NFS Client Authentication

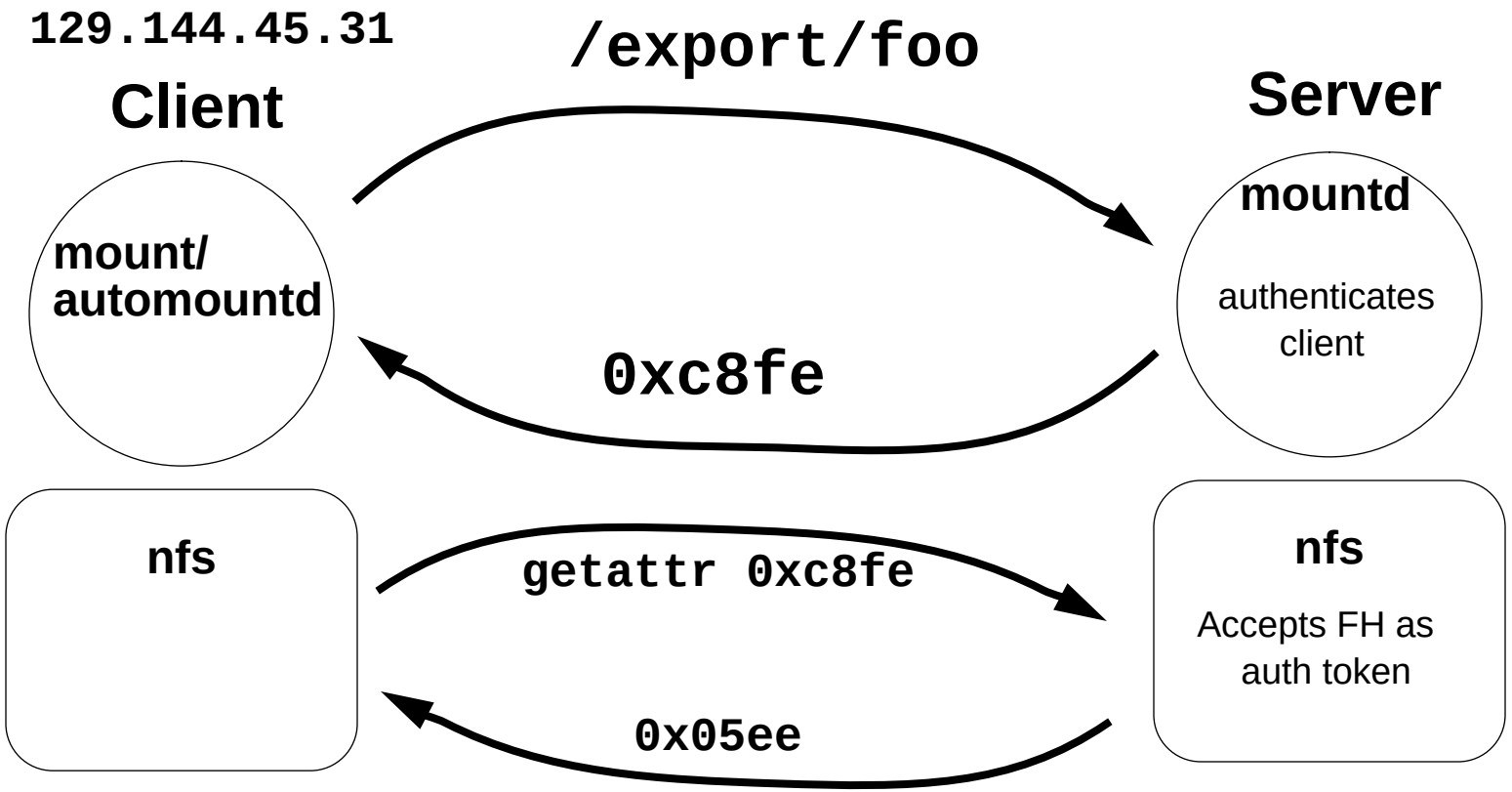
Brent Callaghan

Motivation

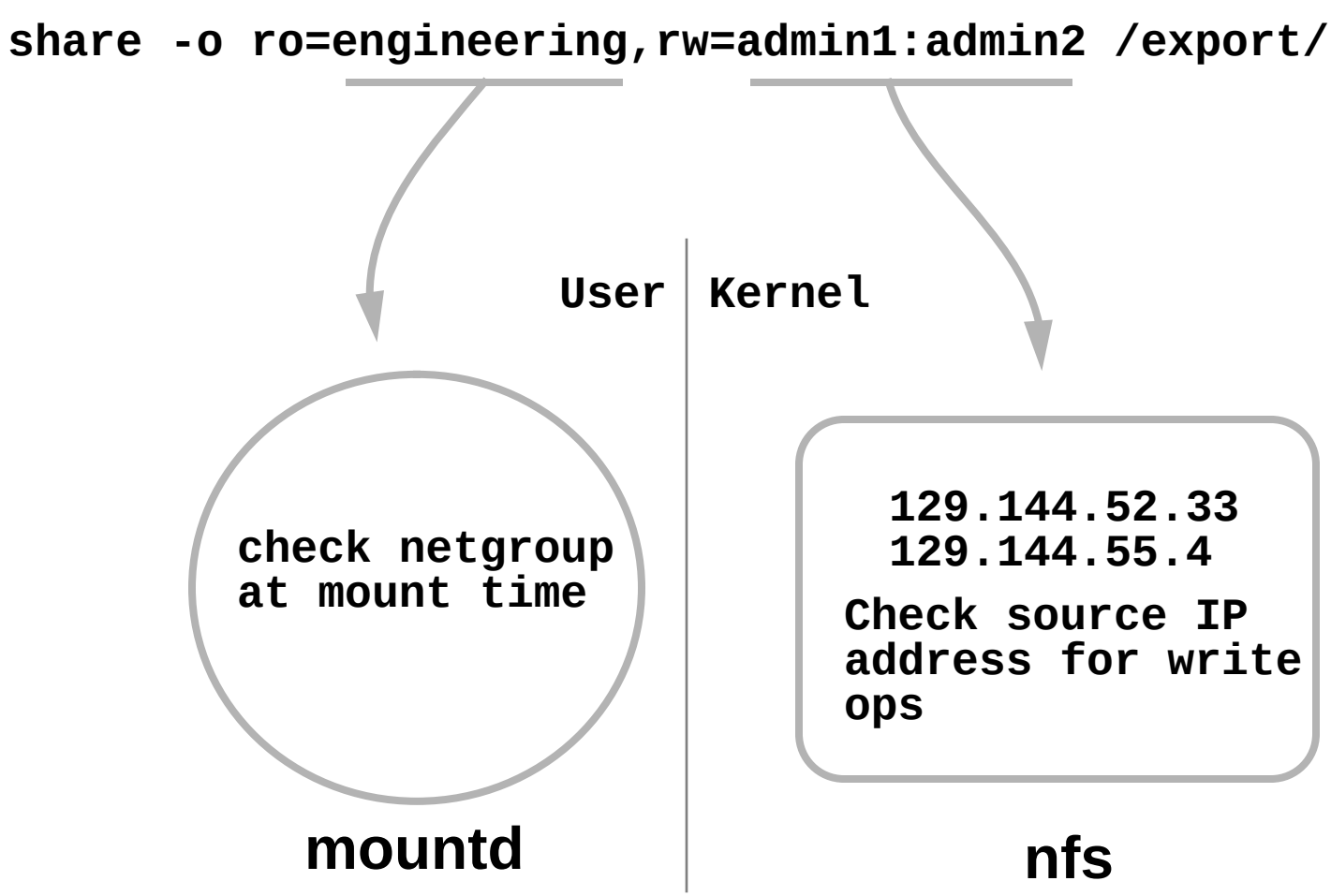
- **Need authentication flavors based on client's identity.**
- **Eliminate NFS filehandle security holes.**
- **More flexibile administration.**

Conventional Authentication

share -o rw=engineering /export/foo



Conventional Authentication (cont)



Chronic Problems

- **Filehandle Security Holes**
  - **Filehandles are shared secrets and can be guessed or snooped.**
  - **Cannot deny filehandle use**
- **Administration**
  - **Netgroup limitations**
  - **Client enumeration**

