DB Authentication and Authorization

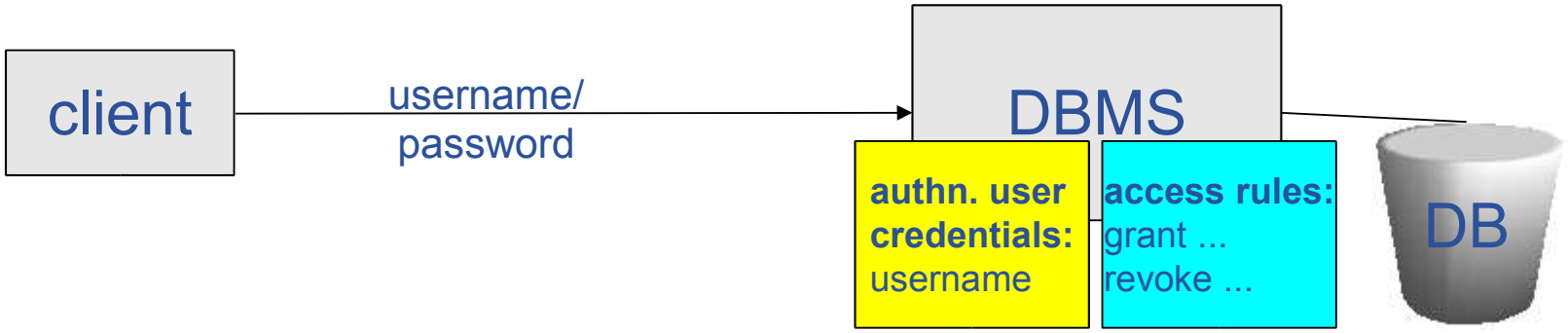
Ákos Frohner

Distributed Deployment of Databases, 2004 December 13-15

Contents

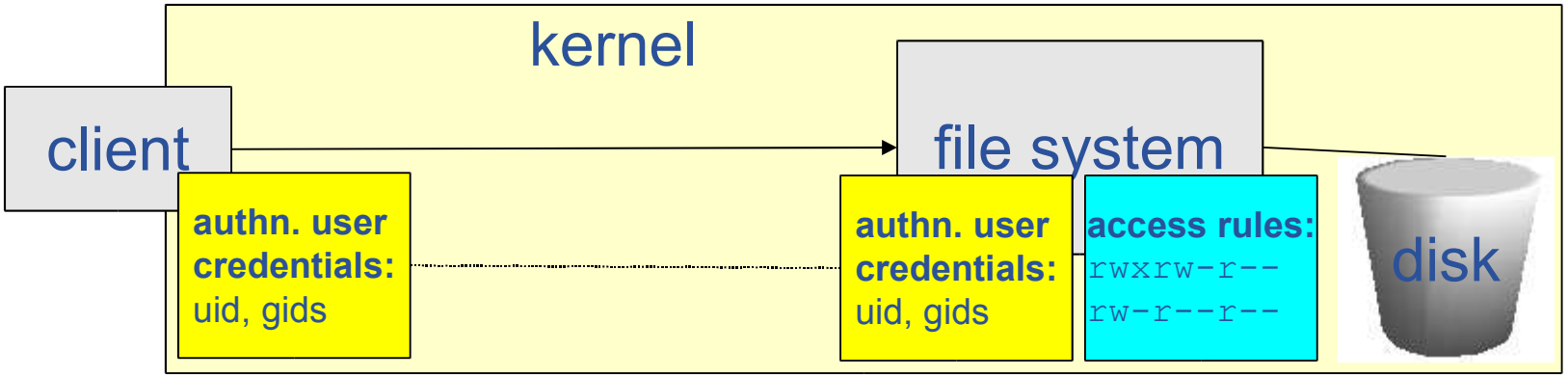
- Problem: the use case of DB access
- Example of Unix authorization
- Pieces of the grid solutions
- Gridified DB access use case
- Summary

DB Access



- client authenticates with username and password
- authentication: password checked
  – authenticated user credential is the *username*
- access rule for tables (objects) is granted on username:
  GRANT ... ON table TO 'joesmith';
- the client can access the table
  SELECT ... FROM table WHERE ...

