Distributed Systems

2015 Exam 3 Review

Paul Krzyzanowski
Rutgers University
Fall 2016

2015 Question 1

What information does each node need to store for a three-dimensional Content Addressable Network (CAN)? Explain how an arbitrary node processes a query for some key, k (just explain what happens on one node).

- Need to store:
  - Addresses of six neighboring node: *left, right, top, bottom, front, back*
  - Range of node in each direction: *xmin, xmax, ymin, ymax, zmin, zmax*
- Query processing

Compute x_hash=hash$_x$(key), y_hash=hash$_y$(key), z_hash=hash$_z$(key)
if (x_hash < xmin) *send to left neighbor*
if (x_hash >= xmax) send to right neighbor
if (y_hash < ymin) *send to bottom neighbor*
if (y_hash >= ymax) send to top neighbor
if (x_hash < xmin) *send to front neighbor*
if (x_hash >= xmax) *send to back neighbor*
else process locally

Key points:
- Six neighbors
- Three hashes

2015 Question 2

Explain the role of dynamic DNS in a content delivery network (CDN).

- It directs the client to a caching server operated by the CDN instead of to the origin server
- This will generally be the closest active server
- DDNS may use load balancing to give addresses if dufferebt servers

Bad answers:

- *Most efficient route (DNS does not dictate routes)*
- *Server that contains the content (DNS doesn't know what the content query will be)*

2015 Question 3

Companies advertise that you should secure your web site with a certificate. Explain how using an X.509 digital certificate at a web server provides security.

- Allows the user to authenticate the web site –
  user validates that the web server has the private key that corresponds to the public key in the certificate
  - Public key is in the certificate
  - User validates the signature on the key (decrypts encrypted hash using CA's public key)
  - User sends a nonce; Server encrypts it with a private key that corresponds to the public key
  - User decrypts the result using the public key in the certificate & compares with the nonce
- Enables exchange of a session key
  - User creates a random session key
  - Encrypts it with the server's public key in the certificate
  - Server decrypts the session key using its private key

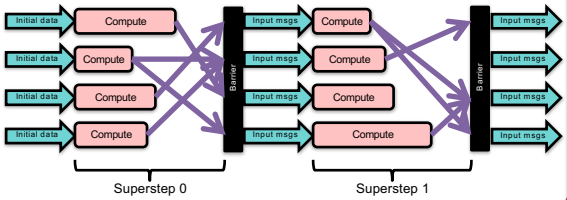
Explain how!

Not:
Cert contains public key

2015 Question 4

A *superstep* is the:

a) Execution of a group of processes between the time they receive inputs to the time they are ready for more input.
b) Execution of several steps on a group of processes until a checkpoint is requested.
c) Ability of a process to skip several steps because it received no messages.
d) Subset of the computation that takes place on a single processor.



2015 Question 5

Pregel addresses fault tolerance by:

a) Replicating the execution of each vertex's compute function on several different servers.
b) Periodically saving all vertex & message state at the end of a superstep.
c) Restarting failed vertices on other computers while the rest of the computation proceeds normally.
d) Storing the results of each superstep into stable storage.

- This is *checkpointing*:
  - Save all state periodically. On failure, restart from last saved state (the last checkpoint)
- It is *not* done at the end of *every* superstep.

2015 Question 6

Under Spark, a Resilient Distributed Dataset (RDD) *cannot* have this property:

a) It can be created by a transformation.
b) It can be partitioned across multiple computers.
c) It can be modified by a task.
d) It can be sorted.

- RDDs are, by definition, immutable
- They are either the original input data or the output of a transformation

2015 Question 7

Spark achieves fault tolerance by:

a) Having each transformation write periodic checkpoints.
b) Storing each RDD on disk as well as in a memory cache.
c) Replicating each RDD onto multiple servers.
d) Keeping track of how each RDD was created.

- RDDs can be recreated by re-running the transformations that created them
- This may require going further back in the chain and re-creating the previous RDD

2015 Question 8

Spanner enables *lock-free* reads by:

a) By waiting out any uncertainty.
b) Sending all read requests through Paxos.
c) Using two-phase locking.
d) Reading versions of data created before a specified time.

- Spanner uses **multiversion concurrency**
  - Spanner stores multiple versions in each field, like Bigtable does
  - A *read* accesses all versions of data < the transaction timestamp
  - Great for long-running reads (such as searches)

2015 Question 9

The TrueTime API in Spanner provides applications with:

a) A globally unique timestamp that may not reflect the actual time.
b) The exact time of day.
c) A time interval that encompasses the current time.
d) The exact local time at the client location while supporting a globally-distributed database.

