

CSE 265:

System and Network Administration

- The Network File System
 - NFS Introduction
 - Server-side NFS
 - Client-side NFS
 - NFS Statistics with `nfsstat`
 - Dedicated NFS File Servers
 - Automatic Mounting

NFS

- Network File System
- Allows systems to share filesystems among computers
- Designed to be transparent and stateless
- Consists of
 - A mounting protocol
 - Mount server
 - File service daemons
 - Diagnostic utilities

Magic cookies

- The server doesn't track which clients have mounted filesystems (stateless)
- Instead, the server discloses a secret/magic cookie that identifies the directory to the server on future access
 - Often the cookie is just the filesystem major and minor device IDs, plus directory inode
- Unmounting and remounting the actual filesystem on the server normally changes the cookie

NFS

- Network File System
 - Version 2: slow
 - Originally released by Sun in 1985
 - Version 3: faster (common)
 - Version 4: security, locking (new)
- Uses Sun's RPC (Remote Procedure Call) protocol (documented in RFC 1050, 1988)
 - Supports UDP or TCP for transport
- File locking is worse under NFS v3 since servers are stateless

Security and NFS

- Not originally designed for security!
- Access to NFS volumes is determined via `/etc/exports`
 - lists hostnames or IP addresses that have access
 - assumes clients will identify themselves correctly
- TCP wrappers can help protect service
- File-level access is managed according to UID, GID, and file permissions
 - Just as in local file systems

NFS Security Problems

- Users with given UID can access any file with that UID (even if different user)
 - Good reason for globally unique UID space!
- Root access on a client can access any file
- NFS typically uses option called “squashing root”
 - Makes incoming requests for UID 0 look like they came from some other user
 - Account named nobody is utilized
- Option `all_squash` does the same for all users

Server-side NFS

- Servers “export” a directory to make it available to others
- Servers run two daemons
 - rpc.mountd to handle mount requests
 - rpc.nfsd for actual file service
- Filesystems to be exported are in /etc/exports

```
# sample /etc/exports file
/                master(rw) trusty(rw,no_root_squash)
/projects        proj*.local.domain(rw)
/usr             *.local.domain(ro) @trusted(rw)
/home/joe        pc001(rw,all_squash,anonuid=150,anongid=100)
/pub            (ro,insecure,all_squash)
```

- Can modify and view exports using **exportfs**

Client-side NFS

- NFS filesystems are mounted much like local filesystems using **mount** hostname:directory
- Before mounting, filesystem must be exported
 - Check with **showmount**

```
#showmount -e wume2
Export list for wume2:
/projects2 *.local.cse.lehigh.edu,davison
/projects1 *.local.cse.lehigh.edu,davison
```

- Use **umount** to unmount an NFS filesystem
 - Can't be unmounted while in use
 - Use **lsof** to find processes with open files

Mounting NFS filesystems

- Use **mount** for temporary mounts

```
# mount -o rw,hard,intr,bg server:/home /home
```

- `/etc/fstab` contains mounts for boot time

```
wume1:/home          /home          nfs          \  
    intr,bg,rw 1 1  
wume1:/var/spool/mail /var/spool/mail nfs          \  
    intr,bg,rw 1 1
```

- Common options:
 - rw, ro, bg, hard, soft, intr, tcp, udp

NFS Statistics and Utilities

– nfsstat

```
Server rpc stats:
calls      badcalls  badauth   badclnt   xdrcall
40996991   0         0         0         0
Server nfs v3:
null      getattr   setattr   lookup    access    readlink
2         0% 428484   1% 25913    0% 444794   1% 398283   0% 3174     0%
read      write     create    mkdir     symlink   mknod
10193400  24% 29048042 70% 69068   0% 695      0% 3110     0% 0         0%
remove    rmdir     rename    link      readdir   readdirplus
5014     0% 81        0% 103716  0% 0         0% 38649   0% 1625     0%
fsstat    fsinfo    pathconf  commit
853      0% 356      0% 0         0% 231730  0%
```

– netstat

- General network statistics, may help debugging

– showmount -a

- Shows all systems believed to have mounted filesystems

NFS Tuning

- Default read and write client buffer sizes are just 1KB
 - Recommended is 32KB for TCP
 - Change `/etc/fstab`
- Default input queue size for server is 64KB
 - Recommended is ~256KB
 - Only want to affect `nfsd`, not all network services
 - Change `/proc/sys/net/core/rmem_*`
 - Restart `nfsd`; put original value back

Dedicated NFS File Servers

- Dedicated NFS appliances are available
 - Network Appliance, EMC, HP/Dell (Microsoft), etc.
- Features
 - Provide Network Attached Storage (NAS)
 - Optimized for file service
 - Can scale to lots of storage and users
 - Often provide service to both *nix and Windows clients
 - More reliable
 - simpler software, redundant hardware, RAID
 - Easy to administer
 - Often provide backup and checkpoint facilities

Automatic Mounting

- Separate lines in `/etc/fstab` can be difficult in large networks
 - Maintaining `/etc/fstab` on more than a few dozen machines is tedious
 - Worse is when those machines mount from many hosts
- When an important host crashes, clients are crippled
 - Having a copy of the partition mountable elsewhere would be ideal
- An automounter mounts filesystems only when needed, and can work with replicated systems for redundancy

automount

- A background process that watches for requests for files within a specified directory
 - Uses autofs kernel-resident filesystem driver
 - Then mounts the requested filesystem
- /etc/init.d/autofs script is configured via /etc/auto.master

```
/misc /etc/auto.misc --timeout=60
```

- Each mount point has separate map file (or script), listing all valid subdirectories and how to get them

automount example

```
# This is an automounter map and it has the following format
# key [ -mount-options-separated-by-comma ] location
# Details may be found in the autofs(5) manpage
cd                -fstype=iso9660,ro,nosuid,nodev :/dev/cdrom
brian-sun-windows -fstype=smbfs,rw,noexec,username=brian,password=XYZ,uid=501,gid=501 ://guttenberg/brian
# the following entries are samples to pique your imagination
#linux           -ro,soft,intr          ftp.example.org:/pub/linux
#boot            -fstype=ext2                :/dev/hda1
#floppy          -fstype=auto                 :/dev/fd0
```

```
% mount
/dev/hda2 on / type ext3 (rw)
automount(pid7909) on /misc type autofs (rw,fd=5,pgrp=7909,minproto=2,maxproto=3)
```

```
% ps aux | grep automount
root      7909  0.0  0.1  2644  620 pts/1    S   19:43   0:00 /usr/\
sbin/automount --timeout 60 /misc file /etc/auto.misc
```