CS 6210 Spring 2024 Test 3

(120 min Canvas Quiz)

Max Points: 96

Internet Scale Computing [24 points]

Giant-Scale Services [8 points]

1. [2 points] Traditionally, database systems use IOPS as a metric (number of I/O operations per second) as a figure of merit. Modern cloud-based systems seldom use this metric. Explain why.

2. [6 points] Assume that you are developing a giant scale service that has a corpus of data that is 150 TB. An individual server has a storage capacity of 10 TB, and you want to be able to serve up to 1000 simultaneous queries without any queuing delay, wherein each query requires a full harvest.

a. [3 points] How many machines will you need to meet these requirements? Explain your answer.

b. [3 points] Assuming that a request is successful only with full harvest, what's the best-case and worse-case scenario as to how many simultaneous requests can be handled without any queuing delay if 15 machines went down?

Map Reduce [8 points]

3. (4 points) For this problem assume that a mapper writes its intermediate results to a blob storage in the cloud; and a reducer gets it from the blob storage using RPC. The following items pertain to a given map-reduce application.

- Worker A is tasked with a map operation, which it completes successfully, storing R intermediate files to the blob storage.
- Worker B is assigned a reduce operation.
- Worker A experiences failure immediately after storing the intermediate files before informing the master that it has completed the assigned map task.

With succinct bullets explain what actions would be taken by the map-reduce framework.

4. [4 points] How does the MapReduce framework handle stragglers, and is there a risk of potential conflicts with this approach?

CDN Coral [8 points]

5. [8 points] In the following problem, assume that

• Put(x, y) denotes putting the key-value pair (key = x and value = y)
• Get(x) denotes getting the value corresponding to the key = x
• Assume a Coral system with node-ids 1, 2, 5, 10, 30, 75, 100, 150, 200 in which the "l" value for a node is set to 2; "β" is infinite.

• For simplicity, assume that all puts/gets in this problem using key-based routing go through node-ids: 30, 75, 100, 150, 200 irrespective of the source node of the request.

Charlie wants to use the Coral CDN to store and share a video featuring his adorable cat Apollo.

a. [3 points] To do this, Charlie issues a command Put(200, 10). What do 200 and 10 denote here? Where might this key-value pair be stored?
b. [3 points] Charlie's cat video gets popular enough that his friends Alex and Noah offer to host a copy of the video on their nodes. First, Alex issues the command Put(200,1). Next, Noah issues the command Put(200,2). Where will the key-value pair <200,2> be placed? Explain your answer.
c. [2 points] Diana hears about Charlie's popular cat video and wants to see it. She issues the command Get(200) from node 5. What value(s) will she get back? Explain your answer.

Real-time and Multimedia [12 points]

TS-Linux [4 points]

1. (4 points) Consider the implementation of firm timers in TS-Linux. With succinct bullets, discuss the tradeoffs associated with small and large values of overshoot parameter associated with firm timers.

PTS [8 points]

2. [4 points] Explain two similarities and differences between PTS and Unix Sockets.

3. [4 points] Consider three channels ch1, ch2, and ch3.

Ch1 contains elements with timestamps: 10, 20, 30, 40
Ch2 contains elements with timestamps: 10, 20, 30, 40
Ch3 is empty.

Consider a thread T which executes the following PTS code:

```
<item1,ts1> = Get(ch1, "oldest"); //returns the oldest element from ch1
<item2,ts2> = Get(ch2, "oldest"); //returns the latest element from ch2
Digest = Process(item1, item2); //code to process the two items fetched
Put (ch3, Digest, min(ts1, ts2)); // put digest with timestamp which is the minimum of ts1 and ts2
```

(a) [2 points] What is the timestamp associated with the above Put operation?

(b) [2 points] In PTS, is there a way to get corresponding timestamped items from ch1 and ch2 using a single api call instead of invoking Get twice? If yes, explain. (No credit without reasoning)

Failures and Recovery [27 points]

LRVM [10 points]

1. (10 points) Consider that a region R has been mapped from disk using the LRVM primitive "map()" to the virtual address space 0x10000000 to 0x1000ffff. The metadata m1 is located at the virtual address 0x10001000 with size of 4 bytes and initial value 10. The metadata m2 is located at the virtual address 0x10005000 with size of 4 bytes and initial value 11. For all the cases discussed below, data is immediately persisted to disk upon flush.
Consider the threads T1 and T2 as described below.

