15-440/15-640: Homework 4

Due: December 6, 2017 11:59 pm

Name:
Andrew ID:

1 Spark (20 points)

You have been hired as a distributed systems engineer at a superfamous institution called "AIT". AIT is currently researching on a new and faster PageRank algorithm which performs iterative computation on the data. The datasets used for simulations and testing are in the order of Terabytes. To process this data, the team has decided to use MapReduce.

From your experience at 15-440/640, you don't like the idea and instead propose using Spark for the processing.

1. Why using MapReduce is not a great idea here? **[5 points]**

   MapReduce reads from and writes to disk/ssd at every iteration. This involves a lot of I/O which can lead to bad performance. As a result, MapReduce is not efficient for iterative processing.

2. What are the benefits of using Spark in this scenario? Mention atleast two. **[5 points]**

   Two benefits of using Spark here:
   i. Spark maintains lineage of operations (DAG) applied to the dataset and uses lazy evaluation to process. As a result, multiple operations can be pipelined together which improves performance.
   ii. Spark uses in-memory data abstraction which allows working datasets to be in memory. This reduces redundant I/Os (unlike in MapReduce).

3. Your boss reads through spark definition and suddenly freaks out after seeing "in-memory computation". He claims that the processed results can be lost and the framework has high failure probability. What do you say to calm your boss? **[5 points]**

   Spark keeps the lineage of all operations applied on the dataset in the stable storage. In case of a failure of a node (or power failure), it can recompute the lost data partition using the lineage it maintained.

4. What are the performance challenges of using Spark in distributed machine learning? **[5 points]**

   There can be many challenges. Some of them are: 1. Some machine learning algorithms employ fine-grained updates which is not suitable for Spark. 2. Machine learning often requires lot of data to train which might not fit in memory.

2 Blockchains (25 points)

Blockchains are used to implement distributed ledgers, which, for example contain monetary transactions like Daniel sends 0.1 Bitcoins to Devdepp. There are two important implementation problems: reaching consensus on which transactions are added to the ledger and preventing the manipulation of the ledger to change transactions that happened in the past.

For efficiency, consensus is decided on multiple transactions at once (a block). A blockchain comprises of a sequence of blocks (a chain), each block containing thousands of transactions. Like in

the Paxos consensus protocol, every member of the blockchain P2P network could potentially make a proposition for which transactions go into the next block (as all transactions are broadcast). However, there are millions of members, and some of them might have malicious (Byzantine) intentions.

1. State two reasons why popular blockchains like Bitcoin do not use Paxos to solve the blockchain consensus problem. [**5 points**]

   Scalability: Paxos cannot scale to millions of nodes. There would be a lot of communication needed, etc.
   Byzantine failures: Byzantine versions of Paxos are even less scalable.
   (Variant): Paxos allows one vote per IP address, giving people to make malicious majority votes if the only can rent a sufficient number of IP addresses.

2. In practice, the consensus problem is solved using a technique called proof of work. For a block to be considered valid, it must be accompanied by a proof of some (computational) work that the proposer of the block performed. This is typically the solution of a mathematical puzzle based on the data in the block, such that if the data in the block is changed, the solution of the puzzle also changes. What properties must this puzzle have? Hint: If too many peers solve the problem, it leads to the same issue of how to decide which block to accept. [**5 points**]

   1. Hard to compute
   2. Easy to verify

3. What method is used to make the modification of an entry in an already accepted block much harder than solving the puzzle for just one block? [**5 points**]

   Each block has a hash pointer to the previous block. Hence, if one block is changed, every subsequent block must also be changed, and the puzzle for each of these blocks needs to be solved.

4. BitCoin uses a proof-of-work mechanism called hashcash, which has a difficulty parameter D. If solving hashcash with difficulty D=k requires time T on average, how much time will it take to solve hashcash with D=k+3? ? [**5 points**]

   8*T

5. What is the maximum number of bitcoins that can ever be in circulation? [**5 points**]

   Let a = 21,000,000 (= 100 * 210,000).
   Sum of series 0.5a + 0.25a + 0.125a + ... = a.
   Hence, the maximum number of bitcoins that can exist are limited to 21,000,000.