File Access in Unix

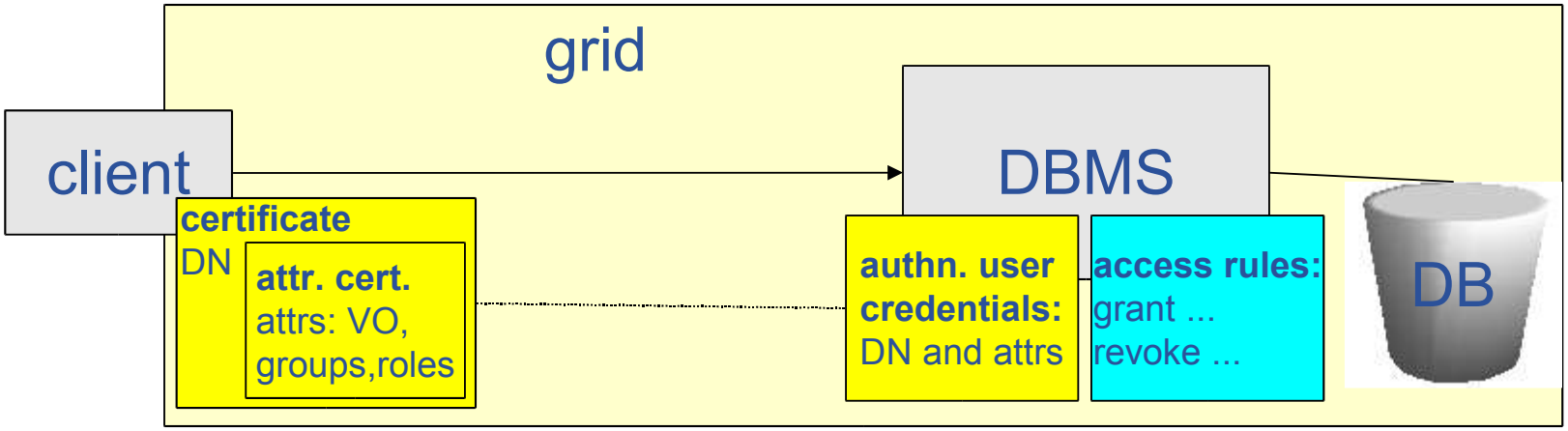


- client authenticates to the kernel
  – authenticated user credentials: userid and group ids
- authenticated credentials are protected by the kernel
  file system uses the credentials for authorization
- not only the userid, but also group ids are used in ACL

Grid Specialities

- authentication is based on X.509 certificates
  usually via an SSL connection
- short term, proxy certificates (RFC 3820)
  /C=CH/O=CERN/OU=GRID/CN=Joe Smith/CN=proxy
- additional attributes in attribute certificates (RFC 3281)
  user's VO: /egee
  user's role: /egee/Role=ProductionManager
- user credentials, DN and attributes,
  are granted by authorities (CA or VO management)
  and protected by cryptographic signatures
- trick: attribute certificate inside the proxy certificate
  extra attributes arrive via a normal SSL channel

DB Access with Grid Credentials



- client authenticates with proxy certificate
- authentication: validity of the certificate(s) checked
  – authenticated user credential is the *DN and attrs.*
- access rule for tables (objects) is granted on DN or attrs.:
  GRANT ... ON table TO '/C=CH/O=CERN/OU=GRID/CN=Joe Smith';
  - Oracle alternative, with pre-creation of users:
    CREATE USER user0001 IDENTIFIED BY '/C=CH/O=CERN/OU=...';
    GRANT ... ON table TO user0001;
- the client can access the table
  SELECT ... FROM table WHERE ...

Summary

If you already have SSL support, then we “only” need:

- attribute certificates; RFC 3281
- proxy certificates; RFC 3820
- agreement on tricks (e.g. how the attr. cert. is delivered)
- native integration with the authorization system
  e.g. DN can be used instead of DB username

... in other words: user management outside the DBMS.