**CS550 Written Assignment 3 (WA#3)**

**FNU Mohd Sadique**

**CWID: A20380442**

**Chapter 7**

**1. A file is replicated on 8 servers. List all the combinations of read quorum and write quorum that**

**are permitted by the voting algorithm.**

**Ans.** The following possibilities of are (read quorum, write quorum) legal. (1,8), (2, 7), (3, 6), (4, 5), (5, 4), (6, 3), (7, 2) and (8, 1).

**2. Explain the difference between linearizability and sequential consistency, and why the latter is**

**more practical to implement, in general.**

Sequential consistency:

* Sequential Consistency cares about program order that Each process issues operations in program order
* Assumes all operations are executed in some sequential order.
* In Sequential Consistency, any valid interleaving is allowed.
* All processes are on the same interleaving.
* Each process preserves its program order.
* Nothing is said about most recent writes.
* All subsequent read ops should return the same result until the next write, regardless of the clients
* It is suited for transaction database.
* The ordering of operations preserves the program order of each client.
* Active replication can provide sequential consistency.
* With linearizability, the interleaving across all clients is pretty much determined already based on time.

Linearizability consistency:

* Linearizability cares about time
* Stronger than sequential consistency because of this condition If TS(x) < TS(y) then op(x) should precede op(y) in the sequence.
* It is for read/write operation.
* The ordering of operations is determined by time.
* Primary-backup can provide linearizability.
* Chain replication can also provide linearizability.
* With sequential consistency, the system has freedom as to how to interleave operations coming from different clients, as long as the ordering from each client is preserved.

**Sequential consistency is more practical to implement in general** because linearizability is a local property and sequential consistency is a global property. Linearizability is looking from the point of view of a subset of operations against a record where serializability is looking from the point of view of all operations against the whole system.

**3. Consider a system that combines read-your-writes consistency with writes-follow-reads consistency. Is this system also sequentially consistent? Explain your answer.**

Ans. A system that combines read-your-writes consistency with writes-follow-reads consistency is not sequentially consistent as to be sequential consistent all the process should agree on the same interleaving and each process should preserve its program order. In the above case combinations of read-your-writes and writes-follow-reads **would not be able to satisfy the** mentioned conditions.

**Chapter 8**

**4. Suppose we have a system with 99.9999% availability, how much downtime a year can it have?**

Ans. Downtime per year for a system availability with 99.9999% is 31.56 seconds

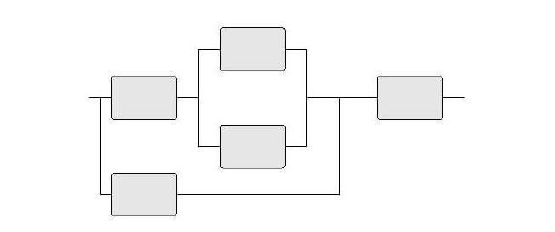
Availability = 99.9999%

Downtime = (100-99.9999)/100 \*(365 \* 24\*60\*60)

Downtime = 31.56sec

**5. Write the reliability expression *Rsystem(t)* of the following series/parallel system, assuming that**

**each of the five modules has a reliability of *R(t)*.**



**Ans.** Reliability expression Rsystem(t) for the above series/parallel system

Each modules has Reliability of R(t).

For the above parallel system the reliability =

For the above system the reliability =

For the above system the reliability =

For the above series-parallel system the reliability

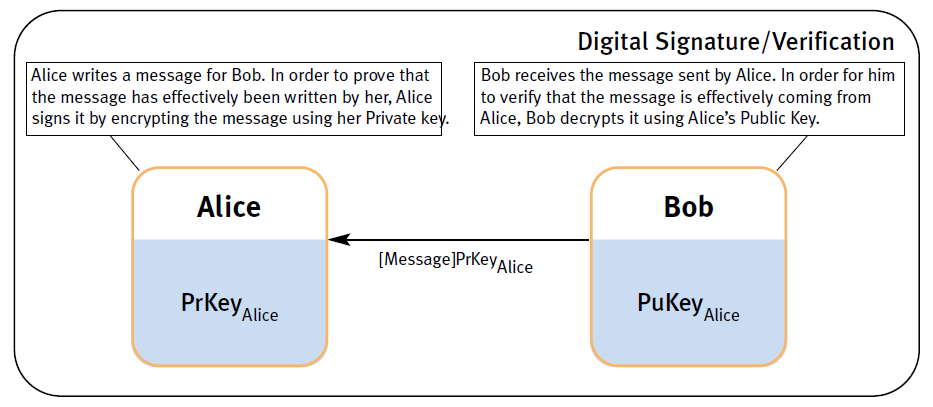
=

So, reliability expression:-

**Chapter 9**

**6. Devise a simple authentication protocol using signatures in a public-key crypto-system.**

Ans. If Alice wants to authenticate Bob, she sends Bob a challenge *R*. Bob will be requested to return *KB*(*R*), that is, place his signature under *R*. If Alice is confident that she has Bob's public key, decrypting the response back to *R* should be enough for her to know she is indeed talking to Bob.



**7. How are ACLs implemented in a UNIX file system?**

Ans. Access Control List (ACL) is implemented in a UNIX file system in each file has three associated entries: one for the owner, one for a group that is associated with the file, and one for everyone else. For each entry, the access rights can essentially be specified as read, write, execute.

**Chapter 11**

**8. Explain whether or not NFS is to be considered a distributed file system.**

Ans. Network File System(NFS) is not distributed file system, but merely a protocol that allows local file systems to become accessible to remote clients. In fact, most of the actual file system functionality is not implemented by NFS. Instead, it relies on the Virtual File System interface available in most operating systems.

**9. In UNIX-based operating systems, opening a file using a file handle can be done only in the kernel. Give a possible implementation of an NFS file handle for a user-level NFS server for a UNIX system.**

**Ans.** The problem to be solved is to return a file handle that will allow the server to open a file using the existing name-based file system interface. One approach is to encode the file name into the file handle. The obvious drawback is that as soon as the file name changes, its file handles become invalid.

**10. Despite that GFS (Google File System) scales well, it could be argued that the master is still a potential bottleneck. What would be a reasonable alternative to replace it.**

**Ans**. Considering that master uses a file name to look up a chunk server, we could also implement the master in the form of a Distributed Hash Table (DHT)-based system and use a hash of the file name as the key to be looked up. In this way, one would obtain a fully decentralized master.