CLOUD COMPUTING APPLICATIONS

Docker Networking

Prof. Reza Farivar

Docker Networking

- Pluggable, using drivers
  - Bridge
  - Host
  - Overlay
  - macvlan
  - None

Docker Bridge Networking

- `$ docker network create my-net`
-
```
$ docker create --name my-nginx \
--network my-net \
--publish 8080:80 \
nginx:latest
```
- connects Nginx container to the `my-net` network
- Any other container connected to the `my-net` network has access to all ports on the `my-nginx` container, and vice versa.
- It ***also*** publishes port 80 in the container to port 8080 on the Docker host, so external clients can access that port
- User-defined bridge networks provide DNS resolution

Docker Overlay Network Driver

- creates a distributed network among multiple Docker daemon hosts
- sits on top of (overlays) the host-specific networks
- Docker transparently handles routing of each packet to and from the correct Docker daemon host and the correct destination container
- In a swarm, two overlay networks play a role
  - `ingress`, which handles control and data traffic related to swarm services.
  - User-defined overlay networks
    - `$ docker network create -d overlay my-overlay`

User-defined Overlay Networks

- Swarm services connected to the same overlay network effectively expose all ports to each other
- Encryption
  - All swarm service management traffic is encrypted by default
    - using the AES Algorithm in GCM mode
  - To encrypt application data: `--opt encrypted`
- Subnet CIDR
  - ```
    $ docker network create \
     --driver overlay \
     --subnet=10.11.0.0/16 \
     --gateway=10.11.0.2 \
     --opt encrypted \
     my-overlay
    ```
    ```