T2CP-AR: A system for transparent TCP active replication

Narjess Ayari, Denis Barbaron, FT R&D – LANNION, France
Laurent Lefèvre, Pascale Primet, INRIA / LIP, France
Agenda

- Fault tolerance and high availability: the big picture and the constraints
- Why T2CP-AR?
- What is T2CP-AR?
- T2CP-AR issues
- Conclusion & future directions
Fault tolerance and high availability

The big picture

- A key issue that has been considered in different areas
  - Internet routing, Internet servers, large scale computing, etc.
- FT frameworks use resource redundancy to provide the reliable execution of a service when its legitimate processing server goes down
- FT frameworks provide high availability features by means of
  - Fault detection mechanisms
  - Fault recovery mechanisms
- FT frameworks need to meet different challenges related to the fault handling procedure characteristics in terms of
  - Robustness
  - Performance
  - etc.
Fault tolerance and high availability

The constraints
Why T2CP-AR?

- Transport protocols rely on an explicit association between a service and its location for the wired Internet
  - When a server fails, the end-to-end communication is aborted

- TCP does not provide high availability capabilities
  - TCP does not distinguish between a packet loss due to a server failure or to a link failure
  - TCP reacts to lost or delayed segments by retransmitting them to the same remote end point
  - TCP tolerates short periods of disconnection no longer than few RTTs

- Several Internet services use TCP to control the end-to-end communications
  - RTSP, HTTP, FTP, VoIP, etc.
  - They have different high availability requirements and constraints
    - Allowed packet loss ratio, Delay sensitivity, etc.

- It is important to support transport level failover
What is T2CP-AR?

- **Active replication basic idea**
  - Make all the replicas receive the offered network traffic to a legitimate node and concurrently execute the service

- **A system for client transparent TCP active replication**
  - Efficiently active replicates flows among replicas
  - Is transparent to clients
  - Incurs a minimal overhead to the end-to-end communication during failsafe periods
  - Performs well during failures
What is T2CP-AR?

Topology
What is T2CP-AR?

General architecture
T2CP-AR detailed architecture

During failure free periods, the incoming traffic processor at the backup node

- Passively intercepts and filters the full duplex traffic originally intended to the legitimate server
- Modifies the resulting traffic before delivering it to the network layer
- Synchronises the states of the replicated TCP flows
  - Identify the sources of non deterministic behaviour at the transport level
  - Synchronize the flow states from when they are created
T2CP-AR detailed architecture

The outgoing traffic processor at the backup node

During failure free periods, the outgoing traffic processing at the backup node

- Ensures that only one server is replying to the client requests
  - Drops the data produced by the replica
    - Netfilter based.
- Uses the intercepted traffic flowing from the server to the clients to detect as early as possible any TCP datagram loss
  - Meet the synchronization requirements independently of the primary node failure
T2CP-AR detailed architecture

The traffic recorder at the gateway

- During failure free periods, the traffic recorder module at the gateway
  - Stores windows of the traffic flowing from the clients to the legitimate server
- Once a datagram loss is detected at the backup node, the datagram is recovered from the gateway
T2CP-AR detailed architecture

The failure detector at the gateway

The failure detector at the gateway

- Assumes fail-stop failures
- Is based on a heartbeat-like protocol
- Assumes different types of failures
  - Planned & unplanned failures
  - Application & host level failures
T2CP-AR detailed architecture

The failure recovery at the gateway

- Is triggered once the legitimate server failure is detected
- Provides
  - Network level high availability
    - Ensures the processing of the offered traffic related to new flows
  - Transport level high availability
    - Ensures the processing of the offered traffic related to the already established flows
T2CP-AR detailed architecture

The failure recovery module at the backup

It calls four functions

- **IP takeover handler**
  - Recover the IP @ of the primary node

- **TCP flow freezing**
  - Avoids packets loss during the failover

- **Active mode entry**
  - Once the takeover succeeds
    - Disable the incoming traffic interception, filtering and modification
    - Disable the outgoing traffic destruction

- **TCP connection takeover**
  - Announces the availability of the server
    - Positive advertisement window
Dealing with the non deterministic behaviour at the application level

- Sources are few: during signalling

Addressing multiple and heterogeneous flow based services

- A client session spans over multiple flows used for the signalling and for the data exchange all along the session lifespan
  - Voice over IP
  - Video streaming
  - etc.

- Use DPI to synchronize the application level states
Conclusion and future directions

- We proposed an active replication architecture of TCP flows
  - Requires few changes to the legitimate server
    - Does not influence the end-to-end throughout during failure free periods
  - Incurs a minimal failover overhead to the highly available end-to-end communication

- We aim to use the active replication concept to provide high availability
  - At the entry & inside a cluster of networked servers
Thanks
Any Questions?

Contact: narjess.ayari@orange-ftgroup.com
Backup slides
The early TCP connection state synchronization

The diagram illustrates the sequence of events in the early TCP connection state synchronization process. The connection starts with a SYN=1, ISN=ISNc, <options> packet from the client to the primary server. The primary server responds with a SYN=1, ISN=ISNc, <options> packet. The client acknowledges with an ACK=1, SYN=1, ISN=ISNp, Ack=ISNc+1, <options> packet. The primary server acknowledges with an ACK=1, SYN=1, ISN=ISNb, Ack=ISNc+1, <options> packet. If the backup server is needed, it updates its state with TCP CB update on B.
The early TCP segment loss detection