ΞMQ

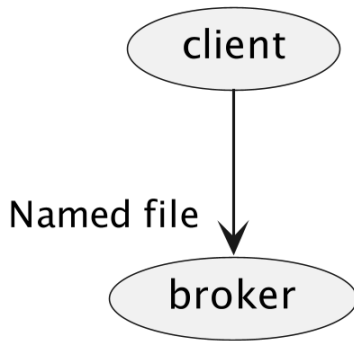# EMQX File Transfer

Ilya Averyanov, Andrew Mayorov

2023

# File Transfer
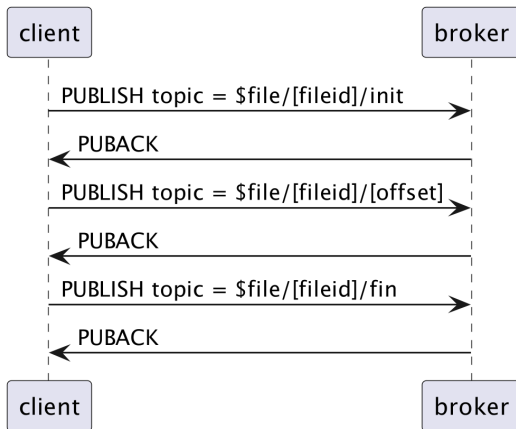Idea

# File Transfer
Constraints

▶ EIP 21

▶ Over MQTT

▶ Chunked

▶ No additional connections from client

▶ No subcription to special topics

▶ Resumable

# File Transfer
Protocol

```
client                                              broker
  │                                                   │
  │  PUBLISH topic = $file/[fileid]/init              │
  │ ────────────────────────────────────────────────▶│
  │                                                   │
  │                  PUBACK                            │
  │ ◀────────────────────────────────────────────────│
  │                                                   │
  │  PUBLISH topic = $file/[fileid]/[offset]          │
  │ ────────────────────────────────────────────────▶│
  │                                                   │
  │                  PUBACK                            │
  │ ◀────────────────────────────────────────────────│
  │                                                   │
  │  PUBLISH topic = $file/[fileid]/fin               │
  │ ────────────────────────────────────────────────▶│
  │                                                   │
  │                  PUBACK                            │
  │ ◀────────────────────────────────────────────────│
  │                                                   │
client                                              broker
```

File Transfer
Challenges

- ▶ Avoid storing data and even metadata in Mnesia
- ▶ Deal with clients migrating between nodes
- ▶ Deal with potentially long-running operations (involving disk and network I/O)
- ▶ Be minimally invasive*
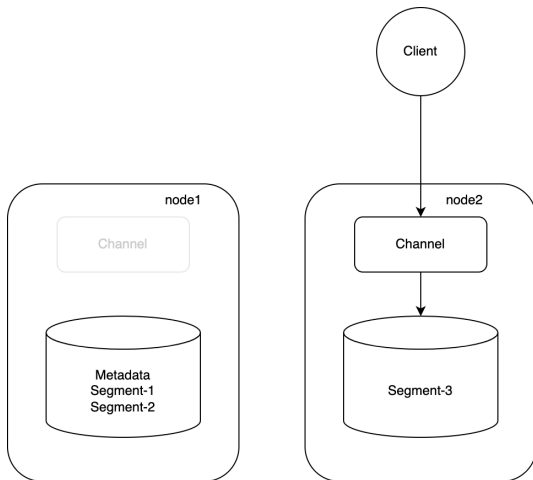
# Challenge 1: Avoid Mnesia

```
_build/emqx/rel/emqx/data/file_transfer
└── client-2
    └── f-1
        ├── frags
        │   ├── MANIFEST.json
        │   ├── SEG.0.100
        │   └── SEG.100.68
        ├── result
        │   └── test-file.txt
        └── tmp

5 directories, 4 files
```

# Challenge 2: Migrating clients

## Challenge 2: Migrating clients
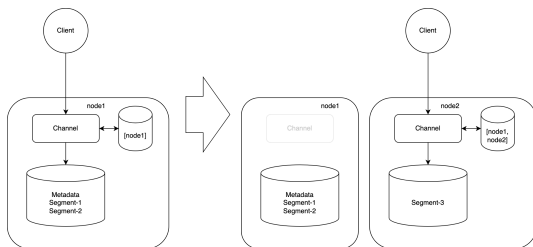Approach 1: Two step assemble



Step 1
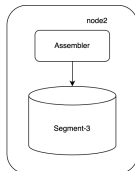
Step 2
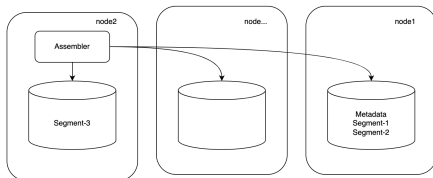
## Challenge 2: Migrating clients
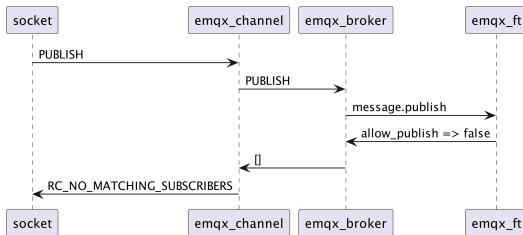### Approach 2: Replicated data

## Challenge 2: Migrating clients
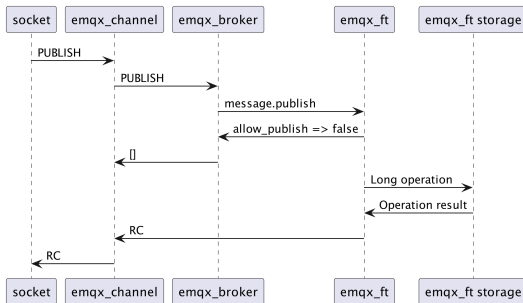Approach 2: One step assemble
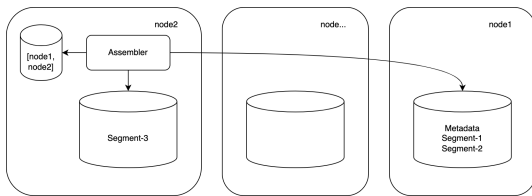
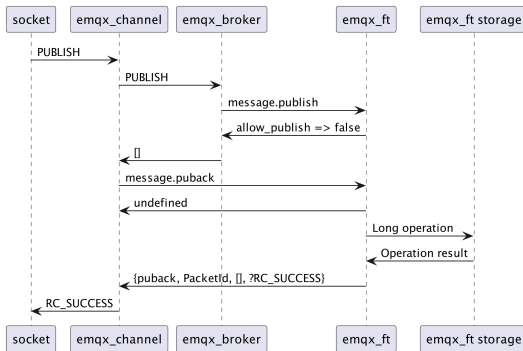# Challenge 3: Long-running operations
PUBLISH handling

ΞMQ

# Challenge 3: Long-running operations
PUBLISH handling: Desired

# Challenge 3: Long-running operations
PUBLISH handling: Implemented

File Transfer
Must have: TODO

- ▶ Ensure reliability of the storage part (testing, IO error handling, GC, etc.)
- ▶ Ensure reliability of the MQTT part (testing, MQTT client scenarios handling, etc.)
- ▶ Consider making all FT methods async, not only `fin`
- ▶ Improve observability (API)

File Transfer
Must have: Estimates

- **Ilya** - 7 story points
- **Andrew** - 7 story points

File Transfer
Should have: TODO

► Implement quotas: storage, bandwidth, tps
► Implement metrics

Thank you!