



# **CASE STUDY SUN NFS - VFS, ACCESS CONTROL, NFS SERVER INTERFACE, PATH NAME TRANSLATION, MOUNT SERVICE**

Shashwat Singh

IT-B

Roll no :- 21

# Case Study SUN NFS - VFS



- ❖ The Sun Network Filesystem (NFS) provides transparent, remote access to filesystems.
- ❖ Unlike many other remote filesystem implementations under UNIX, NFS is designed to be easily portable to other operating systems and machine architectures.
- ❖ The NFS client and server modules communicate using remote procedure calling.
- ❖ The overall design goals of NFS were :-
  - ✓ Machine and Operating System Independence
  - ✓ Crash Recovery
  - ✓ Transparent Access



# Case Study SUN NFS - VFS

- ❖ Virtual File System (VFS) interface defines the operations that can be done on a filesystem
- ❖ The virtual file system layer has one VFS structure for each mounted file system and one v-node per open file.
- ❖ A VFS structure relates a remote file system to the local directory on which it is mounted.
- ❖ The v-node contains an indicator to show whether a file is local or remote.



# Access Control

- ❖ NFS server is stateless and does not keep files open on behalf of clients .
- ❖ Therefore, server must check the user's identity against the file's access permission attributes on each request , to see whether the user is permitted to access the file in the manner requested.
- ❖ The Sun RPC protocol requires clients to send user authentication information with each request and this is checked against the access permission in the file attributes.





# NFS server Interface

- ❖ The file and directory operations are integrated in a single service; the creation and insertion of file names in directories is performed by a single create operations.
- ❖ The other NFS operation on directories are *create*, *remove*, *link*, *symlink*, *readlink*, *mkdir*, *rmdir*, *readdir* and *statfs*.
- ❖ They resemble their UNIX counterparts with the exception of *readdir* , which provides a representation-independent method for reading the contents of directories, and *statfs*, which gives the status information on remote file systems.

# Mount Service

A decorative blue wavy line with a gradient, starting from the top right and curving downwards and to the left, ending near the center of the slide.

- ❖ The mounting of sub-trees of remote filesystems by clients is supported by a separate mount service process that runs at user level on each NFS server computer .
- ❖ On each server , there is a file with a well-known name (/etc/exports) containing the names of local filesystems that are available for remote mounting .
- ❖ An access list is associated with each filesystems name indicating which hosts are permitted to mount the filesystem .



# Path name translation

- ❖ In NFS, pathnames cannot be translated at a server, because the name may cross a ‘mount point’ at the client.
- ❖ So pathnames are parsed, and their translation is performed in an iterative manner by the clients.
- ❖ The *lookup* operation looks for a single part of a pathname in a given directory and returns the corresponding file handle and file attributes. The file handle returned in the previous step is used as a parameter in the next *lookup* step.



THANK YOU