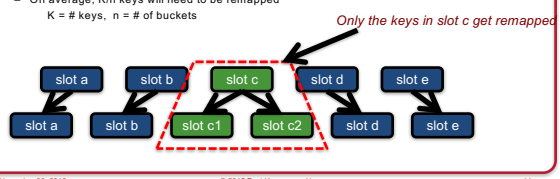
- We cannot get the exact time.
- TrueTime give us the *earliest* and *latest* timestamps
  - TT.now().earliest = time guaranteed to be <= current time
  - TT.now().latest = time guaranteed to be >= current time

2015 Question 10

Differing from conventional hash functions, with a consistent hash:

a) A key repeatedly hashes to the same value for a given table size.
b) Most keys will not hash to new slots when the hash table size changes.
c) The result of the hash is always a fixed number of bits, regardless of the size of the key.
d) It is impossible for two keys to hash to the same value..

Consistent hashing
- Most keys will hash to the same value as before
- On average, K/n keys will need to be remapped
  K = # keys, n = # of buckets



2015 Question 11

In the simplest implementation of Chord with n nodes, where each node knows only its successor, the lookup time is:

a) O(*n*).
b) O(n^2).
c) O(*n* log *n*).
d) O(log *n*).

The search progresses from node to node until a node responsible for the key is found.

2015 Question 12

A *finger table* is a:

a) Hash table of all the keys stored at a node.
b) Table of frequently referenced keys that are located on other nodes.
c) Table of successor nodes.
d) Hash table of frequently referenced nodes.

Finger table = partial list of nodes

At each node, i^{th} entry in finger table identifies node that succeeds it by at least 2^{i-1} in the circle

- finger_table[0]: immediate (1^{st}) successor
- finger_table[1]: successor after that (2^{nd})
- finger_table[2]: 4^{th} successor
- finger_table[3]: 8^{th} successor
- …

2015 Question 13

Dynamo optimizes lookups via:

a) Storing the table of all nodes in the system at each node.
b) Finger tables.
c) Consistent hashing.
d) Storing all key, value sets in a hash table at each node.

- Clients and nodes have the full list of all nodes in the system
- Enables O(1) lookup: no need to forward a query

2015 Question 14

Virtual nodes in Dynamo:

a) Improve geographic scalability since virtual nodes may be spread throughout data centers..
b) Improve performance since each virtual node is responsible only for a small part of the table.
c) Improve fault tolerance since a virtual node is a mirror of a physical node.
d) Simplify load balancing by assigning varying numbers of virtual nodes to physical nodes.

Enable balanced load distribution

- Dead node ⇒ load is dispersed evenly among multiple physical nodes
- New node ⇒ accepts equal load from multiple other nodes
- Newer, faster nodes ⇒ can be assigned more virtual nodes

2015 Question 15

Any component may exhibit a Byzantine or fail-silent failure. To guard against the failure of *n* components, you need:

a) $n + 1$ components.
b) $2n + 1$ components.
c) $3n + 1$ components.
d) $n^2 + n + 1$ components.

- Byzantine failures
  - Need n+1 good components to out-vote n bad components
  - Total components = 2n+1
- Fail-silent failures
  - Need 1 good component to out-vote n bad components
  - Total components n+1
- Byzantine failures are the ones that determine the # of components needed

2015 Question 16

Chubby uses:

a) A passive-passive configuration.
b) Triple modular redundancy.
c) An active-passive configuration.
d) An active-active configuration.

- Only one node (master) takes requests at any time
- Other nodes in the chubby cell get updates

2015 Question 17

System Area Networks (SANs):

a) Provide high-speed, high bandwidth connections among computers.
b) Connect the peripherals within a computer with one network.
c) Enable multiple computers to share common storage.
d) Are similar to a LAN but are designed to span multiple datacenters.

- System area networks (such as Infiniband) are designed to provide low latency switched networking among computers.
- They generally allow communications directly via the system bus (RDMA, remote direct memory access), avoiding the need to go through a network software stack (and deal with checksums, retransmissions, resequencing, flow control, etc.)

2015 Question 18

The difference between a clustered file system and a network file system is that in a clustered file system:

a) Data is replicated among multiple servers for fault tolerance.
b) The operating system uses remote procedure calls to access remote files.
c) File data is distributed across multiple computers for high performance.
d) Multiple operating systems simultaneously access the same file system at the block level.

- A cluster file system is a SINGLE file system that multiple computers may access concurrently
  - The access is at the block level (read block, write block)
  - As with local disks, the file system driver in the operating system is responsible for parsing file names and knowing the structure of the file system (location of inodes, bitmaps of free blocks, block groups, etc.)
  - A distributed lock manager (DLM) is used to coordinate access and ensure two operating systems aren't modifying shared data at the same time.

2015 Question 19

In contrast to a shared-nothing cluster, a *shared-disk* cluster relies on a:

a) Quorum service.
b) Heartbeat network.
c) Cluster membership service.
d) Distributed lock manager (DLM).

- Multiple machines may issue read/write requests for the same block at the same time. A DLM will ensure mutual exclusion.

2015 Question 20

Warm failover is recovery:

a) That does not have critical time limits.
b) From an active server.
c) From a reboot.
d) From a checkpoint.

