

Student ID _____ Student Name _____

Software Architecture

Final Exam May 2019

PRIVATE AND CONFIDENTIAL

1. Allotted exam duration is 2 hours.
2. Closed book/notes.
3. No personal items including electronic devices (cell phones, computers, calculators, PDAs).
4. Cell phones must be turned in to your proctor before beginning exam.
5. No additional papers are allowed. Sufficient blank paper is included in the exam packet.
6. Exams are copyrighted and may not be copied or transferred.
7. Restroom and other personal breaks are not permitted.
8. Total exam including questions and scratch paper must be returned to the proctor.

4 blank pages are provided for writing the solutions and/or scratch paper. All 4 pages must be handed in with the exam

BE VERY CAREFUL WITH THE GIVEN 2 HOURS AND USE YOUR TIME WISELY. THE ALLOTTED TIME IS GIVEN FOR EVERY QUESTION.

Write your name and student id at the top of this page.

Write your answers clearly. If I cannot read it, you don't get points for it.

Question 1 [10 points] {15 minutes}

Explain clearly how we make our microservices secure in a microservice architecture

Your answer:

We use token based security using OAuth2.

The authentication service knows the username-password-role combination.

A service request a token based on username-password for a certain user. With this token, a service can retrieve the role for this user. If one service calls another service, it sends this token so that the second service can retrieve the role for this user.

We can use JWT tokens so that tokens are signed by the authentication service

Question 2 [20 points] {20 minutes}

- a. Suppose you have to implement a microservice architecture of a webshop. In this webshop, the order service calls the product service regularly. The only problem is that if the product service is down, the order service does not work properly. Explain clearly how we can solve this problem

Your answer:

We can run multiple instances of the product service. The order service will use client side load balancing (Ribbon).

- b. Suppose you have to implement a microservice architecture of a webshop. Clients call the services through the API gateway. The only problem is that if the API gateway is down, clients cannot shop anymore. Explain clearly how we can solve this problem

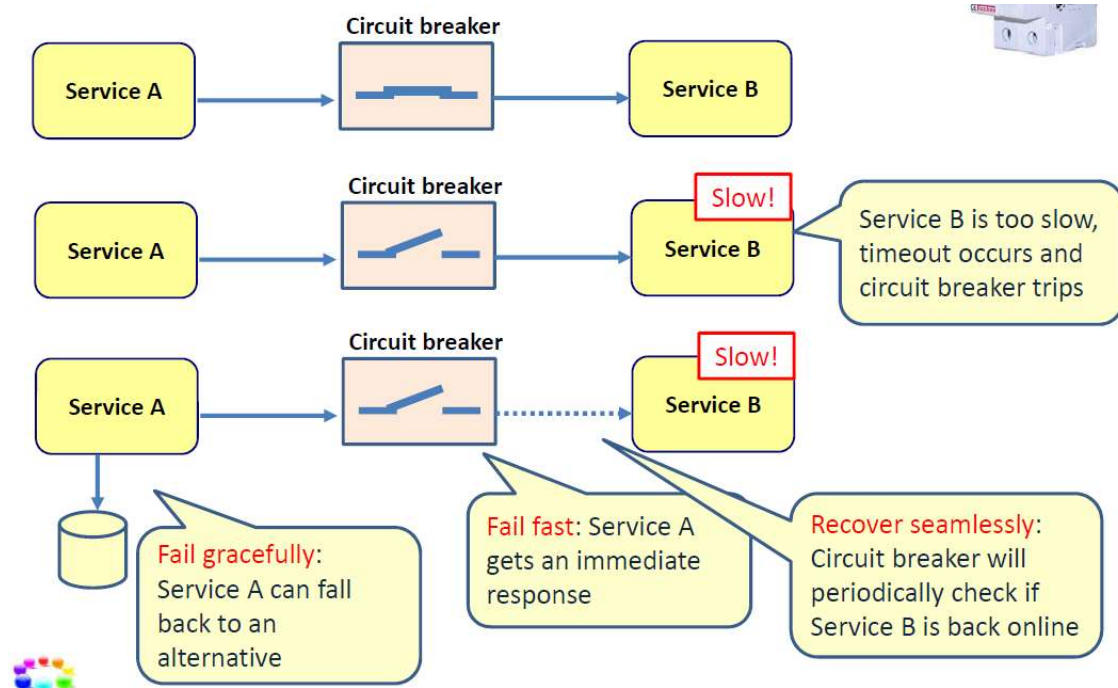
Your answer:

