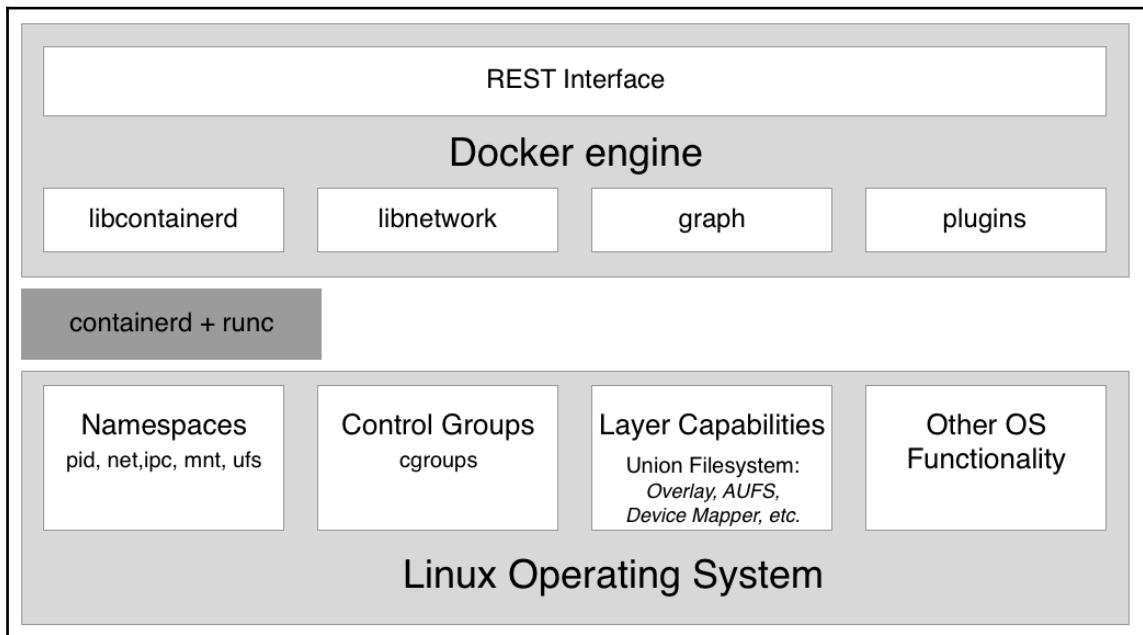


Chapter 1: What Are Containers and Why Should I Use Them?



Chapter 2: Setting up a Working Environment

```
$ docker-machine ls
NAME      ACTIVE   DRIVER      STATE      URL
default    -        virtualbox  Running    tcp://192.168.99.100:2376
$ █
```

```
$ docker-machine create --driver virtualbox default
Running pre-create checks...
Creating machine...
(default) Copying /Users/gabriel/.docker/machine/cache/boot2docker.iso to /Users/gabriel/.docker/machine/
machines/default/boot2docker.iso...
(default) Creating VirtualBox VM...
(default) Creating SSH key...
(default) Starting the VM...
(default) Check network to re-create if needed...
(default) Waiting for an IP...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with boot2docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: docker
r-machine env default
$ █
```

```
docker@default:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
ca4f61b1923c: Pull complete
Digest: sha256:97ce6fa4b6cdc0790cda65fe7290b74cfecd9fa0c9b8c38e979330d547d22ce1
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

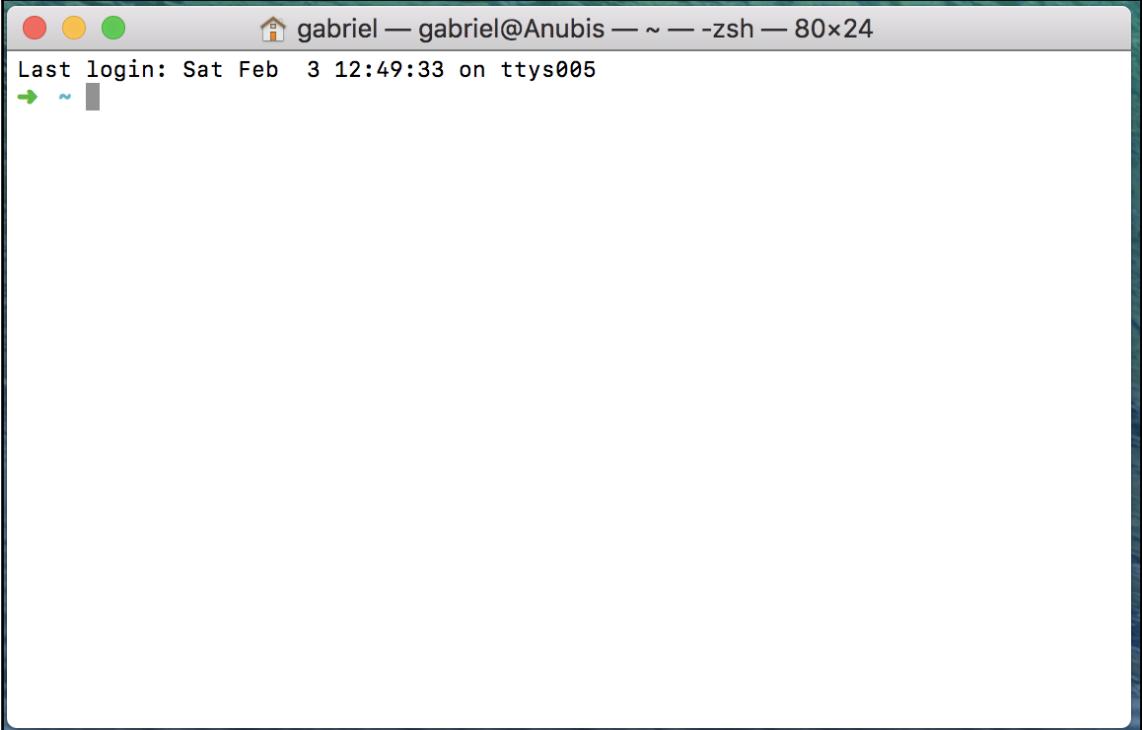
Share images, automate workflows, and more with a free Docker ID:

```
https://cloud.docker.com/
```

For more examples and ideas, visit:

```
https://docs.docker.com/engine/userguide/
```

```
docker@default:~$ █
```



```
$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
ca4f61b1923c: Pull complete
Digest: sha256:97ce6fa4b6cdc0790cda65fe7290b74cfecd9fa0c9b8c38e979330d547d22ce1
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

<https://cloud.docker.com/>

For more examples and ideas, visit:

