DECADE Requirements

draft-ietf-decade-reqs-03

Yingjie Gu, David Bryan, Y. Richard Yang, Richard Alimi

Outline

- -02 submitted on May 17
  - A couple of new requirements, in preparation for reviewers
- -03 submitted on July 11
  - Substantial changes/corrections based on reviewer comments (thanks!):
    - Dave McDysan
    - Akbar Rahman
- Open Issues
  - Metadata
    - Do we allow arbitrary metadata?
    - If yes, is it mutable or immutable?

Major Changes: -01 to -02

- New requirements added to fill some gaps
  - MUST provide a mode for data to pass over a Secure Transport
    - Confidentiality, integrity, and authentication
  - Unique names:
    - Data objects with different contents can't have the same name
  - Discovery: DECADE Client can locate suitable DECADE Servers
    - Suitability: DECADE Client can read/write and authorize other DECADE Clients to read/write to that Server
    - Discovery may yield 0 suitable servers (i.e., Discovery process fails)
  - Discovery: works if Clients behind NATs and Firewalls
  - Discovery: protocol should use existing protocols if possible

Major Changes: -02 to -03 (part 1)

- Removed requirements
  - A couple of duplicates had slipped in
- New Requirements
  - Default Access is no access
- Clarifications
  - Unique Names:
    - Large enough namespace, collisions at least statistically unlikely
    - Mechanism SHOULD be provided in DECADE to handle collisions
  - “Credentials not IP-based” → “Cryptographic Credentials”
  - Storage Status:
    - MUST provide stats aggregated over all authorized clients
    - MAY provide stats per authorized client

Major Changes: -02 to -03 (part 2)

- Cleanup of Open Issues
  - "Removal of Duplicate Data Objects" → Discussion section
    - Within server: Implementation detail
    - Across servers: Implementation detail, but there is a requirement for a redirect mechanism to help support this
  - "Gaming of the Resource Control Mechanism" → Discussion section
    - Protocols and implementations should be aware of it
- Security Considerations filled out a bit
  - Authentication:
    - DECADE Clients responsible for authenticating other Clients
    - Tokens communicate authorization, and Servers implement the checks
  - Data Encryption:
    - DECADE Servers store raw data; Clients may store encrypted data

Open Issue: Metadata yes or no?

- Premise:
  - It might be useful to allow Applications (via DECADE Clients) to attach custom name/value pairs to DECADE Data Objects
    - Annotate data based on context (e.g., chunk 123 of Stream A)
    - Resume after a restart, or resume playback on a different physical device
  - Does DECADE *need* to provide this?
- Alternatives:
  - Applications can store name/value pairs in data object itself (e.g., first K bytes)
    - Unappealing to design a list/query operation for the DECADE Server
  - Another layer on top of DECADE could provide this
    - Application-dependent, or maybe even a standard protocol down the line

Complications with Metadata

- If these name/value pairs are provided...
- Are they part of the object's name?
  - i.e., is hash also done over metadata?
  - Yes: more complex (but maybe not too bad) implementations
  - No: Implications on cross-server de-duplication
- Is metadata immutable?
  - Yes:
    - Not too bad to handle, aside from naming issue above
    - Are we giving up too much in flexibility?
  - No: Locking, caching issues (intermediaries and within applications)

Next Steps

- Pending resolution of metadata issue (here or on list)...
  - Is this ready for WGLC?