New Requirement

- **Multiple flavor support**

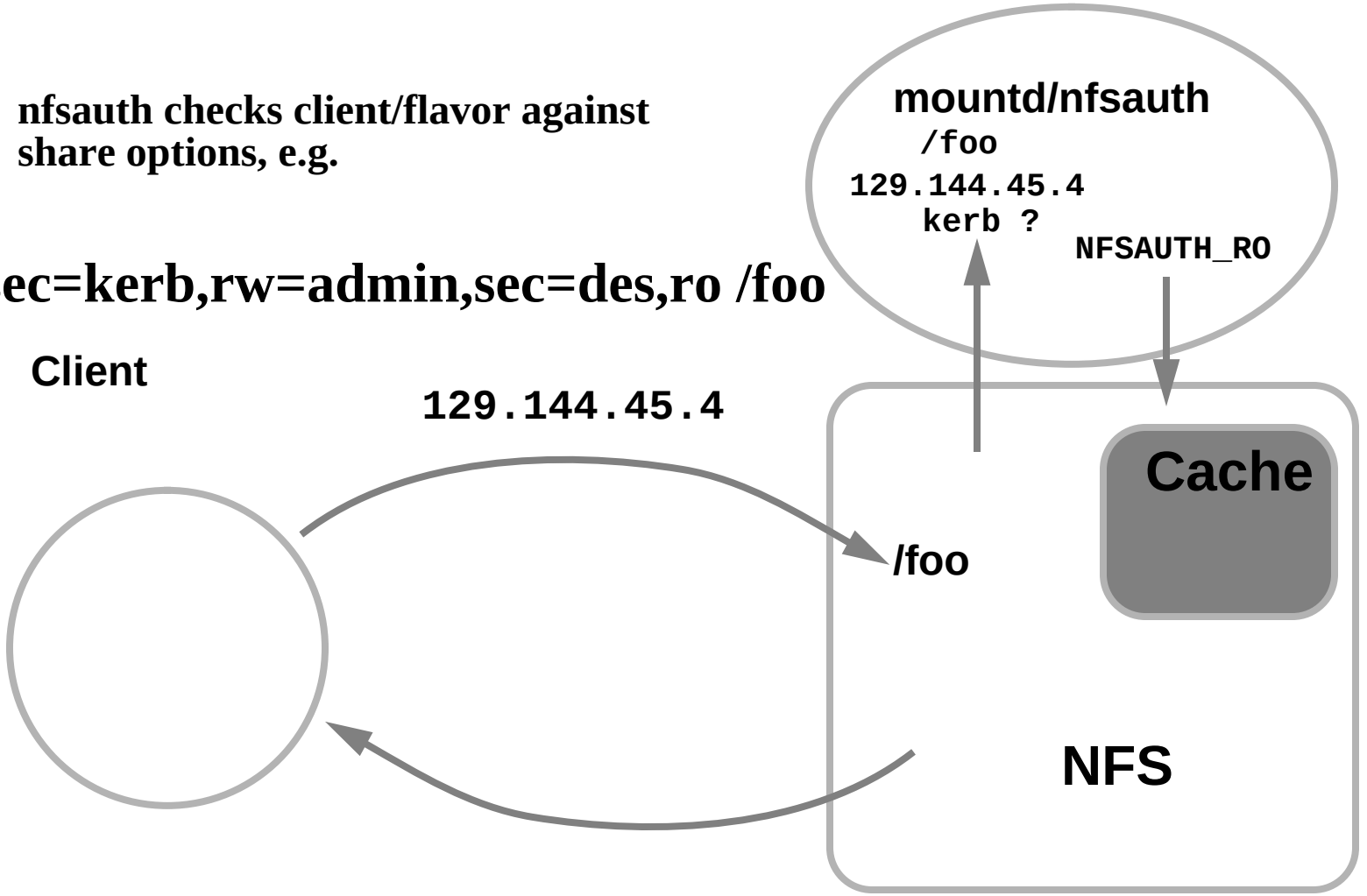
```
share -o sec=kerb,rw=admins,sec=unix,ro
```

- **Partitions client domain into groups of clients**
- **Each group has different permitted access flavors**
- **Need to dynamically check allowed flavors**
- **Old mount-time based model not sufficient**

nfsauth service

nfsauth checks client/flavor against share options, e.g.