<https://docs.docker.com/engine/userguide/>

```
$ █
```

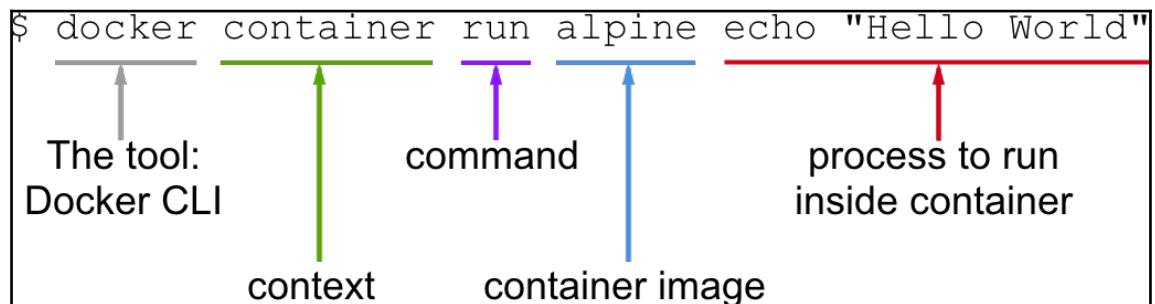
```
$ docker version
Client:
Version:          18.03.0-ce-rc4
API version:      1.37
Go version:       go1.9.4
Git commit:       fbedb97
Built: Thu Mar 15 07:33:28 2018
OS/Arch:          darwin/amd64
Experimental:     false
Orchestrator:    swarm

Server:
Engine:
Version:          18.03.0-ce-rc4
API version:      1.37 (minimum version 1.12)
Go version:       go1.9.4
Git commit:       fbedb97
Built:            Thu Mar 15 07:42:29 2018
OS/Arch:          linux/amd64
Experimental:     true
$
```

```
Starting local Kubernetes v1.9.0 cluster...
Starting VM...
Downloading Minikube ISO
 142.22 MB / 142.22 MB [=====] 100.00% 0s
Getting VM IP address...
Moving files into cluster...
Downloading localkube binary
 162.41 MB / 162.41 MB [=====] 100.00% 0s
 0 B / 65 B [-----] 0.00%
 65 B / 65 B [=====] 100.00% 0sSetting up certs...
Connecting to cluster...
Setting up kubeconfig...
Starting cluster components...
Kubectl is now configured to use the cluster.
Loading cached images from config file.
$ █
```

```
$ kubectl version
Client Version: version.Info{Major:"1", Minor:"9", GitVersion:"v1.9.0", GitCommit:"925c127ec6b946659ad0fd596fa959be43f0cc05"
, GitTreeState:"clean", BuildDate:"2017-12-15T21:07:38Z", GoVersion:"go1.9.2", Compiler:"gc", Platform:"darwin/amd64"}
Server Version: version.Info{Major:"", Minor:"", GitVersion:"v1.9.0", GitCommit:"925c127ec6b946659ad0fd596fa959be43f0cc05",
GitTreeState:"clean", BuildDate:"2018-01-26T19:04:38Z", GoVersion:"go1.9.1", Compiler:"gc", Platform:"linux/amd64"}
$ █
```

Chapter 3: Working with Containers

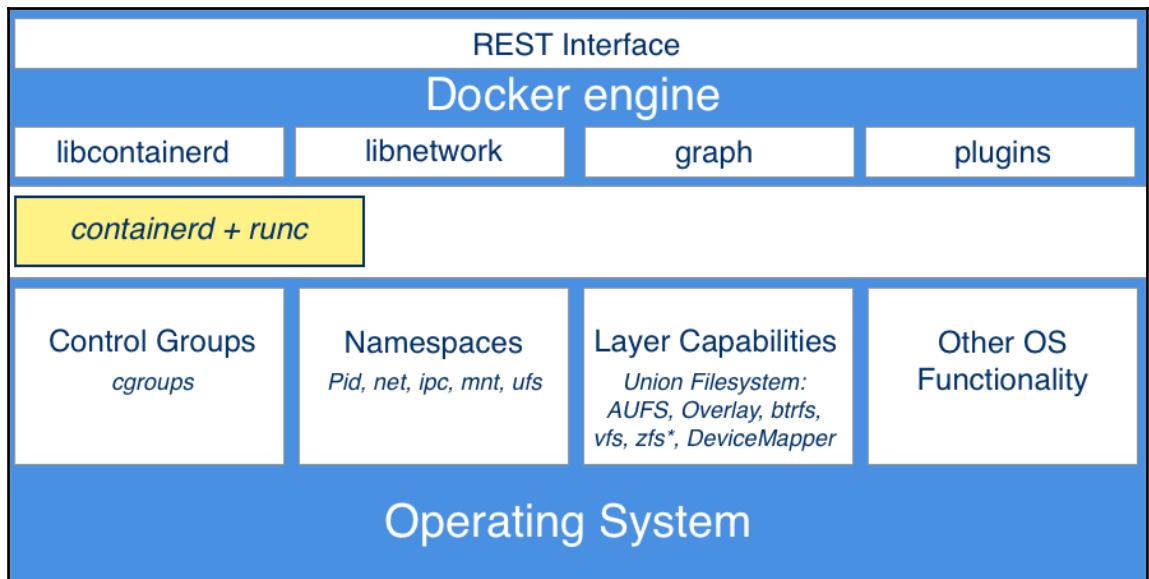


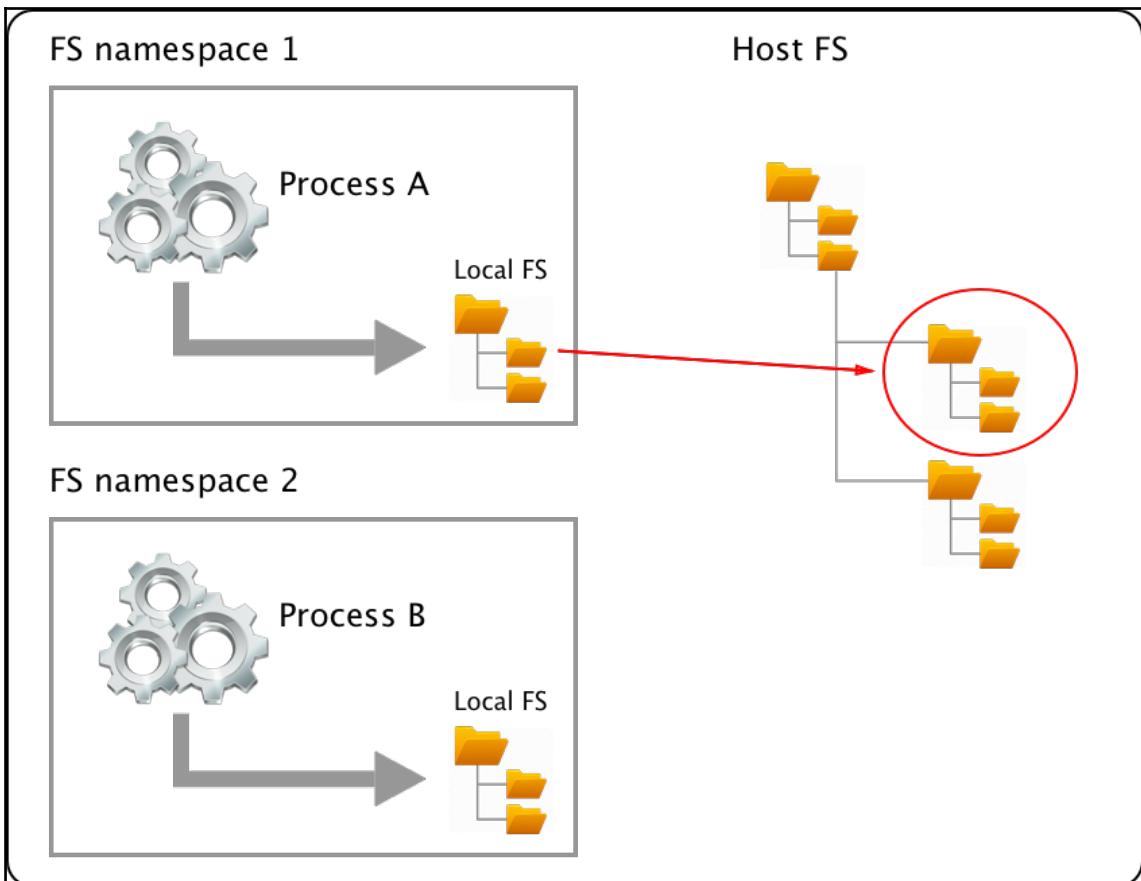
```
$ docker container ls -l
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS               NAMES
6ce5e46da7ce        alpine              "/bin/sh -c 'while :...'"   41 seconds ago    Up 16 seconds          quotes
```

