Federating Data in the ALICE Experiment

Costin.Grigoras@cern.ch

Outline

- Data access methods in ALICE
- Storage AAA
- Storage monitoring
- SE discovery

Data access methods in ALICE

- Central catalogue of logical file names (LFN)
  - With owner:group and unix-style permissions
  - Size, MD5 of files
  - Metadata on subtrees
- Each LFN is associated a GUID that can have any number of replicas (PFNs)
  - root://<redirector>//<HH>/<hhhhh>/<GUID>
    - *HH* and *hhhhh* are hashes of the GUID
  - Same namespace on all storage elements
- Files are immutable on the SEs

Data access methods in ALICE (2)

- Exclusive use of xrootd protocol
- Jobs are (usually) only downloading configuration files with xrdcp
- Data files are accessed remotely
  - The closest working replica to the job
    - Jobs go to where a copy of the data is
- At the end of the job N (2..4 typically) replicas are uploaded from the job itself (xrdcp again)
- Scheduled data transfers only for raw data
  - Andreas' xrd3cp tool (disk server-to-disk server)

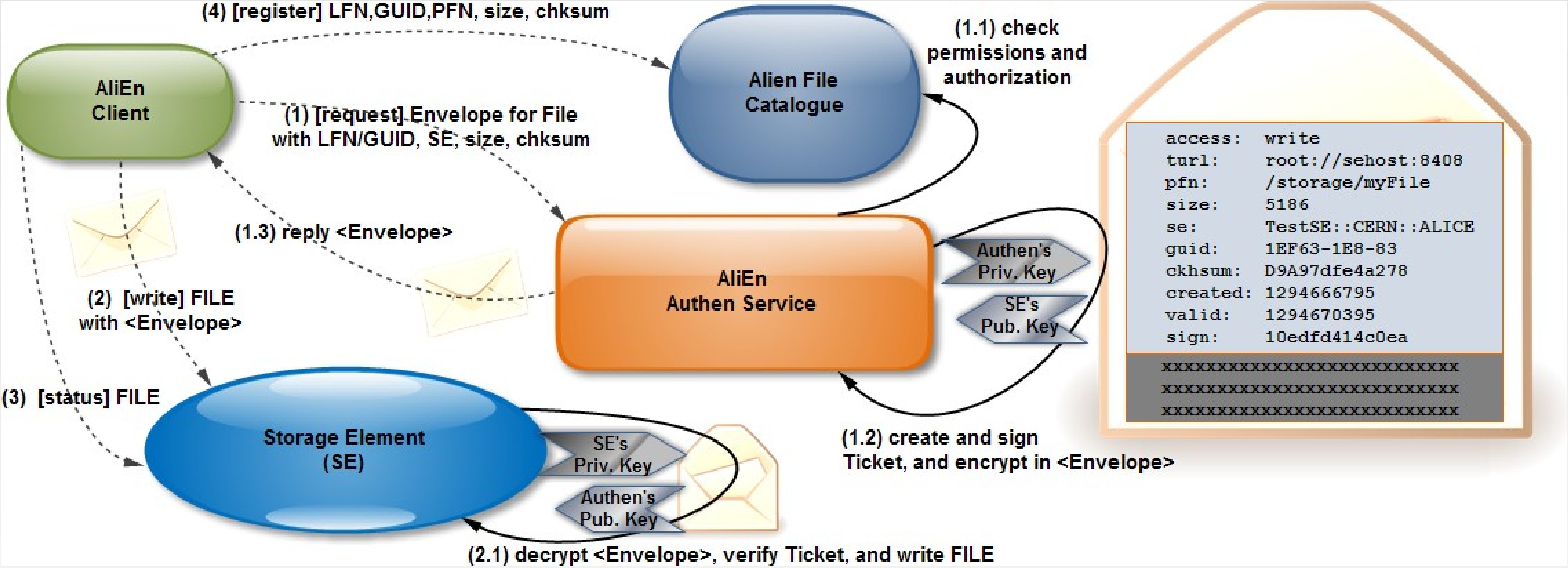
Some figures

- 59 disk SEs, 9 tape SEs (T0 and T1s)
  - 57x xrootd, 1x EOS, 2x DPM, 4x CASTOR, 4x dCache
- 25PB in 220M PFNs
- Average replication factor is 3
- 2 copies of the raw data on MSS:
  - Full copy at CERN T0
  - One distributed copy at T1s (full runs)