3 Virtualization (20 points)

1. Describe the two main differences between Type I (Hypervisor) and Type II (Hosted) VMMs. [**5 points**]

   Type I VMMs run directly on the system hardware and are also known as bare-metal hypervisors. They provide higher performance, availability, and security than Type II VMMs. eg: Citrix XenServer.
   Type II VMMs run on a host operating system such as windows, Linux etc. It uses a guest operating system which runs on top of the hypervisor. eg: VMWare workstation.

2. What is the main problem with the Trap and Emulate way of virtualizing the x86 architecture? What are some of the approaches to solve the problem? **[5 points]**

x86 is not classically virtualizable and has 17+ untrappable but unprivileged instructions. Ex: POPF. When these instructions are executed in the guest OS, they fail silently. Para-virtualization, Full virtualization, or hardware support are some of the approaches to solve this problem.

3. In virtualization, can the hypervisor (or VMM) allocate and assign more than the actual physical resources it has available at its disposal (memory, processors) to individual's Virtual Machines (VMs)? Please explain. **[5 points]**

YES. Virtualization allows you to overcommit resources and provide more virtual re- sources than actual physical resources, for example by time multiplexing the same set of CPU/processors across multiple VMs or using memory ballooning to use less or more memory. (either CPU or memory example is fine)

4. You are a data center engineer. Explain two (2) advantages and two (2) disadvantages of using Containers (Docker) over Virtual Machines in the data center to your manager? Which one would you choose in a multi-tenant environment and why? **[5 points]**

Advantages: 1. Fast boot-times 2. Light-weight **Disadvantages**: 1. No strong isolation guarantees. 2. Limited Security. Virtual machines provide strong isolation guarantees between different tenants that is not achievable with the current container techonology.

4 Byzantine Fault Tolerance (15 points)

All the students (and faculty) in 15-440/640 are chosen as captains for the next Star-Trek mission. Each member will lead a Federation spaceship. Your mission is to attack and destroy the stronghold of a villain named "The Borg".

However, Borg, using his evil powers have brainwashed some of you to betray the Federation. Suspecting this, "Captain Yuvraj" has decided to form practical Byzantine fault tolerant system for the mission. Captain Yuvraj(aka "the client") sends an attack signal to all the ships. Also, the Federation has appointed "Captain Srini" to be the organizer who then directs the other ships as per the BFT protocol (Assume that Captain Srini properly forwards the request.)

Given, Number of students + Number of Faculty = 150. To determine whether to attack, Srini must count the number of photon torpedoes among the ships. If we have more than 1000 torpedoes, we should attack.

1. If all of the 150 members are participating, what is the maximum number of people who will be brainwashed by The Borg before we can't gurantee BFT consensus? **[4 points]**

For it to work: 150 = 3f + 1. Solving for f, we get f = 49 (rounding off to lowest integer)

2. Captain Yuvraj is watching the replies of ships. After how many matching committed replies, will he know that any further non-brainwashed member will reply the same? **[4 points]**

For it to work at all: 150 = 3f +1, solving for f, we get f = 49. For a final answer: f+1 = 50 Replies.

3. On the way to destroy The Borg, you encounter the "Asteriod belt" and 147 ships are destroyed. You decide that even 3 ships(i.e A, B, C) are able to destroy the enemy if all of them attack simultaneously. Unfortunately, Captain Sandeep who is leading ship B is brainwashed and remaining captains are honest. Describe a scenario where the honest captains cannot decide to attack even though they have 1000 torpedoes just among the honest captains. **[7 points]**

Here, you have to find a situation where A and C don't attack together. Assuming B initiates the command and sends following signal to A and C: B tells A: "Attack", B tells C: "Withdraw". In this scenario, A would go ahead and attack while B and C won't. This is an example of byzantine failure.

