

Introduction to SWE II

1 Kubernetes

1. Explain the "Works on My Machine" problem. Using the details from the presentation, describe how different software versions and operating systems can cause an application to fail in various environments and why this leads to deployment issues.
2. Describe the fundamental difference between how Virtual Machines (VMs) and Containers achieve virtualization. In your answer, compare their performance characteristics, including speed, weight, and resource efficiency.
3. Define what a Dockerfile, an Image, and a Container are. Explain the relationship between them, describing how one leads to the next in the process of running an application.
4. The presentation explains that once you have a container, you face the problem of scale. Describe the four specific challenges mentioned on slide 8 that arise when you need to manage many containers, and briefly explain what is meant by Scaling, Self-Healing, Rolling Updates, and Service Discovery.
5. Explain the core idea of "Desired State Management" in Kubernetes, as detailed on slide 10. How does a user declare the desired state, and what is Kubernetes's continuous job in response? Use the example of a crashing container to illustrate how Kubernetes maintains this state.
6. The Kubernetes architecture is composed of a Control Plane ("The Brain") and Worker Nodes ("The Muscle"). Describe the primary role of each of these components. List and explain the function of two components from the Control Plane and two components from a Worker Node as detailed on slide 12.
7. According to slide 15, what is a Kubernetes Namespace used for? Explain at least two benefits of using namespaces in a multi-team or multi-environment cluster.
8. Define what a Kubernetes Pod is and why it's considered the "smallest deployable unit". As mentioned on slide 17, what key resources do containers within the same pod share, and what does Kubernetes do if a Pod dies?
9. Explain the function of a Kubernetes Deployment. What essential information does a Deployment's blueprint define, and what key capabilities does it provide for managing application updates and ensuring reliability?
10. Pods in Kubernetes are not permanent and their IP addresses can change. Explain why this creates a problem and how the Kubernetes Service object solves it. Describe the primary function of a Service and what it provides to users and other applications.
11. According to slide 23, why is it beneficial to store configuration data separately from application code? Explain the difference between a ConfigMap and a Secret, and provide an example of the type of data that would be stored in each.
12. Slide 25 presents a structural hierarchy it calls "The Container Chain". List this chain of objects in order, from the largest element (Cluster) down to the smallest (Container). Briefly describe how each object in the chain logically contains or manages the next.
13. What is the difference between Kubernetes and Docker and how are they used together?

What to Submit

Your answers.

Contributors

- Anika Sikka