Storage AAA

- Storage-independent
- Handled centrally by the Authen AliEn service
- Checks client credentials and catalogue permissions and issues access tickets
    - XML block signed and encrypted by Authen
- The client hands these tickets to the respective storage and (for writes) notifies the catalogue of the successful operation
- Implemented as xrootd plugin

Storage AAA (2)

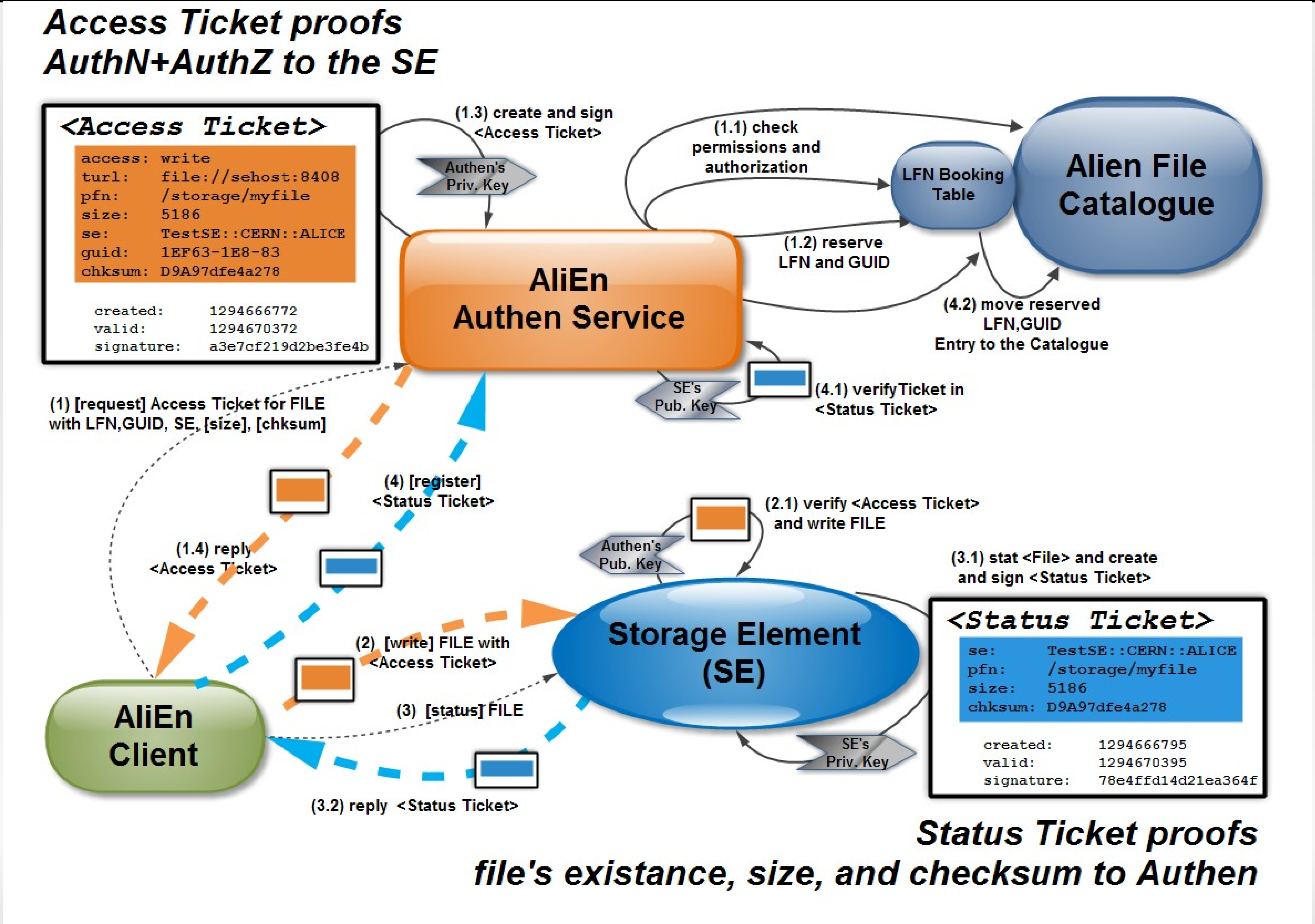

(4) [register] LFN,GUID,PFN, size, chksum
AliEn Client
Alien File Catalogue
(1.1) check permissions and authorization
(1) [request] Envelope for File with LFN/GUID, SE, size, chksum
(1.3) reply <Envelope>
AliEn Authen Service
Authen's Priv. Key
SE's Pub. Key
(2) [write] FILE with <Envelope>
(3) [status] FILE
Storage Element (SE)
SE's Priv. Key
Authen's Pub. Key
(1.2) create and sign Ticket, and encrypt in <Envelope>
(2.1) decrypt <Envelope>, verify Ticket, and write FILE

```
access:  write
turl:    root://sehost:8408
pfn:     /storage/myFile
size:    5186
se:      TestSE::CERN::ALICE
guid:    1EF63-1E8-83
ckhsum:  D9A97dfe4a278
created: 1294666795
valid:   1294670395
sign:    10edfd414c0ea
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Storage AAA – in deployment

- Similar to what is in production now
- Simplified tickets
    - Less text, just signed (no encryption any more)
- Introducing storage reply envelopes
