Docker intro

25 April 2022 10:45

- Docker overview
  - What is Docker ?
    - Command line program
    - Background daemon
    - Container runtime
      - Build Containers
      - Execute Containers
    - Reference implementation for container images and runtimes
    - Origins
      - Sun Solaris
      - Linux namespaces
    - Not Linux specific e.g. Windows
    - Alternative container runtimes/tools e.g. CRI-O, Podman, BuildAh, Skopeo
    - Removal of Docker from OpenShift
  - Containers
    - Constrained application environment
      - Can only access permitted resources
      - Has visibility to limited resources
        - Processes
        - Network
        - Files systems
        - IPC
        - Etc…
    - Isolation from other applications
      - Security/risk management
      - Reduction of conflicts e.g dependency versions, resource usage
    - Contains application and dependencies
      - Application/dependencies packaged as a unit
      - More portable - fast deployment e.g. Dockerhub images
    - Alternative to using Virtual Machines in many use cases
      - Reduce processing/memory overhead of nested OS instances
      - Reduce operating system instances to maintain
  - Building Docker images
    - Images and containers
      - Images
        - Application
        - Dependencies
        - OS Distro e.g. Alpine
        - Access control to external resources
          - Volumes
          - Network access
      - Containers
        - Instantiated image

- Application data
- Multiple processes e.g. Application and Systemd
- Mapping of container resources to OS resources
  - Access to persistent data
  - Network access

Building images
- Image build process
  - Using a base image
  - Execution
    - Creation of file system layers
    - Copy/checkout/build of resources
    - Execution of build
    - Tags
- Dockerfile
  - Commands

Demo of Docker build
- Review Dockerfile of typical app
  - Copy resources
    - Code checkout from Git
  - App build
    - Optimize app build to reduce execution time e.g. cache resources
    - Make build repeatable e.g. NPM CI and package-lock.json
- Run image build
- Show image stored in local environment

Running Image in a container
- Run image mapping network and file access
  - Execution options e.g. restrict CPU
- Access image shell and show limited environment
- Stop/Start of image
- Debugging images
  - Changing ENTRYPOINT

Image registries