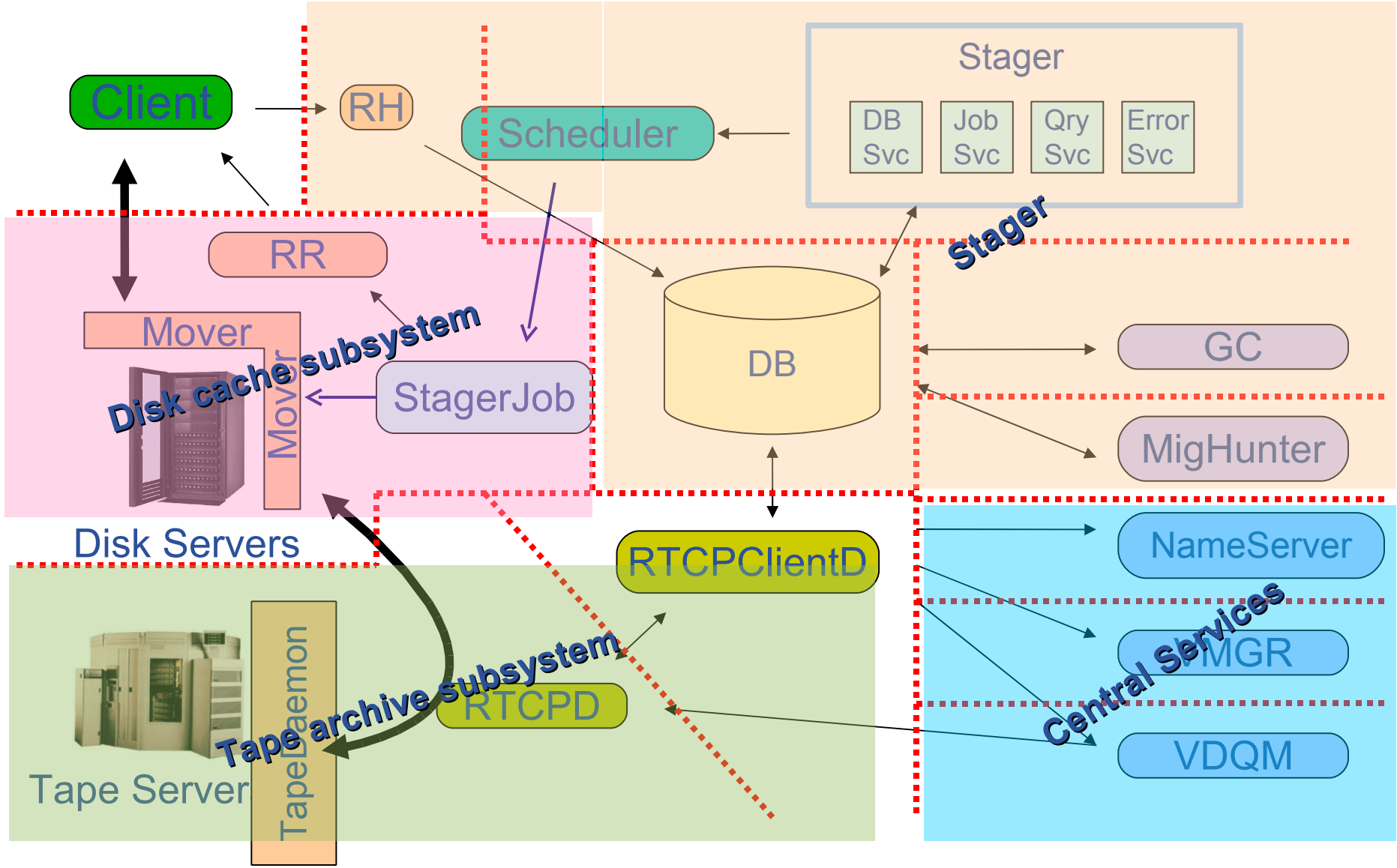
Castor2 Security Plans

Ákos Frohner
on behalf of SMD/DM/IT/CERN
MWSG, Bologna, 2008-03-28

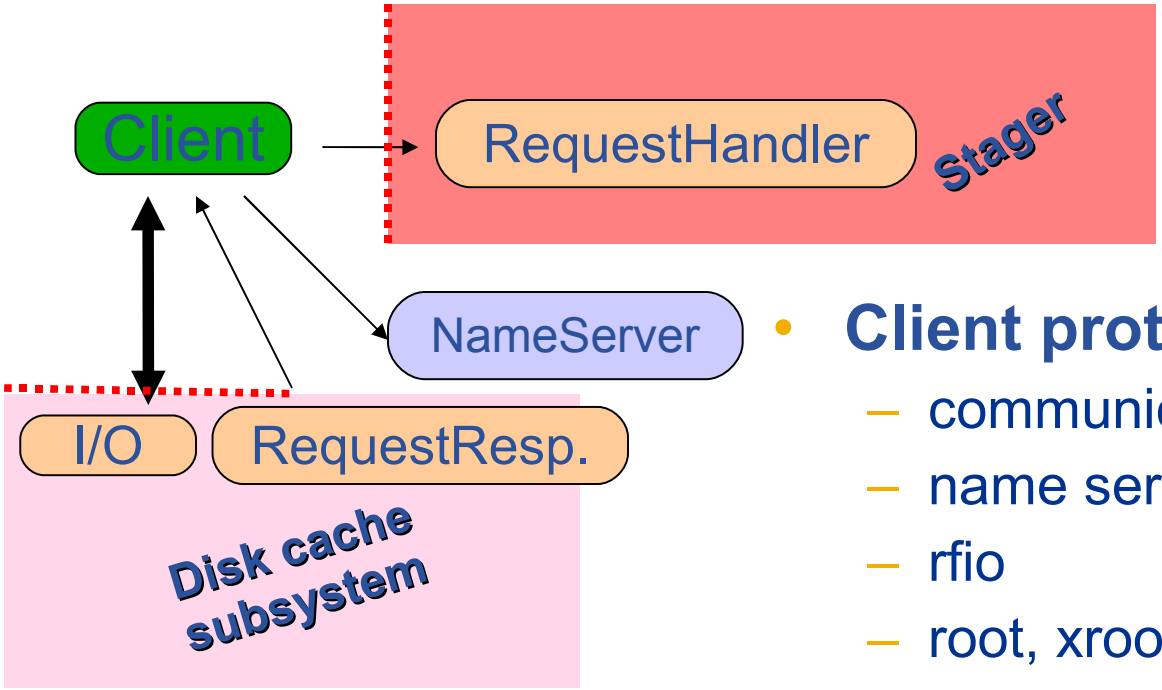
Castor2 Overview

- **Castor = CERN Advanced STORage manager**
  - hierarchical storage management (HSM):
    disk cache + tape backend
  - ~11PB data, ~91M files
  - file access via: rfio, root, gridftp and xrootd
- **Unix like directory hierarchy (/castor/cern.ch/...)**
  - namespace is managed by the Castor Name Server (CNS)
    file names, permissions, ownership
  - disk space is managed by the Castor Stager
    every file access has to be coordinated with the stager

