PES University, Bengaluru
(Established under Karnataka Act No. 16 of 2013)

UE16CS352

MAY 2019: END SEMESTER ASSESSMENT B.TECH. 6 SEMESTER

FINAL EXAM

UE16CS352 - CLOUD COMPUTING

Time: 2 Hrs   Answer All Questions   Max Marks: 60

Instructions:
- For problems, show the working. Providing just the answer is not acceptable. Clearly state your assumptions.
- One memory recall sheet (A4) is permitted in your own handwriting.

1		Answer in 2-3 lines.	5x3=15
	a)	In the Siddhartha Reddy video, he talks about 6 different causes of Partitions. One of them is Garbage Collection. How can Garbage collection lead to partitions?	3
	b)	In the Rob May video on the "Human Firewall", he talks about a survey conducted in liverpool station and makes the statement "with Cybersecurity it is a numbers game". What does he mean by that in the context of the survey and why is the survey a risk?	3
	c)	What does the cpuset cgroup allow you to do?	3
	d)	As compared to HTTP, how does reliable messaging solve the problems of (i) recipient crashing (ii) recipient changing (iii) recovery from recipient crashing	3
	e)	What is the difference between Paravirtualization and Transparent virtualization? Give an example of each.	3

2

a)	Consider a chat system as shown above. P1-P6 are the chat nodes and C1 and C2 are example users. Each user is connected to a chat node as shown. The chat nodes receive messages from the users and process them. One of the nodes is a *coordinator node*. (1) What are the types of node failures possible in the system? (2) How would you recover from these?	2+4
b)	Draw an example of a client-server chat system. Is the diagram above a peer-to-peer chat system? Name one chat function that would be simpler in a client-server chat system than a peer-to-peer chat system.	2+2+2
c)	List and explain 3 security measures that can be used to protect cloud storage.	3

3	a)	You have initially built a IaaS based solution *T20PremierLeague* for management of T20 cricket tournaments. The *T20PremierLeague* is a web application that is deployed on a single Linux VM on AWS. As more requests start coming in, you need to add additional VMs to cater to the requests. Draw an architectural diagram of your scalable solution demonstrating the connection between the client and the VMs and any new component(s) that need to be added to take care of multiple VMs?	3
	b)	To implement a scalability solution for *T20PremierLeague,* you added the following rule to increase the number of VMs with increasing number of client requests, but you observe that even though you have increased the number of VMs, the utilization of the VMs is not high. Why is this? Suggest a modification to the rule to utilize the VMs better?	3

		`if (#requests in last 1 min > n*1000)   [trigger] where n is the current #VMs allocated`
		`    increase #small size VMs by 1 ( n ← n+1) [action]`
		`    wait for 5 mins for your change to take effect`	

	c)	As the designer of the system, the users complain that even though you have implemented the scalability solution by adding a VM when the number of requests increase, the system	3

		does not respond quickly enough and latency of accessing data is high. What could be the reason for this delay in responding to the scale-out action? Can you suggest an alternative to the design in (a) that could solve this issue.	
	d)	The *T20PremierLeague* application consists of both code and data with virtual address ranges as given below	3

	Lower Address Bound	Higher Address Bound
Code	0x30050000	0x30052000
Data	0x40023100	0x40076420
Stack	0x30060000	0x30080000
Linux VM (Guest)	0x10000000	0x10040000

Consider the following address - 0x30050004, 0x40023101c and 0x10030000
Which ones of the above are Guest Virtual Addresses and which ones are Host Virtual Addresses?

	e)	For the problem in 3(d), suppose the *T20PremierLeague* application accesses the guest virtual address 0x3005011c on a machine with 4KB page sizes. The guest physical page and host physical pages associated with this address are 0x7 and 0x20. Show the corresponding entries for this address in both the guest page table and the shadow page table.	3

4	a)	You build a PaaS based solution *T20PremierLeague* for management of T20 cricket tournaments. The t20pl is a web application that supports the following APIs which contain the following APIs -- *CreateFranchise, GetFranchisePromoter, GetMatchScore, AddPlayer* with a simple web based GUI. The solution also contains a module to predict the score at any point of time in the match using the *GetScorePrediction()* API. Draw an architecture diagram of the web application as it will be deployed on a PaaS platform. Highlight in the diagram, which APIs require processing in the backend (WorkerRole) and what requires frontEnd processing (WebRole). How will the web and worker roles communicate with each other?	3
	b)	One of the challenges identified by the team was to keep the data shared between varios worker roles as the number of instances of the WorkerRole increases. Suggest one solution to share data amongst the various WorkerRole instances?	3
	c)	With more customers wanting to use this software, you decide to move your solution to a *SaaS* based solution. What additional design changes do you need to make both in terms of API and GUI? List additional APIs and types of GUI screens to be added for making *T20PremierLeague* SaaS based solution?	3
	d)	Two leagues- *CollegeLeague* and *BollywoodPremierLeague* want to deploy your solution, but *CollegeLeague* wants to associate the college of every college with player while *BollywoodPremierLeague* wants to add the latest released movie with every player so that they can promote the movie and the total they are willing to pay the league to promote their movie. Suppose the database is to be shared by two customers - *CollegeLeague* and *BollywoodPremierLeague*. Explain (i) how this can be done using the pre-allocated column method (ii) how *BollywoodPremierLeague* can write a query to get the total amount collected from players of each team for promoting their movies and (iii) demonstrate how this query will be translated by theSaaS system into the final query on the shared database	2*3=6