5 Security (20 points)

Srini and Yuvraj need to communicate to decide which TA is going to grade the next homework. They have a shared secret key, K_{Profs} that allows them to create unforgettable message authentication codes (MAC) so that Srini can verify that Yuvraj did in fact create any message that is received.

Srini and Yuvraj have a simple protocol: Srini sends a "*Who grades HWX?*" message to Yuvraj in plain text, and Yuvraj replies with one of two messages: $M1 = MAC_{K_{Profs}}$("Sandeep"), or $M2 = MAC_{K_{Profs}}$("Vamshi"). When Srini receives either M1 or M2, he verifies the MAC using K_{Profs} and knows who will grade the next homework.

1. This protocol is insecure. A malicious TA on a router between Srini and Yuvraj might be able to avoid ever having to grade a homework! In one sentence, describe the attack. **[2 points]**

It is subject to a replay attack. The TA could replay an earlier answer for a different TA's name.

2. What simple change to the above protocol could defend against this attack? **[4 points]**

Use a nonce. Along with the "Who grades HWX" message, Srini should also send a random string that must be included in the MAC to ensure that the answer is unique.

Srini and Yuvraj wants to discuss the final exam questions through email. They decided to encrypt their emails to avoid being attacked by genius students like you. The first thing they have to do is to agree on a secret key. A TA suggests that they can use Diffie-Hellman key exchange protocol.

1. Suppose they have agreed on g = 5 and p = 23 (g and p are public). Now Yuraj picks his secret number 6 and Srini picks his secret number 8. What should Yuraj sends to Srini? What should Srini sends to Yuraj? And what is secret key they agree on? **[6 points]**

Either of g = 5 and p = 23, or g = 23 and p = 5 is awarded full points. Using g = 5 and p = 23 Yuraj sends to Srini: $g^a mod p = 5^6 mod 23 = 8$. Srini sends to Yuraj: $g^b mod p = 5^8 mod 23 = 16$. The secret key they agree on: $8^8 mod 23 (16^6 mod 23) = 4$ **(OR)** Using g = 23 and p = 5 Yuraj sends to Srini: $g^a mod p = 23^6 mod 5 = 4$. Srini sends to Yuraj: $g^b mod p = 23^8 mod 5 = 1$. The secret key they agree on: $4^6 mod 5 (1^6 mod 5) = 1$

2. Assuming they use cryptographically secure primes, why can the final exam questions still be stolen by an attacker? Give an example of an attack using the parameters from part 1. **[8 points]**

Use Man-In-The-Middle (MITM) attack. Suppose I can intercept the network traffic.
Using g = 5 and p = 23
When I see $A = g^a mod p = 8$ from Yuraj to Srini, instead of let it go through, I can choose my secretnumber c=7 and send $g^{ac} mod p = A^c mod p = 8^7 mod 23 = 12$.

Similarly, when I see $B = g^b mod p = 16$ from Srini to Yuraj, I will replace it with $g^{bc} mod p = B^c mod p = 16^7 mod 23 = 18$.

Now I agree with both of them on a secret key. I can decrypt whatever they send and they still think they are talking to each other!
(OR)

Use Man-In-The-Middle (MITM) attack. Suppose I can intercept the network traffic.
Using g = 23 and p = 5
When I see $A = g^a mod p = 4$ from Yuraj to Srini, instead of let it go through, I can choose my secretnumber c=7 and send $g^{ac} mod p = A^c mod p = 4^7 mod 5 = 4$.

Similarly, when I see $B = g^b mod p = 1$ from Srini to Yuraj, I will replace it with $g^{bc} mod p = B^c mod p = 1^7 mod 5 = 1$.

Now I agree with both of them on a secret key. I can decrypt whatever they send and they still think they are talking to each other!