Listen to the discussion:

1. Use of a secure channel in D.S.

2. Fault-tolerant service

3. Use of a syn. time clock

4. At most, once delivery

5. Au less than or equal to 0

6. In order to detect

7. Duplicate info.
c. Call: avoid extra communication so end (occurrence)
d. This is fine - if it is a ver
not be considered duplicated even if it is
It may happen that a msg may
Since this is not perfectly aligned
is less than receivers' local time minus C
is considered recent if the time stamp
When the message arrives at receiver, it
message of luck cos if any message (d)
Any chance in the system will
2. Authentication Ticket in Kerberos.

- How can server check whether the time has expired or not.

- With synchronous clock it is easy.

- The server can check against its local clock whether the time \( E \) has expired.

- Ok even if the clocks are not perfectly sync. since providing service for little longer or shorter than \( E \) is ok. 

\[ E \gg \varepsilon \quad |C_{i(t)} - C_{j(t)}| \leq \varepsilon \]
Summary:

- Use of clks is for performance & not for correctness.

  - "performance" - e.g., saving messages (reduce latency, energy, b/w)

  - Of course, clk sync consumes resources.
Order: Zebra, f.t. zebra, replica. Replica is processed.
8:00 AM: Replica reception.
Agreement: Every new friendly replica configuration.
- Replica Coordination
  - Replication

Time:
- 8:00 AM: Agreement
- The other replica domain exchange

2:5:00 AM: Approve and implement failover.
Note: Ordering can be relaxed in certain cases.

If the request is to be made by a client or a request could have been made to the other side, any by client or
in the order they were issued.

The given A.M. can be proceeded by an Rrequest issued by a single client to

02
original tolerance

sensors

feedback system

feedback from tolerance

the power supply

replication

output

replication

output

output