Architecture



Client's View

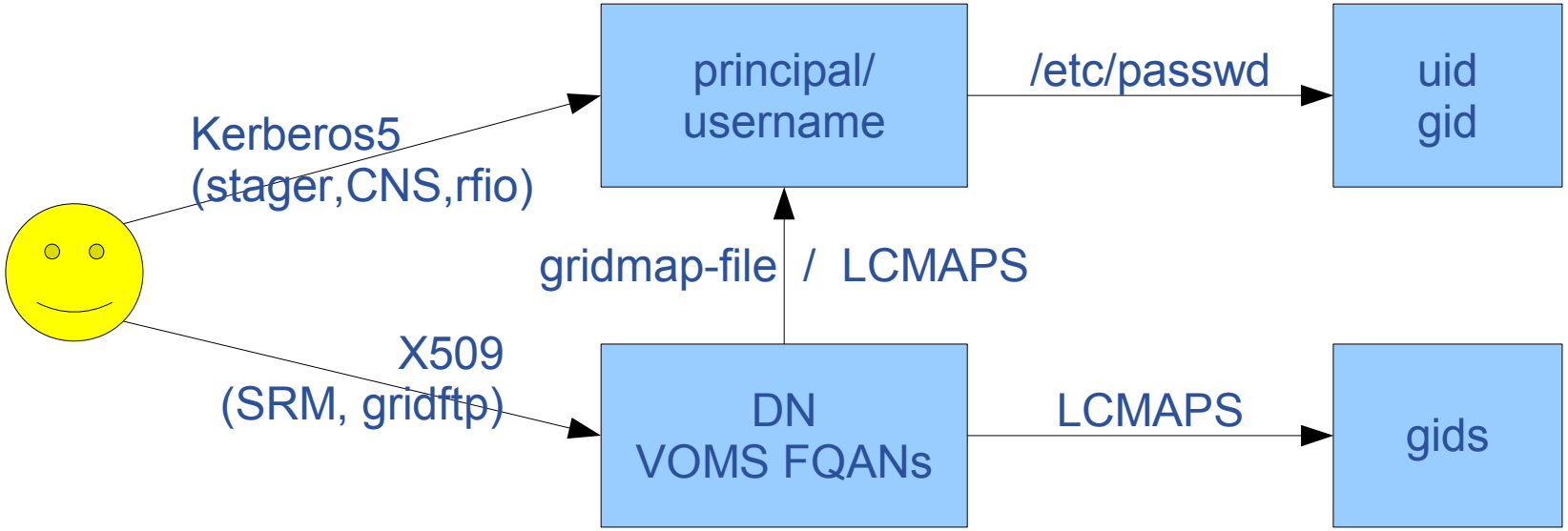


- **Client protocols:**
  - communication with the stager
  - name server
  - rfio
  - root, xrootd
  - srmv1 & srmv2 (X509 authn)
  - gridftp (X509 authn)
- **ownership and permissions by uid/gid**

Goals/Constraints

- **Client authn/authz is primary goal**
  - back-end services are in a controlled environment, so authn/authz of administrative actions comes later
- **Kerberos5 -- yes**
  - every CERN user has Kerberos principal
  - speed of Kerberos5 is better than X509
  - problematic part: external users via SRM & GridFTP
  - goal: X509&Krb5 in stager, CNS, rfio
- **Virtual UID/GID -- no**
  - stager scheduler requires real uid/gid
  - every internal user is already in the CERN user DB
- **Secondary groups – no**
  - passing secondary group information needs lot of changes

Authentication



- **stager, CNS and rfio currently uses uid/gid authn**
  - first goal is to improve this authentication
- **SRM and GridFTP uses X509 with pool accounts**
  - effective permissions are at group level
  - goal is to map individual DNs into individual uids
  - shortcut: CERN DN contains the username
- **How to add secondary group information in Kerberos?**

Authorization

- **Name Service**
  - current authorization is by uid/gid
  - mapping from Kerberos and X509 to uid/gid(s) solves the problem
- **Stager and SRM**
  - checks in the name service the file permissions
  - stores the uid/gid(s) with the request
- **I/O protocols (rfio, gridftp)**
  - one-time services are started for each request
  - requests are granted with a one-time token
  - the authenticated and mapped uid/gid is compared with the one in the request too
- **root, xrootd**
  - read-only access to world readable files yet