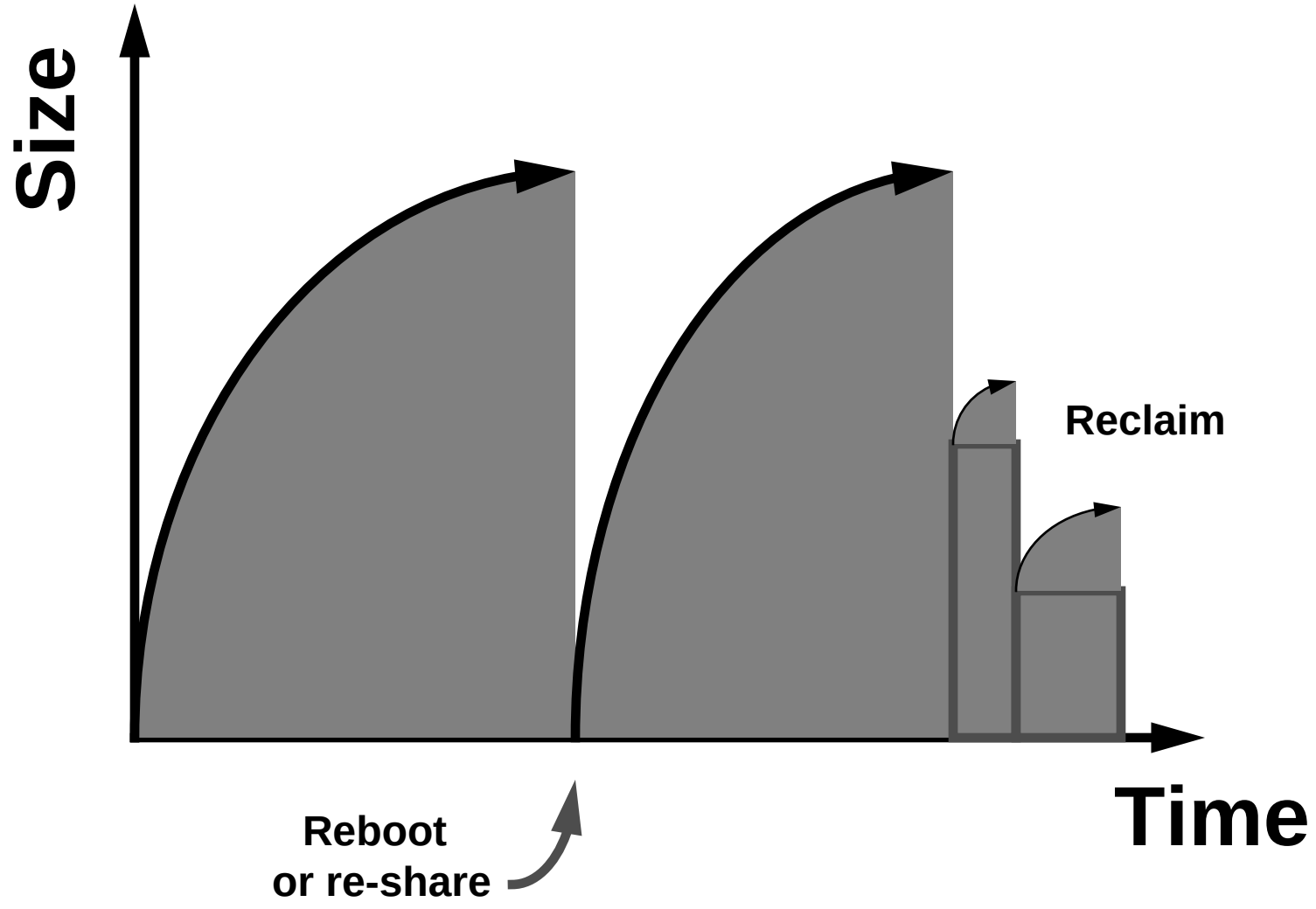
sec=kerb,rw=admin,sec=des,ro /foo



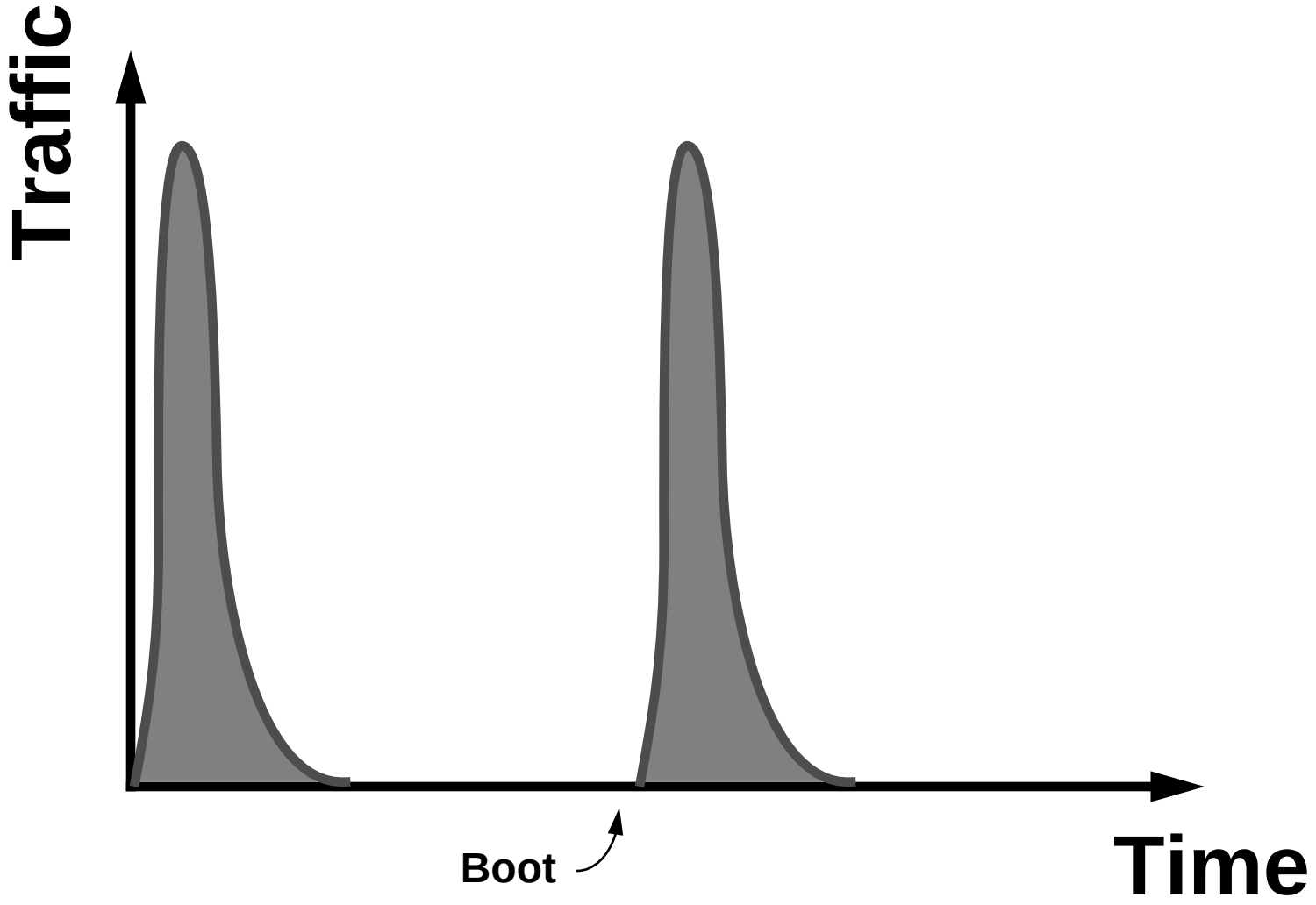
Auth Cache

```
Total cache  hits=194164  misses=212  reclaims=0
Cached client handles = 2

/export5: (unix rw)                                  (rw)

/export7: (unix root=,rw)                            (root=bang)
         1: ashoka:unix:rw
         8: contractor2:unix:rw
        11: ling:unix:rw
        22: universo:unix:rw
        23: teal:unix:rw
        28: mandrake:unix:rw


/export/root/ddfs1-c: (unix root=,rw=)               (rw=ddfs1-c,root=ddfs1-c)
        10: ddfs1-c:unix:root,rw

/export3: (unix root=,rw=)                           (rw=engineering,root=bang)
         1: ashoka:unix:rw
         8: contractor2:unix:rw
        11: ling:unix:rw
        22: universo:unix:rw
        28: mandrake:unix:rw
        30: aqua:unix:rw

/export1: (unix rw)                                  (rw)
```