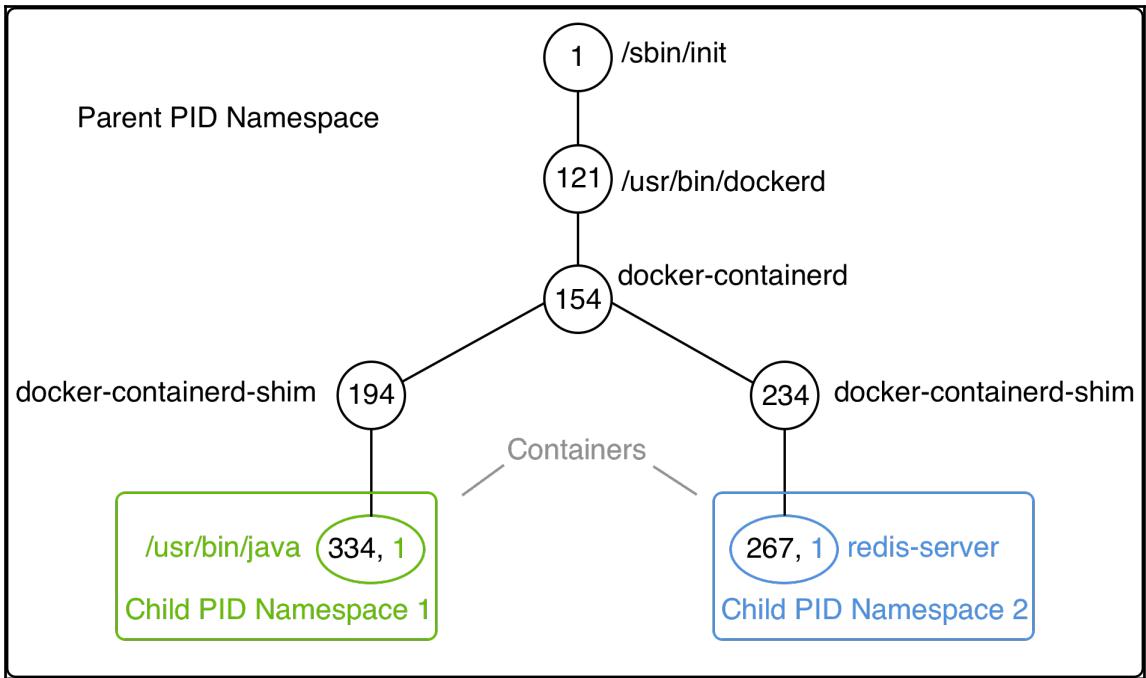
```
$ docker container ls
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS               NAMES
31d719b2f439        nginx:alpine        "nginx -g 'daemon of..."  35 seconds ago    Up 30 seconds         80/tcp              cranky_curie
27b96de70b58        alpine:latest       "ping 127.0.0.1"        23 hours ago     Up 23 hours          c2
35b8dd512acb        alpine:latest       "/bin/sh"              23 hours ago     Up 23 hours          c1
```

```
/ # ps
 PID  USER      TIME  COMMAND
   1 root      0:00  /bin/sh -c while :; do wget -qO- https://talaikis.com/api
   85 root      0:00  /bin/sh
  110 root      0:00  sleep 5
  111 root      0:00  ps
```

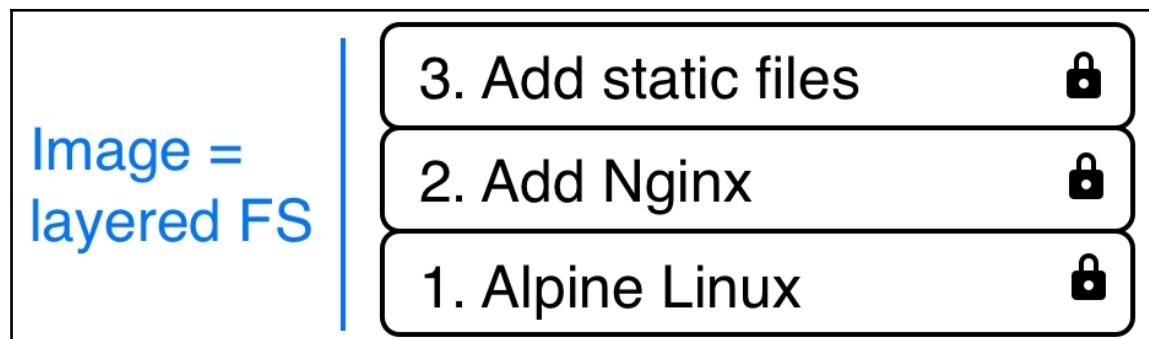
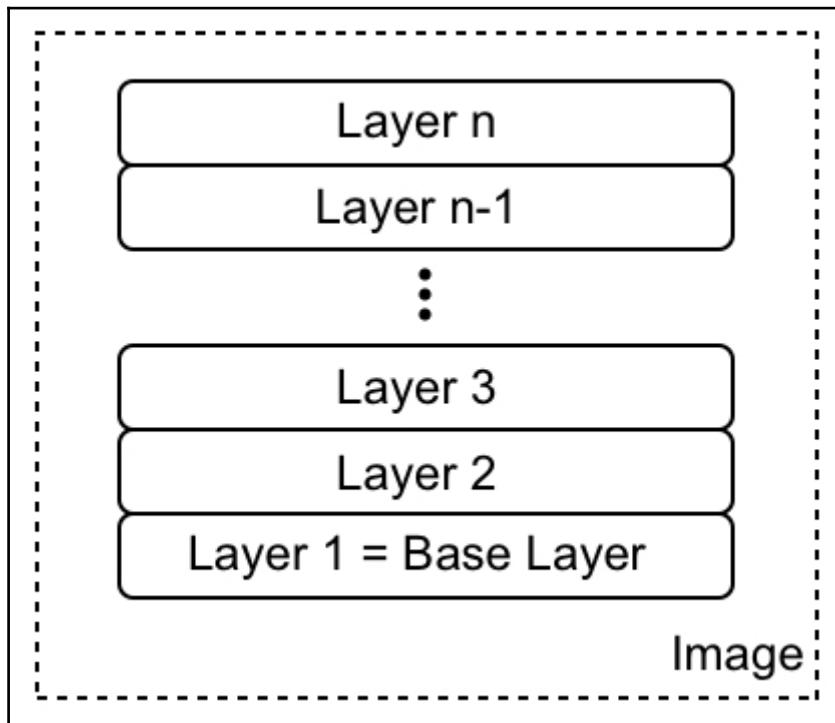
```
$ docker container exec quotes ps
PID  USER      TIME  COMMAND
   1 root      0:00  /bin/sh -c while :; do wget -qO- https://talaikis.com/api
   520 root      0:00  sleep 5
  521 root      0:00  ps
```







Chapter 4: Creating and Managing Container Images



Container Layer

r/w

3. Add static files



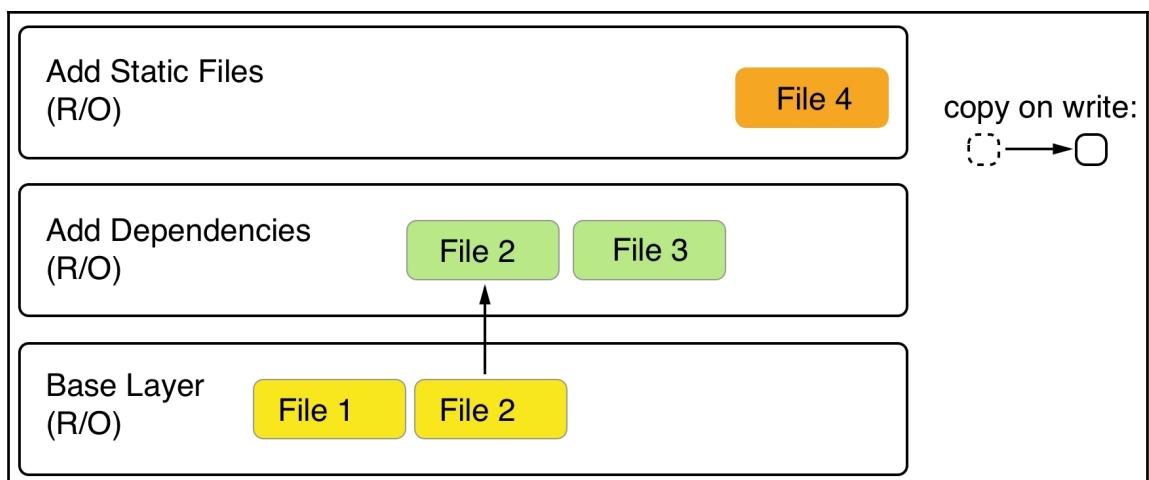
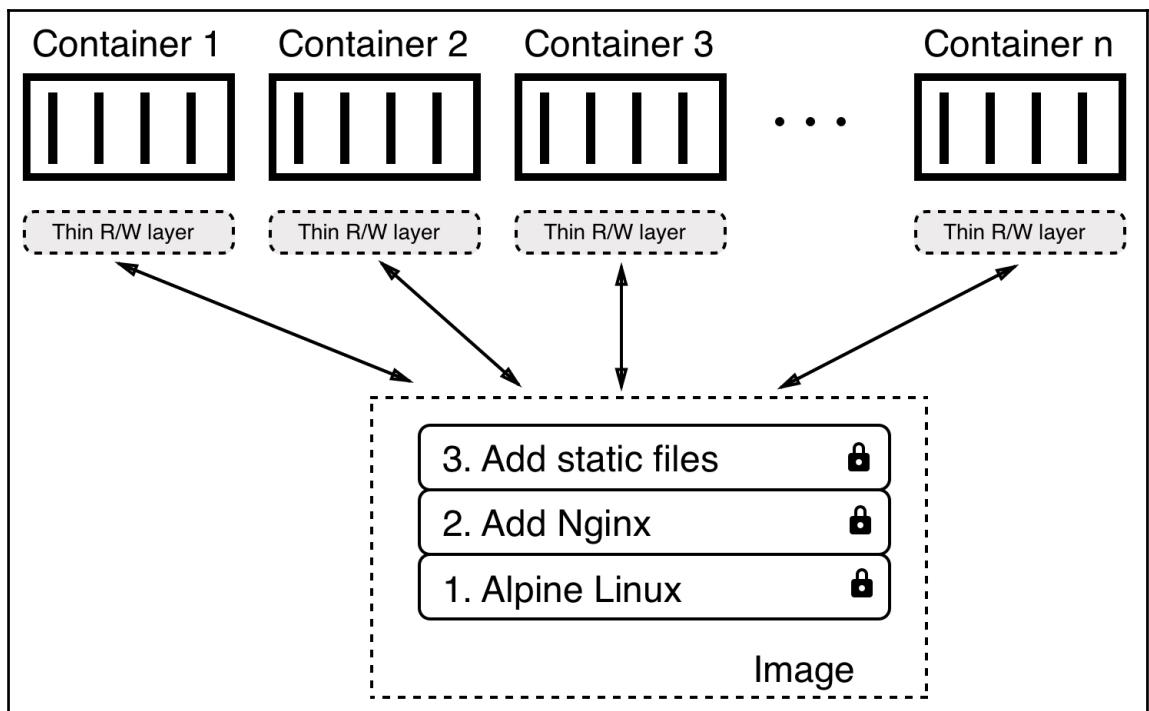
2. Add Nginx

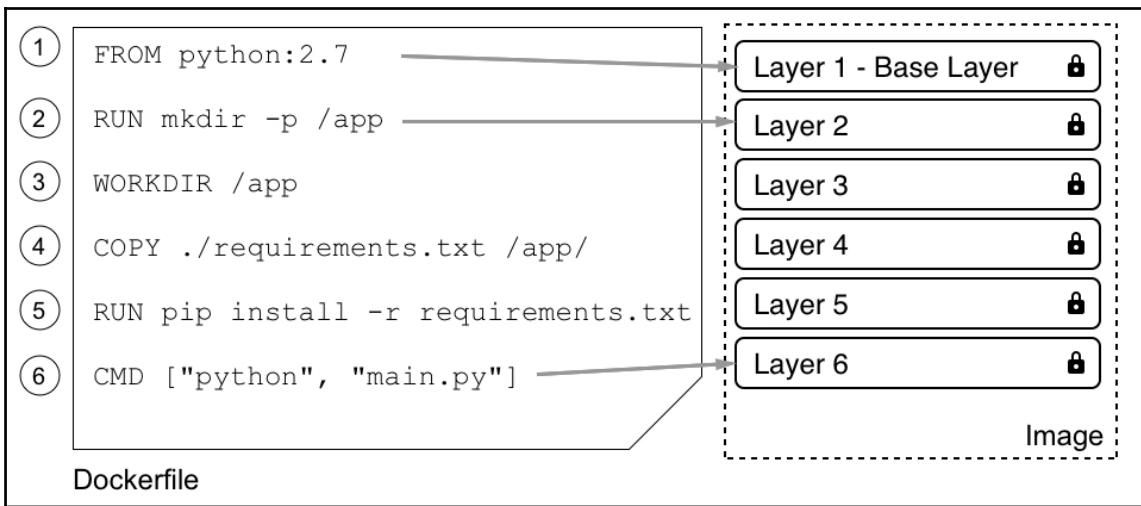


1. Alpine Linux



Container



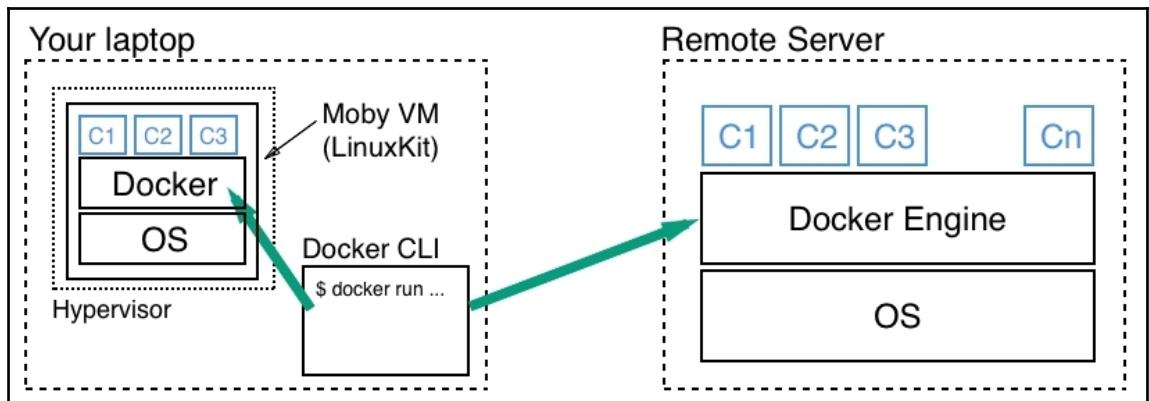


Step	Builder Container	Command	Resulting Image
①	Pull base Image		Layer 1 - Base Layer
②	Container Layer r/w Layer 1 - Base Layer	RUN mkdir -p /app	Layer 2 Layer 1 - Base Layer
③	Container Layer r/w Layer 2 Layer 1 - Base Layer	WORKDIR /app	Layer 3 Layer 2 Layer 1 - Base Layer
④	⋮	⋮	⋮
⑦	Container Layer r/w Layer 6 Layer 5 Layer 4 Layer 3 Layer 2 Layer 1 - Base Layer	CMD ["python", "main.py"]	Layer 7 Layer 6 Layer 5 Layer 4 Layer 3 Layer 2 Layer 1 - Base Layer Final Image

