CS 524 Homework #5

Due: April 16, 2019

This homework is rather straightforward; it is essentially a reading assignment, but it is important to start working on this right away. Note: This assignment should take about five hours (the rest of the time to be spent on the Lab Assignment #3).

Reading assignment: Chapter 6 and Appendix Topic A.3

1. **(5 points)** Explain the motivation behind the two forms of server placement (rack-mounted servers and blade servers). What is sacrificed to make a blade server more compact than a rack-mounted server?

   Answer:
   - The motivation behind the two forms of server placement is that they are optimized to reduce their physical footprint and interconnection complexity (cabling spaghetti). Such optimization is necessary in the face of an ever-increasing number of servers that need to be put in the constrained space of a data center.
   - A blade server (or simply a blade) is even more compact than a rack-mounted server. The smaller form factor is achieved by eliminating pieces that are not specific to computing—such as cooling. As a result, a blade may amount to nothing more than a computer circuit board that has a processor, memory, I/O, and an auxiliary interface. Such a blade certainly cannot function on its own. It is operational only when inserted into a chassis that incorporates the missing modules. The chassis accommodates multiple blades. It also provides a switch through which the servers within connect to the external network. Worth noting here is that the chassis also fits into a rack much like a rack-mounted server.

2. **(5 points)** Why is the use of the Ethernet technology particularly important to the data centers? [Hint: What need does the use of the Ethernet effectively eliminate?]
   Answer:
   - Ethernet technology is particularly important to data center because of its potential to eliminate employing separate transport mechanisms (e.g., FC) for storage and inter processor traffic.
   - Data center networks have evolved from their original mission of providing connectivity between central computing systems and users.
   - "Horizontal" or intra-application traffic, generated in large part by the trend toward componentized software and service-oriented architecture (SOA). This traffic requires low latency but is still somewhat tolerant of packet loss.
   - Storage traffic, created by the migration of storage protocols onto IP and Ethernet (iSCSI and FCoE). Storage traffic is sensitive to latency, but it is even more sensitive to packet loss.

3. **(5 points)** Explain why NAS and SAN but not DAS are readily applicable to Cloud Computing. What are the limitations of DAS? Why is DAS suitable for keeping local data (such as boot image or swap space)?

Answer:

- With NAS, you're usually referring to something where you mount a shared storage space over a network using protocols like CIFS or NFS. The system on which these are mounted does not see them as local storage, it sees them as network storage. This is important because many programs will not allow the use of network storage for various things.
- With SAN, the storage space is mounted via iSCSI or Fiber Channel. You're still using a network to mount the space, but the protocols allow the system doing the mounting to see the space as local storage, thus allowing more programs to use it properly.
- DAS is simply storage directly attached (i.e, not via a network) to a system.
- DAS drawbacks include:
  1. Data not accessible by diverse user groups.
  2. Allows only one user at a time.
  3. High administrative costs.
- DAS is suitable for keeping local data because of High availability, high access rate due to Storage Area Network (SAN) absence, elimination of network setup complications, storage capacity expansion, data security and fault tolerance.

4. **(5 points)** Why is there a need for the *Phy* layer in the SAS architecture? How is it different from the physical layer?

Answer:

- Physical layer deals with the physical and electrical characteristics of cables, connectors and transceivers. Phy layer deals with line coding, out of band-signals and speed negotiation necessary for serial transmission. The name of the layer reflects the logical construct 'phy' that represents transceiver on a device.

5. **(10 points)** List the generic file-related system calls. Why in the NFS there is no RPC invocation for the *close <file>* system call? Under which circumstances other file operations may not result in an RPC invocation?

Answer:

- Genetic file-related system calls includes: open, read, and write
- There are two reasons for that in the NFS there is no RPC invocation for the close system call. First, the NFS protocol does not have the close routine because of the original stateless design of servers (which do not keep track of past requests) to facilitate crash discovery. Second, in this case there is no file modification.
- Under the circumstances when the information is stored in the client cache, which reduces the number of remote procedure calls and improves performance then, other file operations may not result in an RPC invocation

6. (**10 points)** What types of *connection topologies* are supported in *FC-2M?* Which of them is the most flexible? Why?