We run multiple instances of the API gateway, and place a load balancer in front of it. Note that Ribbon cannot be used here. Ribbon is a client side load balancer, not a server site load balancer

Question 3 [10 points] {10 minutes}

Explain clearly how hystrix solves the problem of slow microservices. Explain how this works.

Your answer:



Question 4 [10 points] {10 minutes}

Fill in the following table. In the first column, give the problems we face when we implement a distributed system. In the second column, give the technique/pattern that we learned that helps us to manage this problem. Give all problems and solutions that you know.

Problem we face in a distributed system	Solution that helps to manage this problem
Complex communication	API gateway Registry EDA
Security	OAuth2, JWT
Transactions	SAGA pattern Compensating transactions Eventual consistency
Resilience	Hystrix
Monitoring	Zipkin, sleuth, ELK
Configuration	Config service

Question 5 [10 points] {10 minutes}

For which of the following architecture requirements would you use CQRS?
Circle all architectural requirements so that if this architectural requirements applies, you would apply the CQRS pattern.

- a. When you need future proof systems
- b. When you build a distributed system
- c. When read performance is critical and need to be extremely fast for your microservice
- d. When you design your microservice using Domain Driven Design
- e. When queries and commands have different scaling requirements for your microservice
- f. When your screens start to look very different then your tables in your microservice
- g. When you have an event driven architecture
- h. When you need load balancing for your microservice
- i. When you use a NoSQL database for your microservice
- j. When you apply event sourcing to your microservice
- k. When strict consistency is required for your microservice
- l. When security is important for your microservice

Question 6 [20 points] {25 minutes}

Suppose we design a microservice architecture for a book webshop. If we navigate on the client to a certain book, then a lot of data is shown that is retrieved from different microservices:

- Book data is retrieved from the book microservice
- Author data is retrieved from the author microservice
- How many books are available is retrieved from the stock microservice
- Books from the same genre is retrieved from the book microservice
- Info of people who bought this book are also interested in these books is retrieved from the order microservice
- Price of this book from other bookshops is retrieved from the competitor pricing microservice

Now we have the problem that when a client application want to retrieve the data for a book, it takes too long to retrieve all this data from all these different microservices. One requirement of this application is that it is very important that all this book data is shown completely (not partially) and at once (no delays). Explain clearly how you solve this problem. Make it very clear which technique(s) or pattern(s) you use to solve this problem. The number of points you get for this question depends on how well your solution works, and how well you explained it.

Answer: Use CQRS. The query part of the Book service contains all necessary data for a book. We optimize it for fast reads. When certain data changes in the other services, it will publish this change as an event. The Books query service subscribes to these events, and updates the necessary data.

Question 7 [15 points] {20 minutes}

Explain the difference between orchestration and choreography.

Your answer:

Orchestration: one central brain that orchestrates the process over services

Choreography: no central brain, every service knows what to do in their own context

Give the advantages and disadvantages of both

Orchestration	Choreography
Advantages Process is easy to monitor Works well in a small and/or simple environment	Advantages Works well in a large and/or complex environment
Disadvantages Does not work well in a large and/or complex environment Single point of failure	Disadvantages Process is hard to monitor

Question 8 [5 points] {10 minutes}

Describe how we can relate the **API Gateway** to one or more principles of SCI. Your answer should be about half a page, but should not exceed one page (handwritten). The number of points you get for this questions depends how well you explain the relationship between the **API Gateway** to one or more principles of SCI.

Your answer: