

Docker Senior-Level Interview Questions and Answers

Core Docker Concepts

Q: What is Docker and how does it differ from a virtual machine?

A: Docker is a containerization platform that packages applications and their dependencies into containers. Unlike VMs, containers share the host OS kernel and are lightweight and faster.

Q: What is the difference between an image and a container?

A: An image is a static file with application code and dependencies; a container is a running instance of an image.

Q: How does Docker use layered filesystems?

A: Docker images are built in layers; each command in a Dockerfile adds a layer. This promotes caching and efficiency.

Q: Explain the Dockerfile and its common instructions.

A: Instructions include FROM, RUN, COPY, ENTRYPOINT, CMD, etc., to build images step-by-step.

Production-Grade Questions

Q: How do you troubleshoot a container that keeps crashing?

A: Use 'docker logs <container>' to check logs, 'docker inspect' for details, and check exit codes.

Q: How do you reduce the size of Docker images?

A: Use multi-stage builds, minimal base images, and clean up unnecessary files in the image.

Q: How do you scan Docker images for vulnerabilities?

A: Use tools like Trivy, Clair, or Docker Scan to identify security issues.

Q: Difference between volume, bind mount, and tmpfs?

A: Volumes are managed by Docker, bind mounts map host paths, tmpfs stores data in memory only.

Networking and Compose

Docker Senior-Level Interview Questions and Answers

Q: How does Docker networking work?

A: Docker creates virtual networks allowing containers to communicate securely.

Q: Difference between bridge, host, and overlay networks?

A: Bridge is default isolated network, host shares host network, overlay is for multi-host communication.

Q: How do you scale services in Docker Compose?

A: Use 'docker-compose up --scale service=num'.

Q: How to override configs for staging/prod?

A: Use docker-compose.override.yml files and environment variables.

Security and Real-Time Scenarios

Q: How do you secure a Docker container?

A: Run as non-root, limit capabilities, use seccomp/apparmor profiles, and scan images.

Q: What is Docker Content Trust?

A: It ensures image signing and verification for security.

Q: A container restarts in loop with exit code 137. What does that mean?

A: Exit code 137 means the container was killed (likely due to out-of-memory or kill -9).

Q: How do you handle port conflicts between two containers?

A: Use different host ports or Docker networks to avoid direct conflicts.

Optimization and Miscellaneous

Q: How to isolate container resources?

A: Use resource limits: 'docker run --memory 512m --cpus 0.5 ...'.

Q: What is an ENTRYPOINT vs CMD?

A: ENTRYPOINT defines the executable, CMD provides default arguments. ENTRYPOINT is not overridden

Docker Senior-Level Interview Questions and Answers

by docker run args.

Q: Difference between COPY and ADD in Dockerfile?

A: COPY is for files/folders, ADD also supports remote URLs and unpacking archives.

Q: How to pass environment variables to a container?

A: Use '-e VAR=value' with docker run or define them in docker-compose.yml.