Chapter 5: Data Volumes and System Management

```
$ docker version
Client:
  Version:          18.04.0-ce
  API version:     1.37
  Go version:       go1.9.4
  Git commit:      3d479c0
  Built: Tue Apr 10 18:13:16 2018
  OS/Arch:         darwin/amd64
  Experimental:    true
  Orchestrator:   swarm

Server:
  Engine:
    Version:          18.04.0-ce
    API version:     1.37 (minimum version 1.12)
    Go version:       go1.9.4
    Git commit:      3d479c0
    Built: Tue Apr 10 18:23:05 2018
    OS/Arch:         linux/amd64
    Experimental:    true
$ █
```



```
$ docker system info
Containers: 1
  Running: 0
  Paused: 0
  Stopped: 1
Images: 70
Server Version: 18.04.0-ce
Storage Driver: overlay2
  Backing Filesystem: extfs
  Supports d_type: true
  Native Overlay Diff: true
Logging Driver: json-file
Cgroup Driver: cgroupfs
Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journalctl json-file logentries splunk syslog
Swarm: inactive
Runtimes: runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 773c489c9c1b21a6d78b5c538cd395416ec50f88
runc version: 4fc53a81fb7c994640722ac585fa9ca548971871
init version: 949e6fa
Security Options:
  seccomp
    Profile: default
Kernel Version: 4.9.87-linuxkit-aufs
Operating System: Docker for Mac
OSType: linux
Architecture: x86_64
CPUs: 4
Total Memory: 1.952GiB
Name: linuxkit-025000000001
ID: WV5X:CY7N:LHIP:SWJ2:T55W:P5QM:MEYU:MM3V:550H:RALF:5ZDN:QH7Y
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): true
  File Descriptors: 22
  Goroutines: 42
  System Time: 2018-04-21T12:08:17.962868Z
  EventsListeners: 2
HTTP Proxy: gateway.docker.internal:3128
HTTPS Proxy: gateway.docker.internal:3129
Registry: https://index.docker.io/v1/
Labels:
Experimental: true
Insecure Registries:
  127.0.0.0/8
Live Restore Enabled: false
```

```

$ docker system df -v
Images space usage:

REPOSITORY          TAG        IMAGE ID      CREATED ago    SIZE     SHARED SIZE   UNIQUE SIZE  CONTAINERS
fundamentalsofdocker/ch14-web  1.0        fe6612f845be  12 days ago  72.05MB  72.05MB    1.834kB    0
fundamentalsofdocker/web       2.0        9445c23644d7  12 days ago  72.05MB  72.05MB    1.827kB    0
fundamentalsofdocker/ch13-web  2.0        b47675c8e53c  2 weeks ago   72.05MB  72.05MB    1.836kB    0
builder2              latest     074a85b21f3a  3 weeks ago   1.514GB   1.456GB    58.7MB     0
builder              latest     9864221c5187  3 weeks ago   1.464GB   1.456GB    8.051MB    0
ruby                 alpine    b620ae34414c  4 weeks ago   55.52MB   4.148MB    51.38MB    1
microsoft/azure-cli      latest     a52f6e53d04c  4 weeks ago   400.4MB   0B         400.4MB    0
nginx                alpine    91ce6206f9d8  4 weeks ago   18MB     4.148MB    13.86MB    1
perl                 5.26      3c2c43b2e15   5 weeks ago   879.2MB   879.2MB    0B         0
fundamentalsofdocker/ch08-web  1.0        92e085ed002   7 weeks ago   72.01MB   68.02MB    3.992MB    0
fundamentalsofdocker/ch08-db   1.0        4953d535c17   7 weeks ago   39.46MB   4.148MB    35.31MB    0
node                 9.6-alpine  a88ff852e3d4   8 weeks ago   68.02MB   68.02MB    0B         0
alpine               latest     3fd90b5eadf02  3 months ago  4.148MB   4.148MB    0B         0
confluentinc/cp-enterprise-kafka  4.0.0     07d41f8648f5  3 months ago  565.1MB   0B         565.1MB    0
hello-world           latest     f2a91732366c  5 months ago  1.848kB   0B         1.848kB    0
hseeberger/scala-sbt      latest     da0e1be3bb79   9 months ago  925.4MB   0B         925.4MB    0

Containers space usage:

CONTAINER ID        IMAGE      COMMAND      LOCAL VOLUMES   SIZE     CREATED ago    STATUS      NAMES
afe0dcab9bc4      nginx:alpine "ping 8.8.8.8"  0        0B     6 seconds ago  Up 4 seconds xenodochial_easley
2a2d742604af      ruby:alpine  "/bin/sh"    0        16.7MB  5 hours ago   Exited (0) 13 minutes ago keen_lumiere

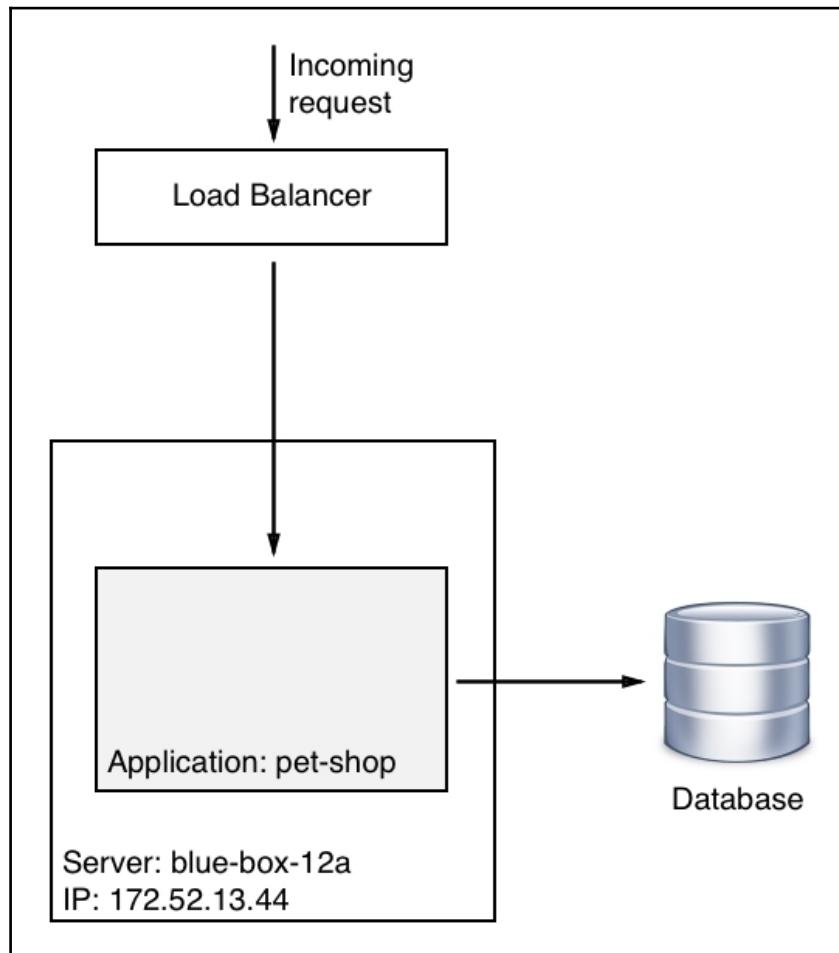
Local Volumes space usage:

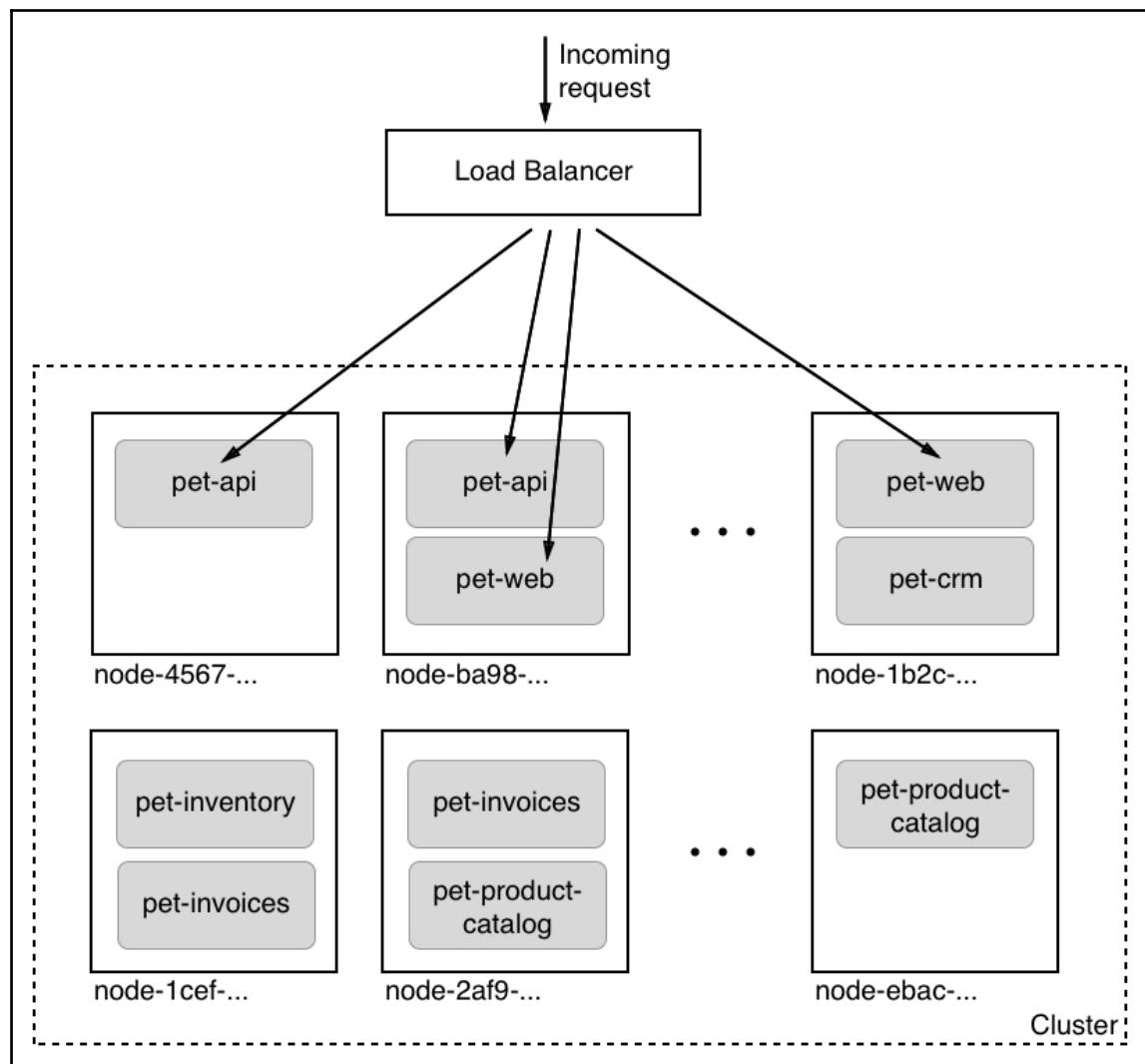
VOLUME NAME        LINKS      SIZE
ch08_pets-data     0          47.24MB

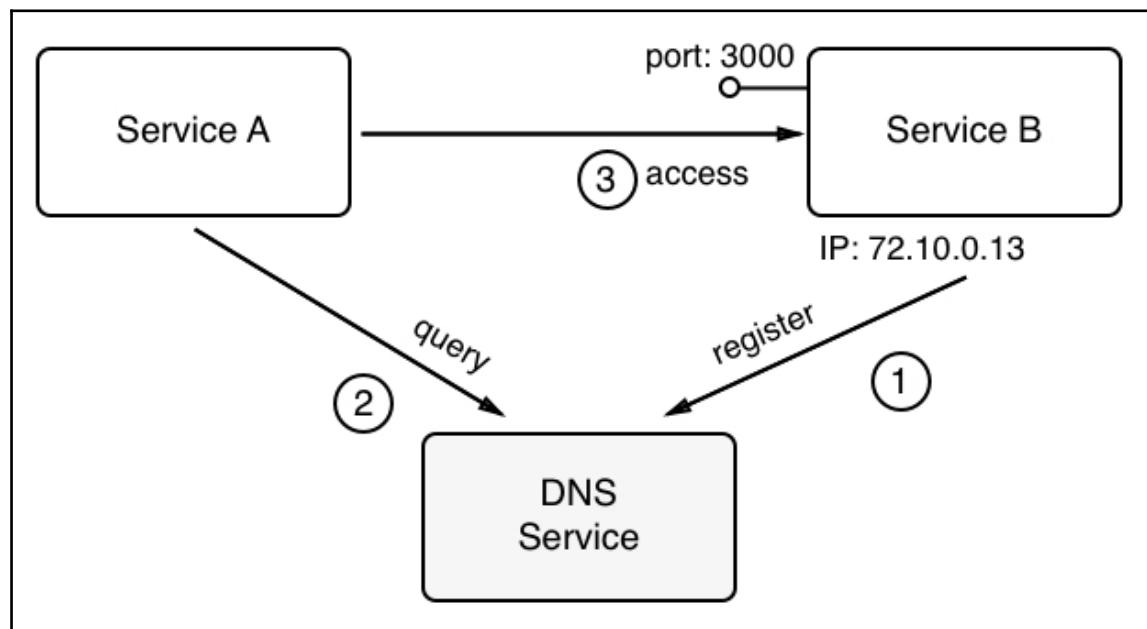
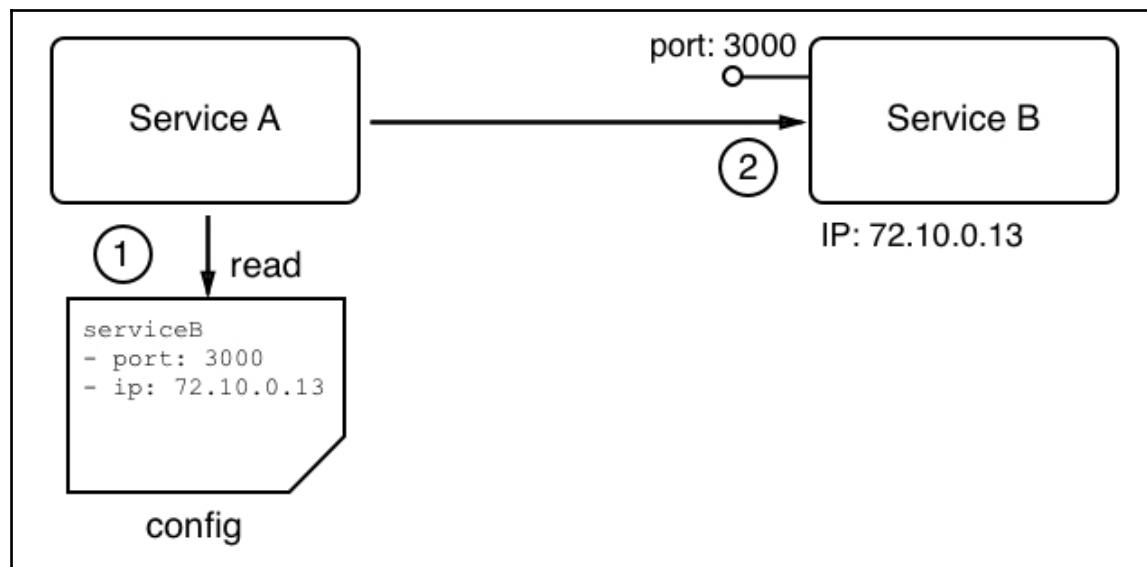
Build cache usage: 0B
$ 

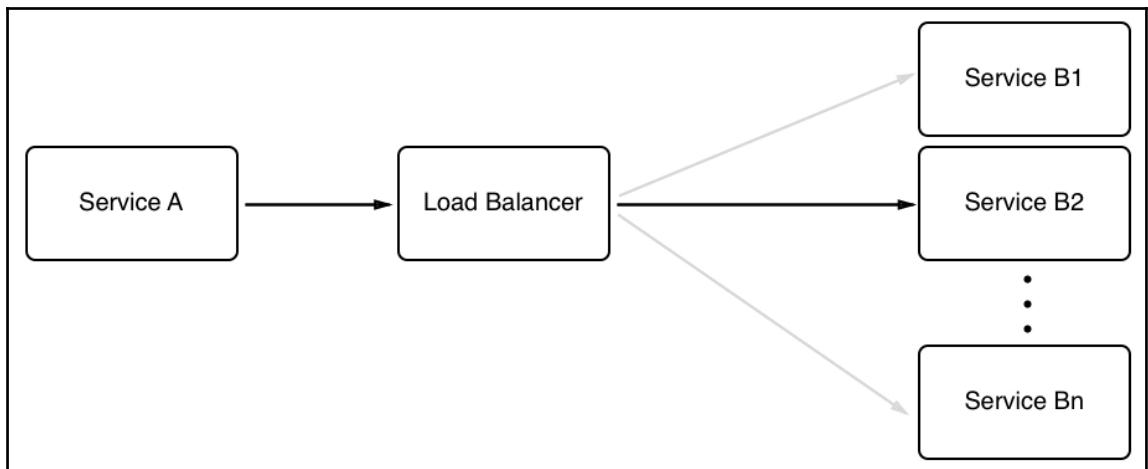
```