Further Information

- **Castor Documentation: http://castor.web.cern.ch**

Reminder: DPM/LFC permissions

POSIX style file and directory permissions

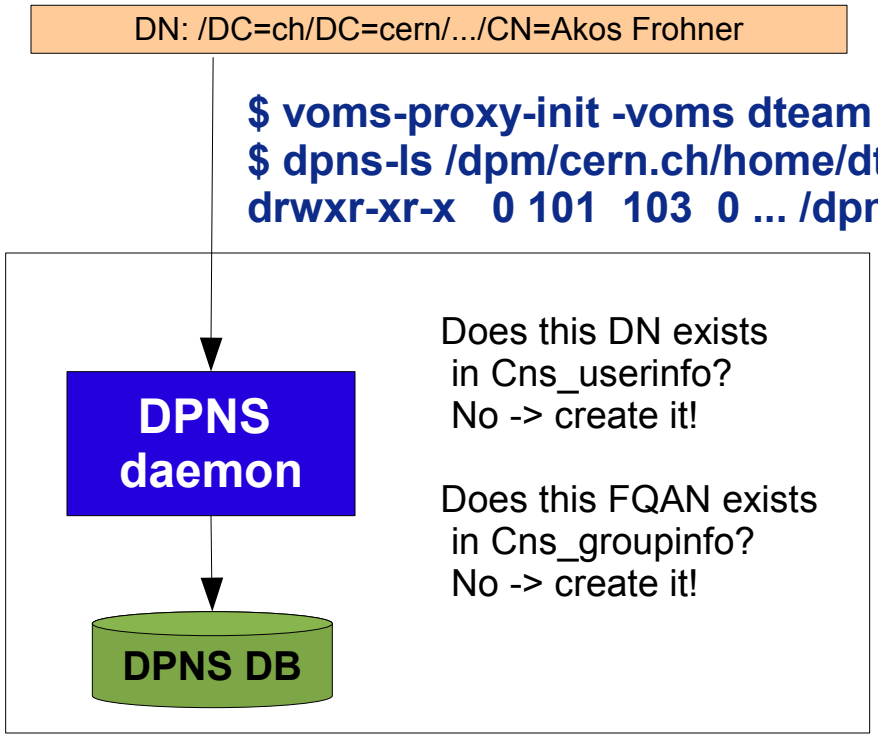
- **owner = DN of the creator**
- **group = first VOMS FQAN of the creator**
  - except, with set-group-id directories, where the group is inherited
- **basic read/write/execute permissions for user/group/others**
- **POSIX ACL:**
  - Access ACLs: set permissions for other users and groups
  - Default ACLs on directories: they are inherited by each entry created within.

Exact match: any of the user's DN or VOMS FQANs has to match exactly one of the permissions on a file.

Reminder: DPM/LFC exsmples

```
$ dpns-mkdir /dpm/cern.ch/home/dteam/akos
$ dpns-chmod 0755 /dpm/cern.ch/home/dteam/akos
$ dpns-setacl -m d:u::rwx,d:g::r-x,d:o:- /dpm/cern.ch/home/dteam/akos
$ dpns-setacl -m 'g:biomed:r-x,m:rwx' /dpm/cern.ch/home/dteam/akos
$ dpns-setacl -m \
    'u:/DC=ch/DC=cern/.../CN=Remi Mollon:rwx,m:rwx' /dpm/cern.ch/home/dteam/akos
$ dpns-getacl /dpm/cern.ch/home/dteam/akos
# file: /dpm/cern.ch/home/dteam/akos
# owner: /DC=ch/DC=cern/.../CN=Akos Frohner
# group: dteam
user::rwx
user:/DC=ch/DC=cern/.../CN=Remi Mollon:rwx  #effective:rwx
group::r-x                  #effective:r-x
group:biomed:r-x        #effective:r-x
mask::rwx
other::r-x
default:user::rwx
default:group::r-x
default:other::---
```

DPM/LFC Virtual uid and gid



- no need to create pool accounts
- no need to change the /etc/passwd file
- faster check on ACL than with string/pattern matching on DN/FQAN