Answer:

- There are three types of connection which are supported in *FC-2M*. They are point-to-point, fabric, and arbitrated loop in which the fabric topology is most flexible.
- The fabric topology is most flexible because it involves a set of ports attached to a network of interconnecting FC switches through separate physical links. The switching network (or fabric) has a 24-bit address space structured hierarchically, according to domains and areas.
- An attached port is assigned a unique address during the fabric login procedure (which we will discuss later). The exact address typically depends on the physical port of attachment on the fabric (or switch, to be precise). The fabric routes frames individually based on the destination port address in each frame header.

7. **(5 points)** How does the FCF respond to a discovery solicitation from the ENode?
   Answer:
   - An ENode selects a compatible FCF based on the advertisement and sends a discovery solicitation at which the capability negotiation starts. Upon receiving the solicitation, the FCF responds to the ENode with a solicited discovery advertisement, confirming the negotiated capabilities.
   - Once receiving the solicited discovery advertisement, the ENode can proceed with setting up a virtual link to the FCF. The procedure here is similar to the fabric login procedure in FC.
   - Successful completion of the login page results in creation of a virtual port in the ENode, a virtual port on the FCF, and a virtual link between them.

8. **(5 points)** Please answer the following four questions:
   a) What features of TCP are leveraged in *iSCSI*?
      Answer:
      - Multiple iSCSI nodes may be reachable at the same address, and the same iSCSI node can be reached at multiple addresses. As a result, it is possible to use multiple TCP connections for a communication session between a pair of iSCSI nodes to achieve a higher throughput.

   b) Explain why these features are essential to *SCSI* operations.
      Answer:
      - Because they are reliable in-order delivery, automatic retransmission of unacknowledged packets, and congestion control.

   c) Why is not SCTP used in *iSCSI*?
      Answer:
      - The Steam Control Transmission Protocol (SCTP) is similar to TCP in its support for the features essential to SCSI operations. At the time of standardization of iSCSI, however, the SCTP was considered too new to be relied on and that's why SCTP is not used in iSCSI.

   d) Why does *iSCSI* has to be deployed over an *IPsec* tunnel when its path traverses an untrusted network?
      Answer:
      - Because they allow the existing IP-based infrastructure to be used, obviating the need to upgrade to more costly equipment and complex solutions such as Fibre Channel.

9. **(10 points)** What is *connection allegiance?* Explain how *iSCSI* sessions are managed.
   Answer:
   - SCSI communication takes place between an initiator and a target. The initiator sends a command to the target which then responds. At the end of the command the target returns a Status Code byte which is usually 00h for success, 02h for a Check Condition (error), or 08h for busy.
   - When the target returns a Check Condition in response to a command, the initiator usually then issues a SCSI Request Sense command in order to obtain more information. During the time between the reporting of a Check Condition and the issuing of the Request Sense command, the target is in the special state called the contingent allegiance condition.
   - iSCSI employs this scheme and with this scheme, the initiator can use any connection to issue a command but must stick to the same connection for all ensuing communications.
   - The iSCSI sessions need to be managed. A big part of session management is handled by the iSCSI login procedure. Successful completion of the login procedure results in a new session or adding a connection to an existing session

10. **(10 points)** Why the credential (as defined in ANSI INCITS 458-2011) itself cannot serve as a proof for access control? Give one example of a proof derived from the capability key.
    Answer:
    - At a minimum, it should be verifiable, tamper-proof, hard to forge, and safe against unauthorized use. A credential meets all but the last requirement; there is no in-built mechanism to bind it to the acquiring client or to the communication channel between the client and the storage device. (In contrast, a driver's license has a photograph of the driver to bind the license to the driver, although such a strong binding is not necessary for the problem at hand.) This is clearly not good, especially if the credential is subject to eavesdropping over an improperly protected storage transport. Thus, another proof scheme is in order.