- Cold failover = application restart
- Hot failover = replica application takes over; replica is always up to date
- Warm failover = restart from some previously saved state (checkpoint)

2015 Question 21

Fencing is used to:

a) Establish a quorum in a cluster.
b) Define the machines that make up a cluster.
c) Create a firewall around a set of computers to restrict specific network packets.
d) Isolate a failed component from the rest of the system.

- Fencing allows you to disconnect a component from the system
- For example:
  - Use a controllable power switch to shut off power to the computer
  - Configure an ethernet switch to drop any network packets to/from that computer
  - Configure a disk array to disallow any connections from that computer

2015 Question 22

In contrast to symmetric cryptography, *public key cryptography*:

a) Solves the problem of transmitting a key securely.
b) Is usually much faster than symmetric cryptography.
c) Is designed for group communication.
d) Is useful for digital signatures, not encryption.

- Symmetric cryptography requires both parties knowing a shared secret key
- This has to be transmitted out of band or encrypted
  - Only way to encrypt is using a trusted third party
  - However, this requires getting the trusted third party to know each user's key

2015 Question 23

For Alice to send a message securely to Bob, she encrypts it with:

a) Her private key.
b) Her public key.
c) Bob's private key.
d) Bob's public key.

- Bob will be the only one who can decrypt since only he has Bob's private key

2015 Question 24

A *hybrid* cryptosystem:

a) Has each communicating party use a unique encryption key.
b) Transmits a session key via public key cryptography.
c) Uses two layers of encryption for stronger security.
d) Adds a cryptographic checksum (hash) to each message.

- A hybrid cryptosystem uses a combination of symmetric & public key cryptography
  - Public key cryptography is used for key exchange (transmitting a randomly-generated symmetric key to the other party)
  - Symmetric cryptography is used for encrypting the communication session once both sides have the key (known as a session key)

2015 Question 25

Salt in a password hash:

a) Guards against dictionary attacks.
b) Encrypts the password in the password file.
c) Guards against using precomputed hashes.
d) Speeds up password checking by storing a hash of the password in the password file.

- Salt is extra random data that is added to the item that is to be hashed
- It changes the resulting hash
- It avoids an attacker using precomputed hashes
  - Suppose "test123" hashes to "SiUaLvm79MzDX5erosnL2g"
  - An attacker can store hashes of common passwords and look up "SiUaLvm79MzDX5erosnL2g"
  - However, if we suffix random data to "test123", such as "test123$TyuB", we get VIk4MLxqVkhXIFRDoCzTjA
  - An attacker can no longer use a table of precomputed hashes of common passwords

2015 Question 26

CHAP, the Challenge Handshake Authentication Protocol:

a) Is vulnerable to replay attacks.
b) Transmits a password in plain text (unencrypted)
c) Is vulnerable to man-in-the-middle attacks.
d) Is based on public key cryptography.

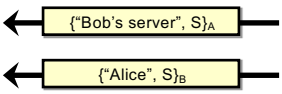
- An intruder in the middle can forward messages between the two parties until authentication is complete

2015 Question 27

Kerberos gives you two items. One of them is a sealed envelope, or ticket. This contains:

a) A session key that you can decrypt for communicating with the service.
b) A session key that the remote service can decrypt but you cannot.
c) A timestamp to guard against replay attacks.
d) The public key of the remote service.

- If Alice requests a session with Bob, Kerberos sends her:
  1. A session key encrypted with her secret key
  2. A session key encrypted with Bob's secret key ⇒ ticket (sealed envelope)



2015 Question 28

SSL, the Secure Sockets Layer, uses a:

a) Symmetric key cryptosystem.
b) Public key cryptosystem.
c) Hybrid cryptosystem.
d) Restricted cipher.

- Hybrid cryptosystem:
  - Public key cryptography for session key exchange
  - Symmetric cryptography for communication

2015 Question 29

OpenID Connect:

a) Enables a third party service to authenticate a user using a protocol of its choosing.
b) Uses public key cryptography to authenticate a user and establish a secure connection.
c) Uses a combination of publicly-readable user IDs and secret passwords to authenticate users.
d) Is designed for services rather than users to identify themselves to each other when they connect.

- OpenID Connect does not define an authentication protocol – it simply delegates that to another service.

2015 Question 30

An IP tunnel:

a) Encapsulates entire IP packets as data when communicating between two networks.
b) Is a point-to-point TCP connection for transmitting data between two nodes.
c) Is a connection where all data between two nodes is encrypted.
d) Enables two computers to communicate without the use of routers.

2015 Question 31

An overlay network is a:

a) Set of connections that define a spanning tree to ensure there are no cycles.
b) Private network of high-speed connections that overlays part of the public Internet.
c) Wireless network that overlays the wired Internet.
d) A logical set of links between nodes that is built on top of a physical network.

The End