Auth Cache Growth



Auth Nameservice Traffic



Public Filehandle

- **Public filehandle circumvents mount authentication:**

```
share -o public,ro=mygroup /export/foo
```

vs

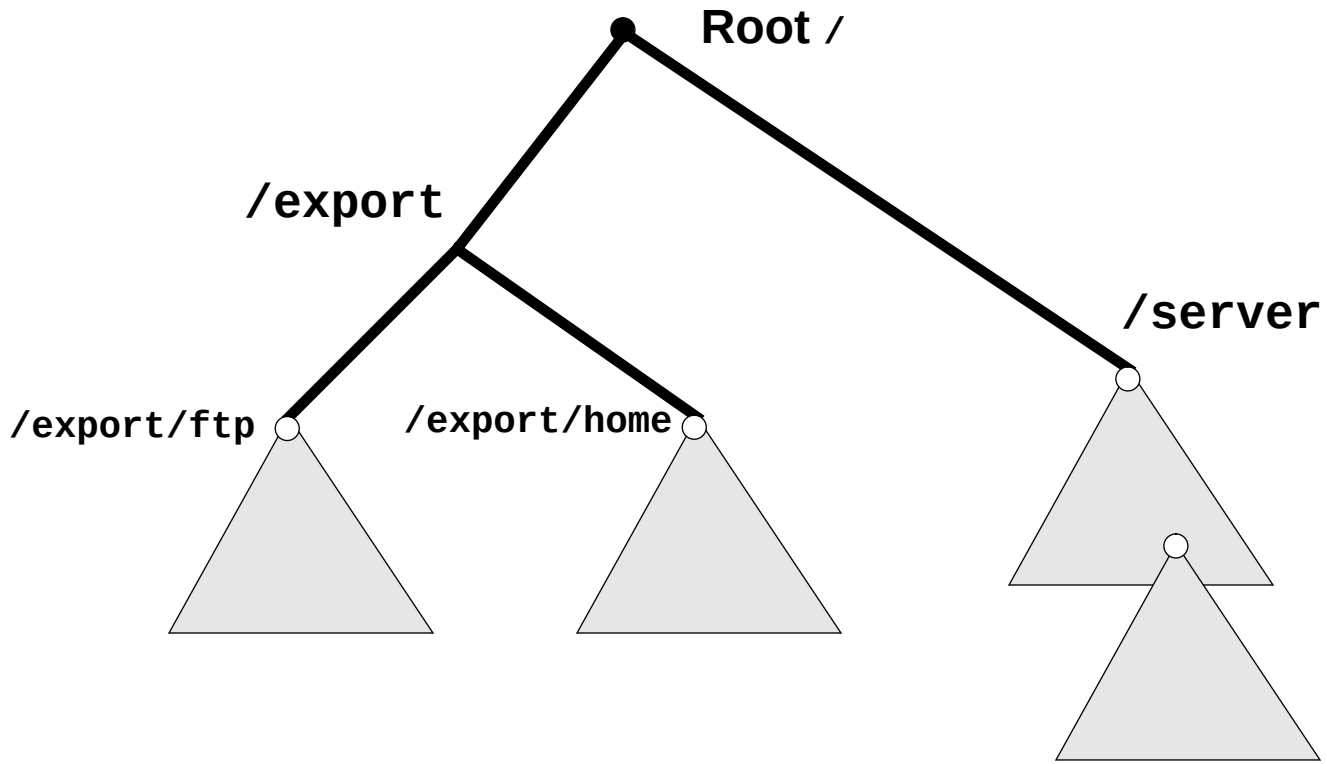
```
share -o public,ro /export/foo
```

- **Per-access checking authenticates *all* NFS requests so mount checking is not required.**
- **Per-access checking enables more liberal use of public filehandle.**

Public Filehandle (cont)

- Solaris default location of public filehandle at root.
- Clients can use public filehandle even if “public” not used in share command.
- Can use multi-component lookup to get a filehandle for any exported filesystem.
- Unrestricted use of MCL paths.
- Public filehandle + per-access authentication is equivalent to mount protocol for mounting.
- However, no auth flavor negotiation as in MOUNT V3.

Public Filehandle (cont)



Share Access Extensions

- **Host and Netgroup Membership**
  - **Require enumeration of all members & NIS or NIS+ support**

    `rw=rinky:dinky:engineering`
- **Negative Membership**
  - **Name those to be rejected**

    `rw=-dinky:engineering`
- **Domain Membership**
  - **By DNS domain.**

    `rw=.sun.com or rw=.edu or rw=.uk`
- **Network Membership**
  - **Network topology**

    `rw=@mtnview-eng:@129.144`

Summary

- **Support per-client auth flavors**
- **Closed security hole - filehandles can be public**
- **No restrictions on netgroups for ro & rw lists**
- **Can revoke a filehandle for chosen clients**
- **Supports negative access: `rw=-hackers,engineering`**
- **Group clients by DNS domain or Network**