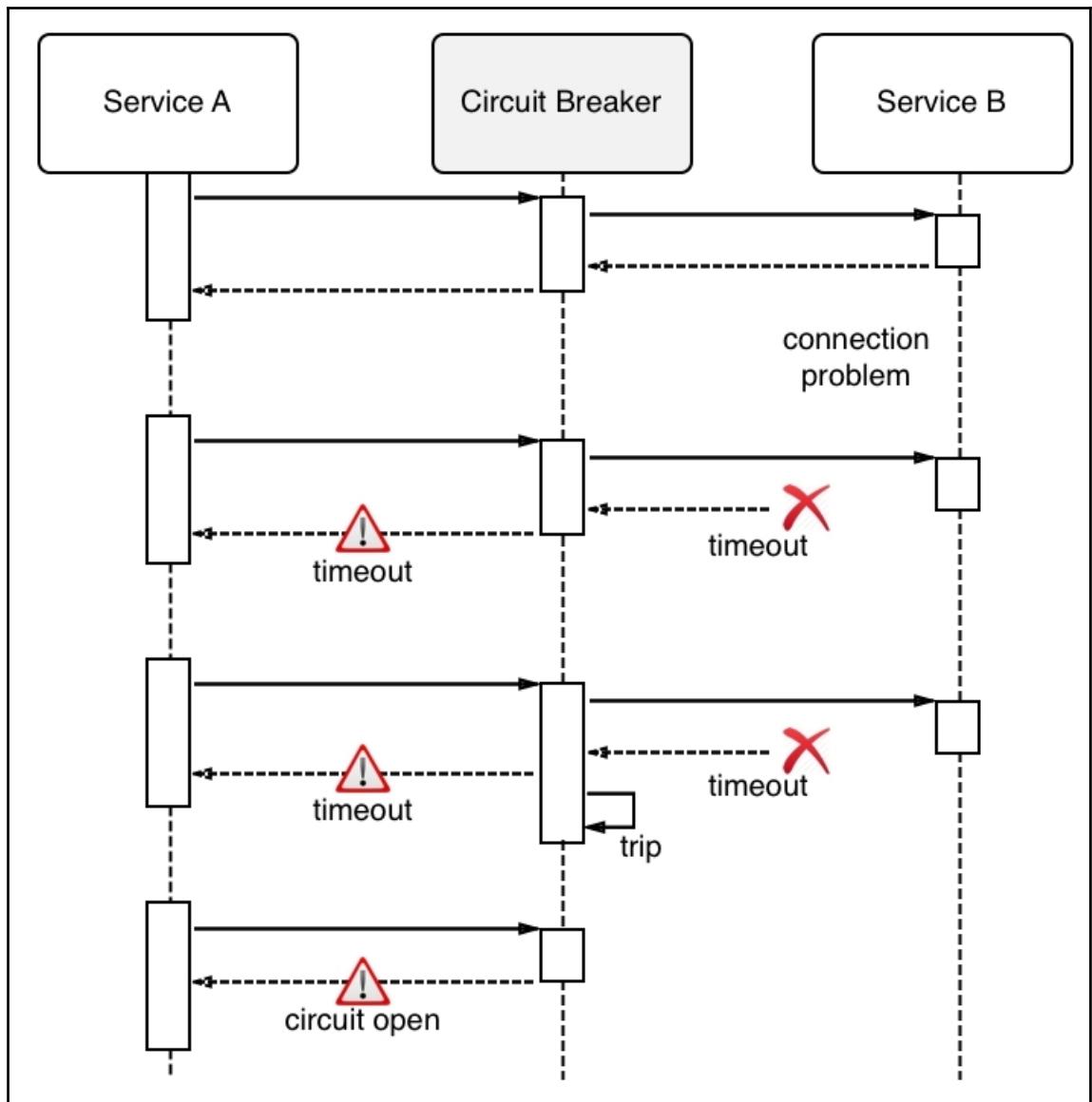
Chapter 6: Distributed Application Architecture

