DLT Gateway Crash Recovery

draft-belchior-blockchain-gateway-recovery-00

R. Belchior (Técnico Lisboa)

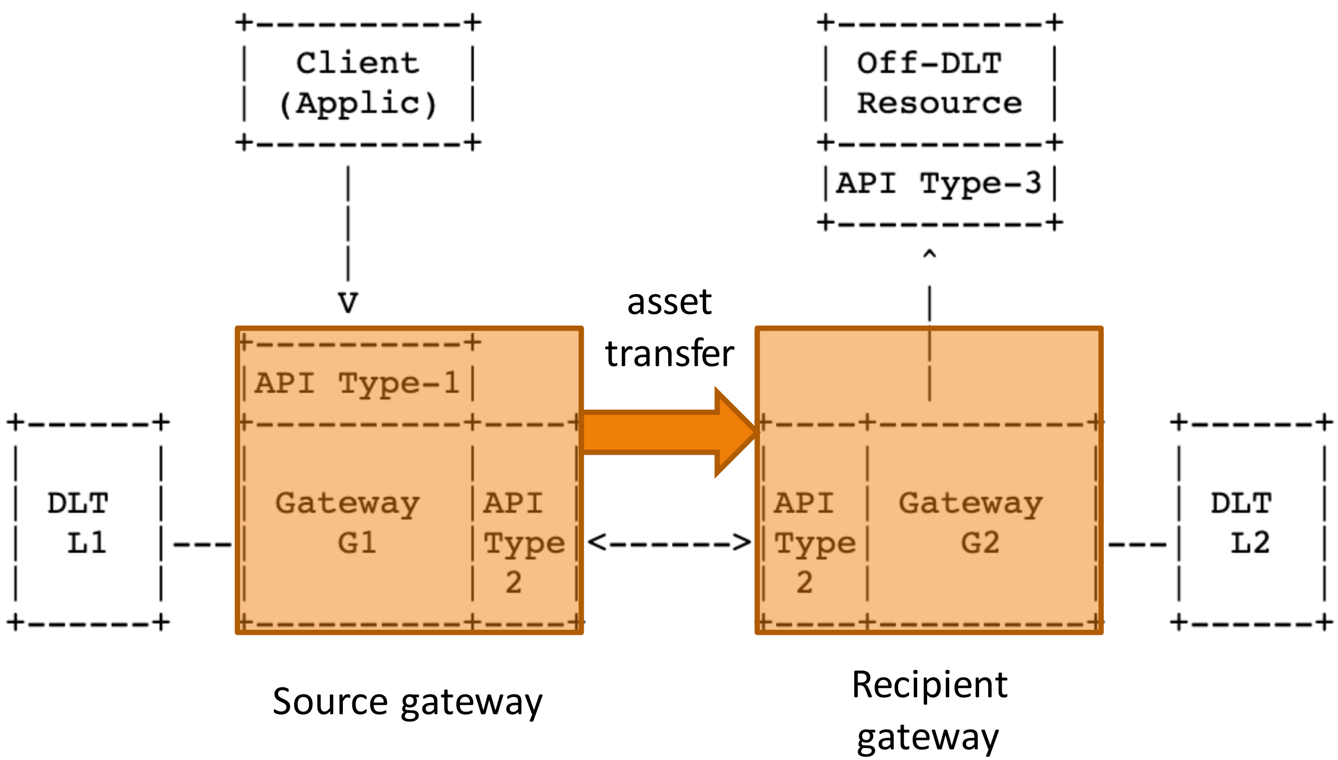
M. Correia (Técnico Lisboa)

T. Hardjono (MIT)

Introduction

- Gateways can fail (crashes)
- Fault-tolerant mechanisms: atomic commit protocol (ACP)
- We need logs

