

Assignment 1

Proposed Solution(s)

- 1 Example: Do quick research online and write your additional assumptions based on the challenge outlined in this part.
Example: No expertise on serverless development.
- Example: You should write a proposed solution(s) here. Need not be perfect but challenge yourself to understand the problem stated and read widely what are the solutions documented in the internet.

Challenge #	Describe the challenge with assumptions		Proposed Solution(s)	Additional considerations/Recommendation
Challenge 1 - Compatibility Issue Scenario: Your superior wants to leverage on serverless technology as it eliminates operational overheads but the software is not serverless ready.	Assumptions: The challenge with using serverless technology for non-serverless software is that serverless applications are typically designed to be stateless and event-driven. This means that they do not maintain any state between requests and are triggered by events, such as HTTP requests, database changes, or messages on a queue. Non-serverless software, on the other hand, may be stateful or have other requirements that make it difficult to port to a	Challenge: In the scenario, it is assumed that my superior is aware of the benefits of serverless technology and has decided that it is the right approach for your software. However, it is also assumed that your superior is not aware of the challenges involved in porting non-serverless software to a serverless architecture.	Proposed Solution(s) There are a few different solutions to the challenge of using serverless technology for non-serverless software: 1)Refactor the software: One approach is to refactor the software to make it more stateless and event-driven. This may involve breaking the software down into smaller, more focused components, and using external services to manage state and perform long-running tasks. 2)Use serverless wrappers: Another approach is to use serverless wrappers. Serverless wrappers are third-party tools that can	recommended that you start by having a conversation with your superior as well as team to discuss their expectations and goals for using serverless technology. Once you have a better understanding of their requirements, you can begin to assess the feasibility of porting your software to a serverless architecture. Eg: Cost Performance and Vendor lock in If you decide that it is feasible to port your software to serverless, I recommend that you start by refactoring the software to make it more stateless and event-driven. This will make it easier to deploy and manage your application on a serverless platform. If you are unable to refactor your software, you may want to consider using serverless wrappers or serverless hybrids. These approaches can allow

serverless architecture. For example, non-serverless software may need to maintain a connection to a database or other external service, or it may need to perform long-running tasks.

be used to package non-serverless code into a serverless-compatible format. This allows developers to run their existing code on a serverless platform without having to refactor it.

3)Use serverless hybrids: Serverless hybrids are applications that combine serverless and non-serverless components. This approach can be used to leverage the benefits of serverless computing for certain parts of an application, while still maintaining the flexibility and control of a non-serverless architecture for other parts of the application.

The best solution for your particular application will depend on a number of factors, such as the type of software, its requirements, and the developer's experience with serverless computing.

you to run your existing code on a serverless platform without having to make significant changes to the code itself.

Challenge 2: Vendor Lock-in	<p>Scenario: You have picked Cloud X as the destination of your migration only to find out that you'll better with Cloud Y as your technology scale.</p>	<p>You have already migrated your applications and data to Cloud X.</p>	<p>It can be expensive and time-consuming to migrate applications and data from one cloud platform to another.</p>	<p>If you find yourself in this situation, there are a few things you can do:</p>	
		<p>You have realized that Cloud Y is a better fit for your needs, especially as your technology scales.</p>	<p>You may have to make changes to your applications and infrastructure in order to make them compatible with Cloud Y.</p>		
		<p>You are concerned about the cost and complexity of migrating to Cloud Y.</p>	<p>You may experience downtime or disruption to your business during the migration process.</p>		
			<p>There is a risk of data loss or corruption during the migration process.</p> <p>Vendor lock-in: If you have signed a long-term contract with Cloud X, it may be difficult and expensive to terminate the contract early.</p>		

1. **Negotiate with Cloud X:** Talk to your account manager at Cloud X and explain your situation. See if they are willing to work with you to improve your experience or make it easier for you to switch to Cloud Y.

2. **Use a cloud migration tool:** There are a number of cloud migration tools available that can help you to move your applications and data from one cloud platform to another. These tools can make the migration process easier and less disruptive.

3. **Use a multi-cloud strategy:** A multi-cloud strategy is an

		<p>Data sovereignty: If you have data that is subject to data sovereignty regulations, you may not be able to migrate it to Cloud Y if Cloud Y does not have a data center in the region where the data is required to be stored.</p> <p>Security & compliance: You will need to ensure that your applications and data are secure & compliant with all applicable regulations on Cloud Y.</p>	<p>approach to cloud computing in which an organization uses multiple cloud providers. This can help to reduce vendor lock-in and improve resilience.</p> <p>If you are considering migrating to Cloud Y, it is important to carefully evaluate the platform to make sure that it meets your needs. You should also consider the cost of migration, the complexity of your application, and the impact on your business.</p>	
--	--	--	--	--

Challenge 3: Processes & Policies

Scenario:
Locally (in data center) configured security policies is not identical to the policies architecture in the Cloud. The deployment process of the current IT Operations Team may not fit well with the Cloud's.

Your organization currently has a set of security policies in place for its on-premises data center environment.

You are planning to migrate to the cloud, but you are concerned about the following:

The cloud's security policies may not be identical to your organization's current policies.

Your organization's current IT operations team may not be familiar with the cloud deployment process.

You want to minimize the challenges of migrating to the cloud and ensure a successful migration.

Security: If your organization's cloud security policies are not identical to its on-premises security policies, there is a risk that your data and applications may be less secure in the cloud.

Deployment: If your organization's IT operations team is not familiar with the cloud deployment process, there is a risk that the migration process may be delayed or unsuccessful.

Security:

Review your organization's cloud security policies and make sure that they are aligned with your organization's overall security strategy. You can use the cloud security policies provided by your cloud provider as a starting point, but you may need to make some changes to ensure that the policies meet your specific needs.

Use a cloud security posture management (CSPM) tool to help you monitor and manage your cloud security posture. A CSPM tool can help you to identify and mitigate security risks in your cloud environment.

Deployment:

Work with a cloud migration partner. A cloud migration partner can help you to develop a migration plan, execute the migration, and minimize downtime and disruption to your business.

Train your IT operations team on the cloud deployment process. This will

Start by developing a cloud migration strategy. This strategy should define your goals for the migration, as well as the applications and data that you need to migrate.

Choose a cloud provider that meets your organization's needs. Consider factors such as security, compliance, performance, and pricing when making your decision.

Develop a migration plan. This plan should include a timeline for the migration, as well as a description of the steps that need to be taken to migrate each application and data set.

Test your migrated applications and data before rolling them out to production. This will help you to identify and fix any problems before they impact your users.

			help them to understand the steps involved in migrating and deploying applications to the cloud.	
--	--	--	--	--

Submission