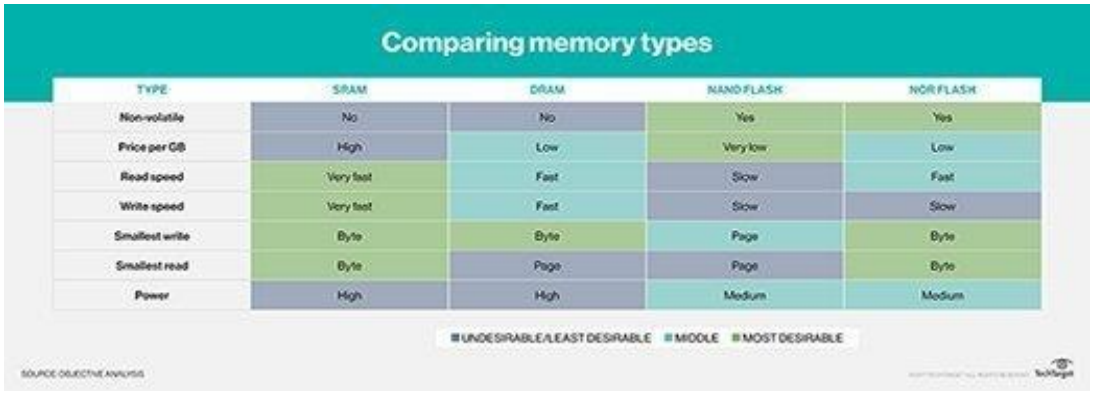
11. **(10 points)** Describe the three approaches to the block-level virtualization. Which approach is most suitable to the needs of Cloud Computing? What are the differences between the *in-band* and *out-of-band* mechanisms of the network-based approach along with their advantages and disadvantages.
    Answer:
    - The three approaches to block-level virtualization depending on where virtualization is done are as follows:
      1. The host
      2. The network
      3. The storage device
    - In the host-based approach, virtualization is handled by a volume manager, which could be part of the operating system. The volume manager is responsible for mapping native blocks into logical volumes, while keeping track of the overall storage utilization. Ideally the mapping should provide a capability to be adjusted dynamically to allow the capacity of virtual storage to grow or shrink according to the latest need of a particular application. A major drawback of the approach is that per-host control is not favorable to optimal

storage utilization in a multi-host environment, not to mention that the operational overhead of the volume manager is multiplied. In the storage device-based approach, virtualization is handled by the controller of a storage system. Because of the close proximity of the controller to physical storage, this approach tends to result in good performance. Nevertheless, it has the drawback of being vendor-dependent and difficult (if not impossible) to work across heterogeneous storage systems.

- In the network-based approach, virtualization is handled by a special function in a storage network, which may be part of a switch. The approach is transparent to hosts and storage as long as they support the appropriate storage network protocols (such as FC, FCoE, or iSCSI). Depending on how control traffic and application traffic are handled, it can be further classified as in-band (symmetric) or out-of-band (asymmetric)
- In-band is managing locally through the network itself, using a telnet connection to a router or by using SNMP-based tools (such as HP's Open View). In-band is the most common way to manage a network. However, for large or business-critical networks, in-band network management alone is not enough. If the network is down, you cannot use the network to reach the affected devices and resolve the problem. You need an alternate or secondary access path to get around the problem or to access the source of the problem—that is essentially what Out-of-Band Management (OBM) provides.
- If there is a problem with a device such a server or a router, and traffic cannot flow through the network, you need an alternate path to reach the network nodes even when the network is down. Management using independent dedicated channels is called OBM.
- OBM provides accessibility when an alternate path is needed to access the network nodes. In addition, OBM can provide coverage to many pieces of manageable equipment or intelligent devices that may not have a direct network connection to the data network, such as uninterruptible power supplies, PBX phone systems and intelligent thermal controls. For some of these intelligent devices that are not networked OBM may provide the only support and management tool. For mission-critical networks, in-band management tools are not enough. You need a secure remote emergency network access path to manage and troubleshoot when the device is not on the network, the device is not network manageable or the data network is down. That is the benefit of OBM console management. OutPost Sentinel's Emergency Network Specialist (ENS 8) and Cyber Command Center offer a suite of both in-band and OBM tools.

12. **(5 points)** Explain the difference (in terms of their capabilities) between the *NOR flash*- and *NAND flash* solid state drives.
    Answer:
    - NOR flash is faster to read than NAND flash, but it's also more expensive, and it takes longer to erase and write new data. NAND has a higher storage capacity than NOR.
    - NAND devices are accessed serially, using the same eight pins to transmit control, addressing and data. NAND can write to a single memory address, doing so at eight bits -- one byte -- at a time.
    - In contrast, older, parallel NOR flash memory supports one-byte random access, which enables machine instructions to be retrieved and run directly from the chip, in the same

way a traditional computer retrieves instructions directly from main memory. However, NOR has to write in larger chunks of data at a time than NAND. Parallel NOR flash has a static random access memory (SRAM) interface that includes enough address pins to map the entire chip, enabling access to every byte stored within it.