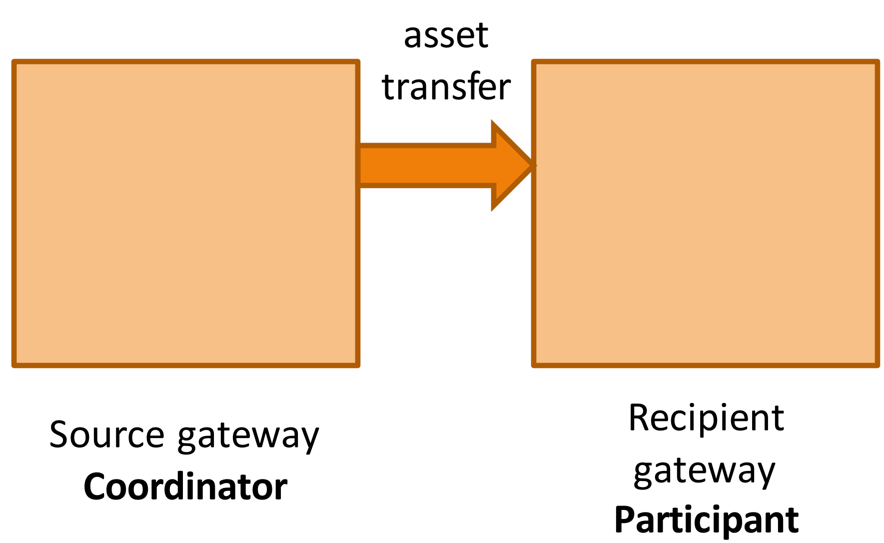
ODAP



Client
(Applic)
Off-DLT
Resource
API Type-3
API Type-1
asset
transfer
DLT
L1
Gateway
G1
API
Type
2
API
Type
2
Gateway
G2
DLT
L2
Source gateway
Recipient
gateway

Gateway Transfer Model

- We assume **Relay Mode**: Client-initiated Gateway to Gateway asset transfer
- ACP: Coordinator and Participants



Crashes and Recovery

- **ACP = 2PC or 3PC:** if a gateway crashes, the transfer may be blocked
- Any gateway before sending one of the messages of the ACP stores data about that message in persistent storage
- We consider two recovery models:

Self-healing mode

- **Self-healing mode:** we assume that after a crash, a gateway eventually recovers
- When a gateway restarts after a crash, it reads the state from the **log**, and continues executing the protocol from that point on
- Recovery: the recovered gateway informs the other party of its recovery and continues the protocol execution
- This is standard 2PC and 3PC

Primary-backup mode

- **Primary-backup mode:** we assume that after a crash, a gateway may never recover, but that this failure can be detected using a timeout
- Crash is **detected** using heartbeat messages and a conservative timeout value T
- **Backup gateway** does essentially the same as the gateway in self-healing mode: obtains the update log, reads it, and continues the process

Log storage

- 3 options
  - Locally in the computer's disk
  - External service, e.g., cloud
  - In the DLT of the gateway
- Log storage API that allows developers to abstract from the storage details
  - Save log entry
  - Get a log entry

Any ideas of similar IETF APIs to "steal" from?

Format of log entries

- Identification:
  - Session ID: unique identifier (UUIDv2) representing an ODAP interaction (corresponding to a particular flow)
  - Sequence number: scoped to the session ID
  - Source & Destination Gateway IDs: public keys + IPs?
  - Source & Recipient DLT IDs: e.g., Hyperldeger-Fabric-EU-Taxes

Format of log entries

- Timestamp: UNIX/ISO 8601

- Payload:

  - Contains subfields: Votes, Msg, Message type

  - Payload Hash: hash of the current message payload

Format of log entries

- <u>Optional:</u>
  - Message Digest: Gateway EDCSA signature over the log entry
  - Last Log Entry: Hash of previous log entry
  - Access Control Profile: the profile regarding the confidentiality of the log entries being stored
  - Logging profile: contains the profile regarding the logging procedure. If not present, a local store for the logs is assumed.