    - Size and checksum of what the server got
        - Signed by the storage and returned by xrdcp, xrdstat
    - When committing a write the above must match what was booked
    - Can later recheck the files for consistency

Storage AAA – in deployment (2)



Access Ticket proofs AuthN+AuthZ to the SE

```
<Access Ticket>
access:  write
turl:    file://sehost:8408
pfn:     /storage/myfile
size:    5186
se:      TestSE::CERN::ALICE
guid:    1EF63-1E8-83
chksum:  D9A97dfe4a278

  created:     1294666772
  valid:       1294670372
  signature:   a3e7cf219d2be3fe4b
```

AliEn Authen Service

Alien File Catalogue

LFN Booking Table

Storage Element (SE)

AliEn Client

(1) [request] Access Ticket for FILE with LFN,GUID, SE, [size], [chksum]

(1.1) check permissions and authorization

(1.2) reserve LFN and GUID

(1.3) create and sign <Access Ticket>

Authen's Priv. Key

(1.4) reply <Access Ticket>

(2) [write] FILE with <Access Ticket>

(2.1) verify <Access Ticket> and write FILE

Authen's Pub. Key

(3) [status] FILE

(3.1) stat <File> and create and sign <Status Ticket>

SE's Priv. Key

(3.2) reply <Status Ticket>

(4) [register] <Status Ticket>

(4.1) verifyTicket in <Status Ticket>

SE's Pub. Key

(4.2) move reserved LFN,GUID Entry to the Catalogue

```
<Status Ticket>
se:      TestSE::CERN::ALICE
pfn:     /storage/myfile
size:    5186
chksum:  D9A97dfe4a278

