Solutions to Exam 2
Fall 2002 15-712

Please write your name on the following line (and on each page of the exam):

____

This exam is open book and open notes.
You should write brief, precise, and legible answers. Rambling brain-dump answers may be punished twice — you run out of time writing them and we run out of patience reading them. :–)
If in doubt, list any assumptions you made about the question.
There are seven questions but we will only grade **five**. We will also grade the two bonus questions, but those do not count towards the five you need to complete.
List the five questions we should grade:

____

Name: ______________________________

1. The Hive system uses a memory protection "firewall" to protect against wild writes and various failure detection mechanisms to identify failed cells. However, it is important that one failed cell not cause all other cells to be viewed as failed. For each question below, provide an answer and a short explanation.

   (a) If the cells all observe the same detector signals (somehow magically), how many cells must exist to unquestionably identify two misbehaving (i.e., broken) cells?

   This question was a bit vague. If one assumed that the set of cells must convince a third party about which ones are failed, then a majority vote is needed and the answer is 5 ($2t + 1$). If one assumed that it meant that all correct cells see a broken cell as broken at the same time, then 3 cells ($t + 1$) are sufficient.

   (b) If an independent never-failing "super cell" observes all detectors, how many rounds of messages it must exchange with the cells to determine the right answer?

   Zero. Since the super cell can observe all results of failure detections at every cell detector, it can simply determine the current state based on the observed results.

   (c) If an independent never-failing "super cell" queries the cells about their observations, how many rounds of messages must it exchange with the cells to determine the right answer?

   The super cell goes through a single round of messages. The super cell issues a request to each cell detector, and after hearing an answer from each of the cells (or after a timeout if some of the cells don't reply), it takes a majority vote to determine the right answer (must be at least $2t + 1$ cells). Responses that timed out (i.e., that were never received), are counted as "wrong" answers.

Name: ______________________________

2. Stackable FSs provide a clean interface and mechanism for inserting FS layers into a system. Identify a safety problem that could arise in loading a new FS module (say, from a random website) into your desktop system and two approaches to dealing with it.

   A new FS module could manipulate system state arbitrarily (either deliberately or accidentally). For example, it could read or write unrelated kernel data structures, violating confidentiality or causing the system to crash, lie, or whatever.

   To prevent such behavior, a stackable FS infrastructure must employ additional protection mechanisms. Some options include software fault isolation, type-safe languages, code verification by a trusted third party, or separate hardware-enforced address spaces. Any of these mechanisms can be used to restrict the downloaded FS module to its structures and allowed interfaces.

Name: ___________________________________

3. A set of clients and servers collectively maintaining a Lamport clock could create a partial ordering among client-server messages. Explain how the "serializer thread" notion could be used in the system to easily achieve a total order even without the Lamport clock. In addition, identify a potential downside for system robustness.

   A serializer thread usually uses a queue to hold all incoming requests. The single thread takes requests from the head of the queue and processes them. A "process to completion" discipline would create a total order on all requests by simply executing them one at a time. A robustness problem with this is that performance would degrade rapidly when the system is busy, since no parallelism is allowed. Another is that the serializer is a single point of failure.

   Another approach (what we were actually thinking of) would be to query the serializer before each message to get a timestamp. This would allow much more parallelism, but have the same form of robustness problems as above.

Name: ____________________________________

4. Some researchers have mockingly claimed that exokernel systems are just very small μ-kernels. Explain why this is not a reasonable characterization and give an example how the xok/ExOS system differs from the μ-kernel model.

   Fundamentally, the μ-kernel mindset is about moving OS functionality from kernel space to user-mode servers, whereas the exokernel philosophy is about changing the boundary between what is "application" and what is "OS". Thus, a μ-kernel provides a set of base abstractions to virtualize the hardware and connect applications with OS servers. An ideal exokernel system would have no hardware abstractions and no OS servers.

   Some example differences are:

   (a) the device drivers are in the kernel, since the device driver interface is an expedient line for protection (and kernel vs. OS server is no difference from an application writer's point of view)

   (b) page faults are reflected to the applications that generate them (rather than to a "virtual memory server"). Applications keep their own page tables.