Thread T1:

```
1: begin_transaction(tid1, mode); // no_restore is NOT set
2:    set_range(tid1, 0x10000000 /*addr*/, 0x2000 /*size*/);
3:    m1 = 12;
4:    m2 = 13;
5: end_transaction(tid1, mode); // no_flush is NOT set
6: no-operation;
```

Thread T2:

```
1: begin_transaction(tid2, mode); // no_restore is NOT set
2:    set_range(tid2, 0x10004000 /*addr*/, 0x2000 /*size*/);
3:    m2 = 16;
4: end_transaction(tid2, mode); // no_flush is NOT set
5: no-operation;
```

a. (2 points) Assume T1 is the only thread that is currently running. The application crashes when T1 is at line 6. What is the value of m1 upon crash recovery? Justify your answer. (No points without justification)

b. (2 points) Assume T1 is the only thread that is currently running. The application crashes when T1 is at line 6. What is the value of m2 upon crash recovery? Justify your answer. (No points without justification)

c. (4 points) Assume T1 and T2 are both running concurrently. The application crashes when T1 is at line "6" and T2 is at line "5", respectively. What is the value of m2 upon crash recovery? Justify your answer. (No points without justification)

d. (2 points) Assume T1 and T2 are both running concurrently. The application crashes when T1 is at line "4" and T2 is at line "5", respectively. What is the value of m1 upon crash recovery? Justify your answer. (No points without justification)

RioVista [7 points]

2. [7 points] A server is built on top of RioVista that uses multiple data segments mapped to different regions of its virtual address space. The server wishes to make sure that changes to the multiple data segments should be atomic.

(a) [3 points] What steps should it take to ensure the desired atomicity?

(b) [4 points] Does RioVista need to do anything special to ensure the desired atomicity during crash recovery? Justify your answer.

QuickSilver [10 points]

3. (10 points) A file system is implemented on top of Quicksilver which uses the built-in recovery management. The file system consists of components that execute at each client workstation (denoted by C), a file server (denoted by F) that keeps meta-data for the files, and data servers (denoted by D1 and D2). Client machines are prone to failures while the server machines (F, D1, and D2) are robust and seldom fail. A client workstation C makes a file system call to the file server F. F in turn calls data servers D1 and D2 to satisfy the client request. The above actions result in the creation of breadcrumbs B1 at C; meta-data M1 at F, and intermediate data I1 and I2, at D1 and D2, respectively.

a) (2 points) what would be the structure of the transaction tree for the above call from C?

b) (2 points) What can be done to ensure the robustness of the file system to ensure proper recovery management?

c) (2 points) What would be logged at each node?

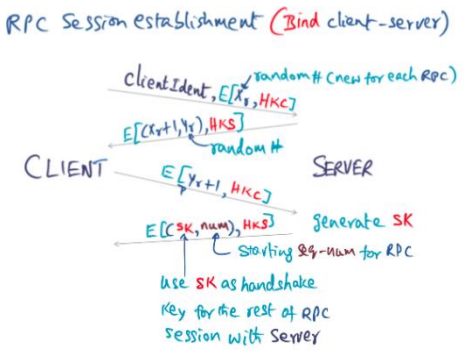
d) (4 points) C designates F as the coordinator for the transaction tree. What should happen if Workstation C crashes and then reboots again?

Security [15 points]

Security Principles, AFS [15 points]

1. [6 points] Assume that you are a student at CMU in the 80s. You walk up to your workstation implemented via the Andrew File System (AFS). Assume that you have been granted a username "usr6210", and you have set your password as "gatech" to login to the system. List the steps taken by Venus during the login process (i.e., what is exchanged between Venus and Vice in plaintext and cyphertext) to set you up for using AFS for RPC sessions for file transfers. For this question, you can use the "bind" protocol for mutual authentication as a primitive abstraction and not describe the message exchanges pertaining to that protocol in the login process.

2. [9 points] Shown below is the bind protocol which is at the core of mutual authentication.



(a) [3 points] Explain the significance of Xr, Yr, HKC, HKS presented in diagram with respect to the client and server authenticity.

(b) [2 points] Alex is in the middle of an RPC session with AFS. During this RPC session Alex sends a request to read a file. What key will be used Venus on behalf of this request?

(c) [2 points] Mallory sniffs Alex's read request and replays it AFS setting the source IP address to her machine. Her message arrives at the server ahead of Alex's message. Will the file system security be breached? If not, explain what exactly will happen that ensures the intended security?

(d) [2 points] Alex's message arrives subsequently at AFS. What will AFS do?

[Hidden] Potpourri [18 points]