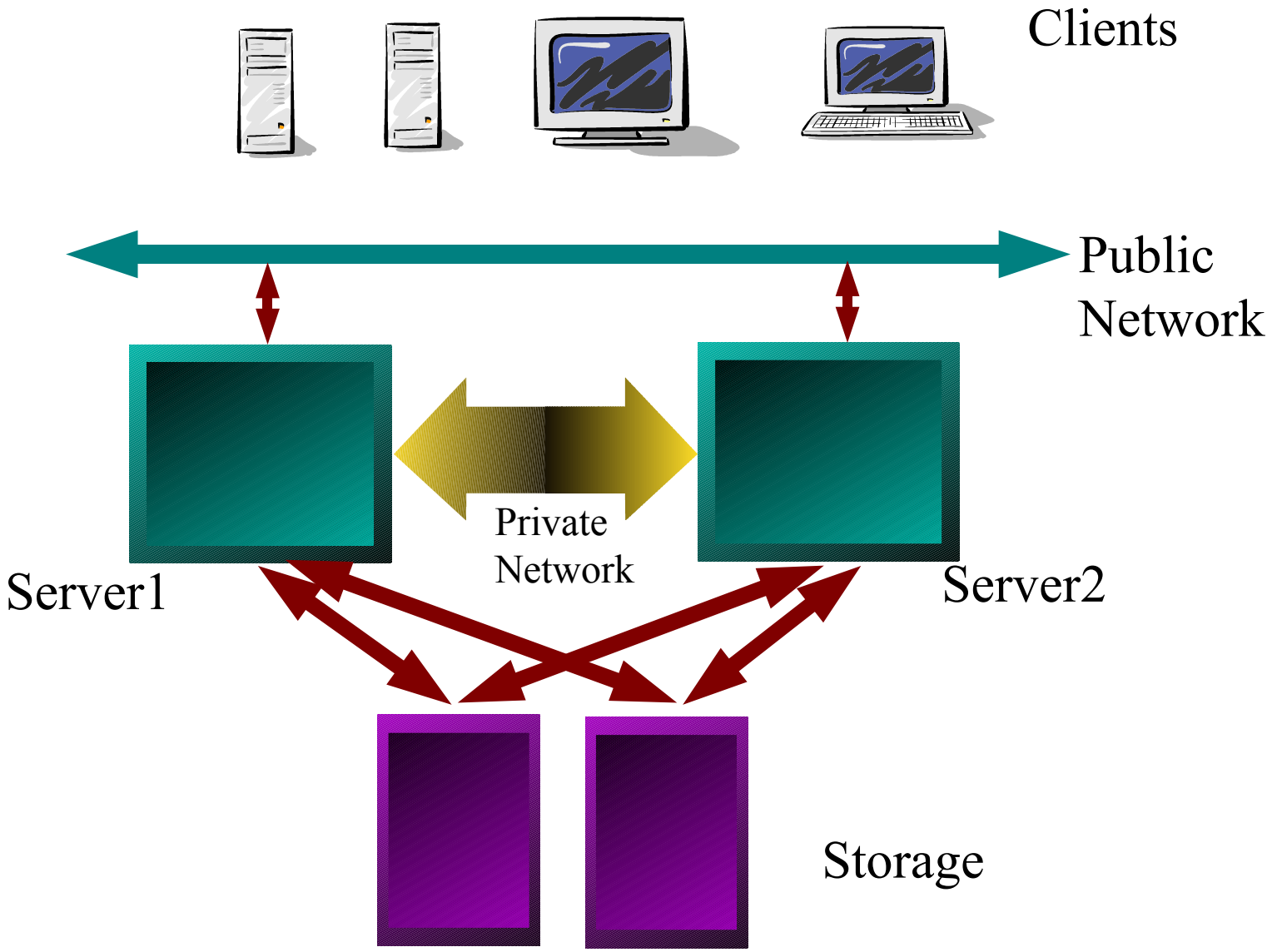
HA NAS Server

- NAS Server is a key infrastructure in the data center
- Typical deployments with Clustering
  - Home directory, Mail server
  - Software repository for key apps
    - SAP
    - Middleware

Why Clustering?

- Reduced downtime
  - Protect from HW failures
    - Server crash
    - Network (NIC, cable, switch/port)
    - Storage (HBA, cable, controller port)
  - SW failures
    - OS crash/hangs
- Manageability
  - Reduce planned downtime
    - Patching, upgrading
    - Load balancing

SunCluster Configuration



How fast is the recovery?

- Recovery from server crash
  - Heartbeat loss detection ~ 10 sec
  - Quorum ~ 9 sec (SCSI-II)
    - SCSI-III and Fiber Channel are much faster
  - Storage Fencing ~ 10 seconds
  - Volume manager failover ~ 2 seconds
    - Mirror recovery, if needed, happens in background
  - Filesystem recovery (fsck), highly variable
    - Use of a logging filesystem is necessary

How fast....

- NFS (v3) recovery
  - Sharing the filesystem and restarting the daemons ~ 1 sec
- Total ~ 32 seconds + FS Recovery
  - NFS client side retries can delay client side recovery further
  - Clients doing locking would experience an additional grace period delay (45 seconds default with NFSv3 on Solaris)
- Other recovery scenarios (e.g. Smooth switchover) can be faster as no heartbeat timeout or Quorum

NAS & Clustering

Integration notes

- Multi-host issues
  - **Clustering environments are multi-hosted**
    - NAS server identity (IPaddress etc.) needs to failover
  - **Firewalls**
    - Clients connect to the failover hostnames, they expect all communication to originate from that hostname
    - Tricky to do with UDP
    - With NFSv3 multiple TCP ports doesn't help much
    - Server calling back to clients is another scenario
    - Will be better in NFSv4, single TCP port

NAS & Clustering

Kerberos Integration

- Kerberos has separate notions of **host** and **service**
  - Host principals vs service principals
- The **host** part is an issue in Clustered environments
  - Service principal *nfs/hostname.domain*
  - NAS server implementations (as well as GSS implementations) need to manage the *hostname* part carefully
  - Use the TCP peername instead of simply using the hostname of the server machine

Using NAS storage in a Cluster

- Cluster is a NAS client
- Applications on the Cluster failover from node to node
- At the time of a node crash, application may have pending I/O, and/or locks

Using NAS...

- SunCluster needs to provide fencing guarantees
  - No well defined way to do this (yet) in NFS protocol
  - One-of solutions are being worked on
  - Clearing locks held by dead nodes is another related issue
  - Need to define this within the NFS protocol itself
    - Security is an issue
    - Leave the client health detection to outside the protocol (Clustering or manual)

Future investigation

- Scalable NFS
  - SunCluster has built-in load balancing
  - SunCluster has distributed filesystem support
    - Proxy Filesystem (a.k.a. Global Filesystem)
    - SAM-QFS
  - SunCluster has built-in heartbeat based membership detection
  - Sharing the same filesystem from all cluster nodes is an easy way to achieve scalability
  - How does this compare with pNFS?
  - NFSv4 is very stateful, how to manage this state on different Cluster nodes?

Summary

- NAS server is a key infrastructure in the data center
- Clustering is a viable approach for protecting this infrastructure
- Scope for better integration between NAS and Clustering