- NOR flash is also more expensive to produce than NAND. That, and its random access function, mean NOR is mostly used for code execution, while NAND is mostly used for data storage.
- NOR flash is most often used in mobile phones, scientific instruments and medical devices. NAND has found a market in devices to which large files are frequently uploaded and replaced, such as MP3 players, digital cameras and USB flash drives.
- Some devices use both NAND and NOR flash. A smartphone or tablet, for instance, may use did not embed NOR to boot up the operating system and a removable NAND card for all its other memory or storage requirements.

13. **(5 points)** What are the three limitations that stand in the way of deploying the *NAND flash* solid state drives in the Cloud?
Answer:
    - The three limitations that stand in the way of deploying the NAND flash solid state drives in the cloud are as follows:
        1. A write operation over the existing content requires that this content be erased first. (This makes write operations much slower than read operations.)
        2. Erase operations are done on a block basis, while write operations on a page basis;
        3. Memory cells wear out after a limited number of write–erase cycles.

14. **(10 points)** Explain the mechanism of *consistent hashing* used in *Memcached servers.*
Answer:
    - Depending on the size of DRAM available on a server, caching the workload data may need more than one server. In this case, the hash table is distributed across multiple servers, which form a cluster with aggregated DRAM. Memcached servers, by design, are neither aware of one another nor coordinated centrally. It is the job of a client to select what server to use, and the client (armed with the knowledge of the servers in use) does so based on the key of the data item to be cached. How should the hash table be distributed so that the same server is selected for the same key? A naïve scheme might be as follows:

s = H(K) mod n,

where H(k) is a hashing function, k the key, n the number of server, and s the server label, which is assigned the remainder of the division of H(k) over n. The scheme works as long as n is constant, but it will most likely yield a different server when the number of servers grows or shrinks dynamically—as is typically the case in Cloud Computing. As a result, cache misses abound, application performance degrades, and all servers in the latest cluster have to be updated. Obviously this is undesirable, and so another scheme is in order. To this end, memcached implementations usually employ variants of consistent hashing to minimize the updates required as the server pool changes and maximize the chance of having the same server for a given key. The basic algorithm of consistent hashing can be outlined as follows:

1. Map the range of a hash function to a circle, with the largest value wrapping around to the smallest value in a clockwise fashion;
2. Assign a value (i.e., a point on the circle) to each server in the pool as its identifier49; and
3. To cache a data item of key k, select the server whose identifier is equal to or larger than H(k).

- The server selected for key k is called k's successor, which is responsible for the arc between k and the identifier of the previous server. As an example, Figure 4 shows a circle of three servers, where server 1 is responsible for caching the associated data items for keys hashed to 6, 7, 0, and 1; server 3 for keys hashed to 2 and 3; and server 5 for keys hashed to 4 and 5. An immediate result of consistent hashing is that a departure or an arrival of a server only affects its immediate neighbors. In other words, when a new server p joins the pool, certain keys that were previously assigned to the original p's successor will now be reassigned to server p, while other servers are not affected. Similarly, when an old server p leaves the pool, the keys previously assigned to it will now be reassigned to p's successor while other servers are not affected. In the example in Figure 6.42, adding a new server 7 would result in reassigning keys 6 and 7 to the new server; removing server 3 would result in reassigning keys 2 and 3 to server 5.

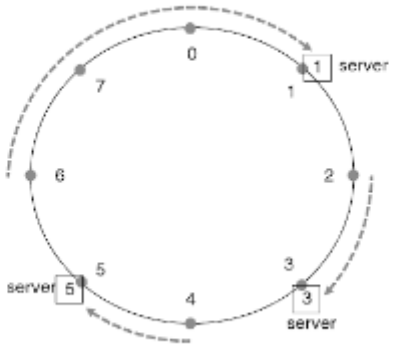


Figure 6.42 A circle in consistent hashing.