   (c) applications can see all hardware names and associations (e.g., buffer cache entries, their physical pages' addresses, their raw network packets, etc.)

Name: ____________________________________

5. Castro and Liskov describe the use of their BFT library to maintain replicas of files on multiple NFS servers, providing an NFS service that can survive Byzantine server failures. Explain what allows Publius to survive Byzantine server failures without needing the servers to use inter-server communication as described in the Practical BFT paper.

   The mechanism that allows Publius to detect Byzantine server failures is the ability of a reading client to (a) use a subset of shares fetched from independent servers, and (b) verify the integrity of data received from servers. Upon retrieving at least k correct shares from k different servers (with $k \leq n$ where n is the number of shares written) , the client can reconstruct the original key K. This key encrypts and decrypts the data file that is replicated on all Publius servers.

   Since the client verifies the integrity of each of the shares against the original name that was used to locate the share, there is no need for the Publius servers to reach BFT consensus. If one of the shares does not pass this integrity check, the client contacts another server to fetch a new $k+1$-th share. Once at least k shares are verified, the client can compose the original key K and decrypt a file.

Name: ____________________________________

6. The Gnutella client program broadcasts a machine's IP address when it requests a file. People have exploited this to publically expose some Gnutella users. Identify a shortcoming, or say why there is not one, with each of these attempts to provide privacy:

   (a) Use a fake IP address for each request.

   Broadcasting the IP address allows a Gnutella server to reply directly to the client's request. With a fake IP address, it will not be possible to route the response back to the client that initiated the request.

   (b) Share a "Gnutella proxy" (like a web proxy) with a community of user such that only a the proxy's IP address is sent with the Gnutella fetch request.

   This solution is feasible. The proxy keeps a mapping of users to the requests it makes on their behalf. This enables the proxy to forward the response back to client.

   This approach is still susceptible to traffic snooping and timing attacks. Encryption of communication with the proxy and careful mixing of requests and replies can address these, if enough clients exist. But, this approach relies on the integrity of the proxy (and its administrators), since the proxy sees everything you're doing with Gnutella.

   (c) Create an "anonymous re-Gnutella-er" network and get the Gnutella world to switch to it.

   This solution maintains the privacy of the users even if a subset of the re-Gnutella-er servers is compromised. While it may be relatively easy to deploy a network of these servers, it requires a substantial number of clients to download new versions of the Gnutella program.

Name: ____________________________________

7. The Spin OS uses type-safe language to enforce memory access control on kernel extensions. But an extension could still cause problems for the system by executing interminably after acquiring resources (e.g., locks). Explain how requiring extensions to execute within transactions could allow the system to solve this problem.

Since transaction processing maintains ACID properties, an extension that executes within a transaction can be safely aborted at any time if the system decides that it is executing interminably. As part of the abort, all resources acquired by the transaction are released and the state of the system is rolled back to the state before the extension started its execution.

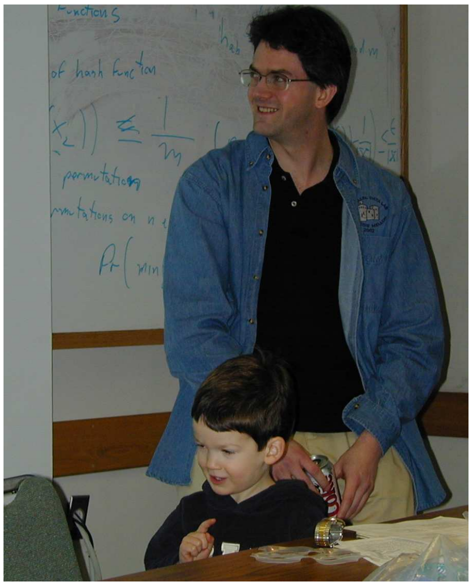
Name: ______________________________

8. **BONUS QUESTIONS**

(a) What was Timmy's Halloween costume this year?

Tigger.

(b) List one thing you learned from the guest lecturer depicted below.



$\{13\} \notin \mathbb{N}$ *(among other things)*