  created:     1294666795
  valid:       1294670395
  signature:   78e4ffd14d21ea364f
```

Status Ticket proofs file's existance, size, and checksum to Authen

Monitoring – host parameters

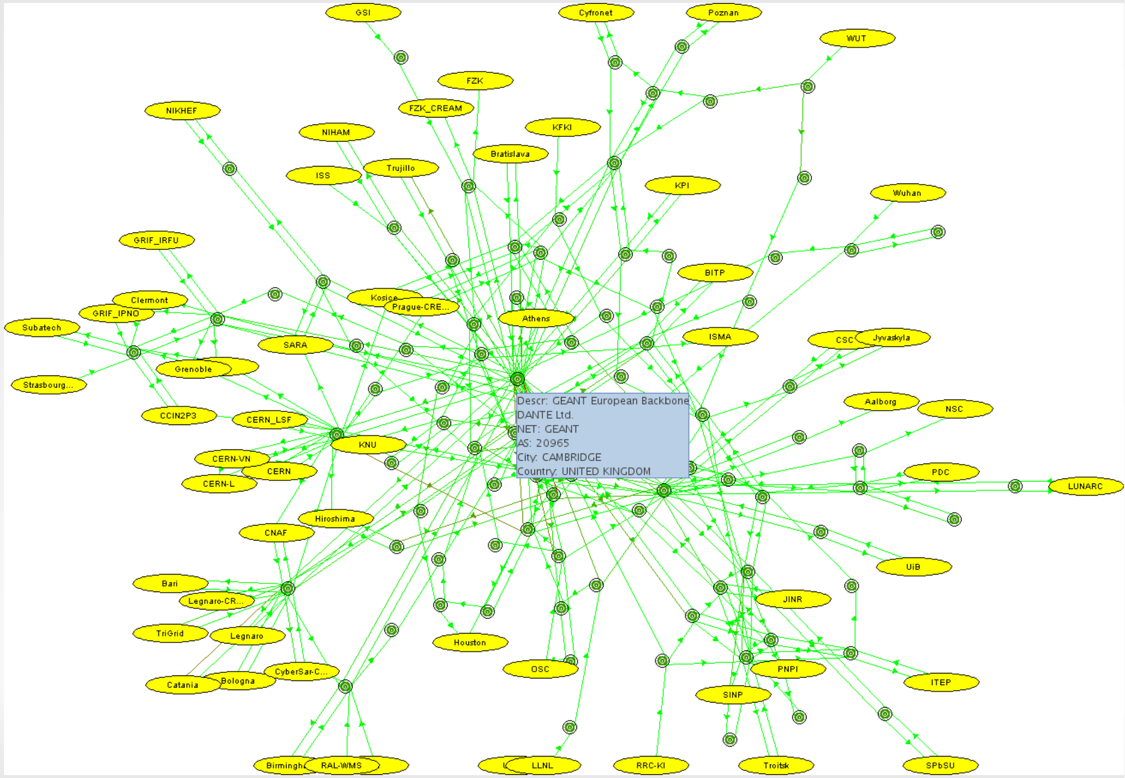
- Integrated in the overall monitoring of ALICE
- xrootd plugin package also brings a host and service monitoring daemon
- Monitoring data from xrootd and the daemon is sent to the site MonALISA instance
- Collected by the central repository and aggregated per cluster
  - http://alimonitor.cern.ch?571
- Under deployment: xrootd 3.2.2 with extended monitoring information

Storage monitoring – functional tests

- add / get / delete performed every 2h
  - From a central location
  - Using the full AliEn suite (like any user or job)
- Results archived for a “reliability” metric
  - Last week * 25% + last day * 75%
- Separate metrics for read and write

Network topology discovery

- Site MonALISA instances perform between every pair of them
  - Traceroute / tracepath
  - Bandwidth estimation
- Recording all details we get a good and complete picture of the network topology



Descr: GEANT European Backbone
DANTE Ltd.
NET: GEANT
AS: 20965
City: CAMBRIDGE
Country: UNITED KINGDOM

SE discovery

- Based on a dynamic “distance” metric from an IP address to a SE
  - Starting from the network topology
    - Same site, same AS, same country, continent...
  - Last functional test excludes non-working SEs
  - Altered by
    - Reliability
    - Remaining free space
    - A small random factor to assure 'democratic' data distribution

SE discovery (2)

- Reading from the closest working replica
  - Simply sorting by the metric for reading, including the non-working SEs, as last resort
- Writing to the closest working SEs
  - Each SE is associated a tag (“disk”, “tape”, “paper”)
  - Users indicate the number of replicas of each type
    - Default is “disk=2”
  - Not excluding the option of specific target SEs
  - Keep asking until the requirements are met or no more SEs left to try

Summary

- ALICE has the entire storage space federated
  - Via a central catalogue
- Specific AAA plugin in each SE
  - The storage layer doesn't have any idea about the users and their rights
  - No entity is to be trusted
- ROOT support as TAlienFile (working with LFNs)
- The distributed storage infrastructure is transparent to the users
  - Automatically managed

Other answers to Fabrizio's questions

- From a site we only need the entry point and not to delete the automatic alerts automatically
- The sites are oblivious to our activity, they just provide the boxes, no* customization or reporting* needed
- Uniform configuration and instrumentation (at least for the xrootd ones)
- Clearly separated components, recyclable
- Sites seem quite happy to run a script and forget about the ALICE storage afterwards