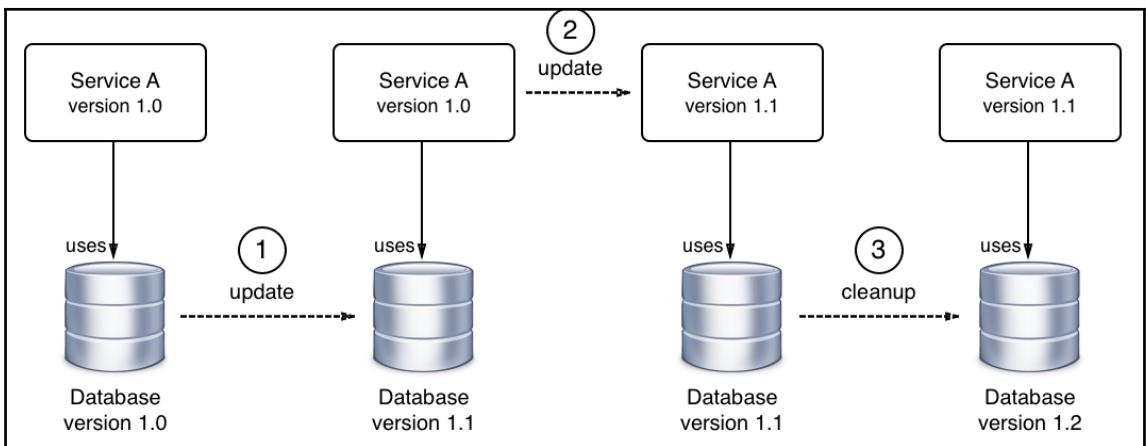
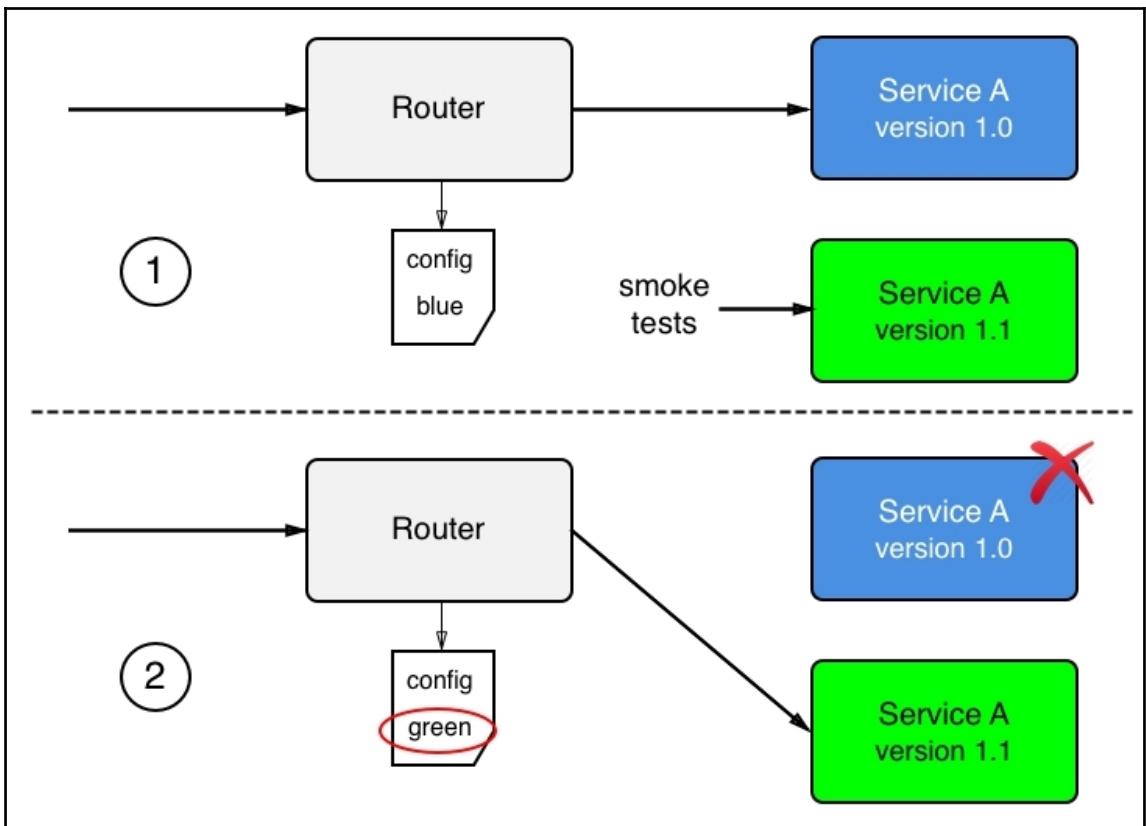




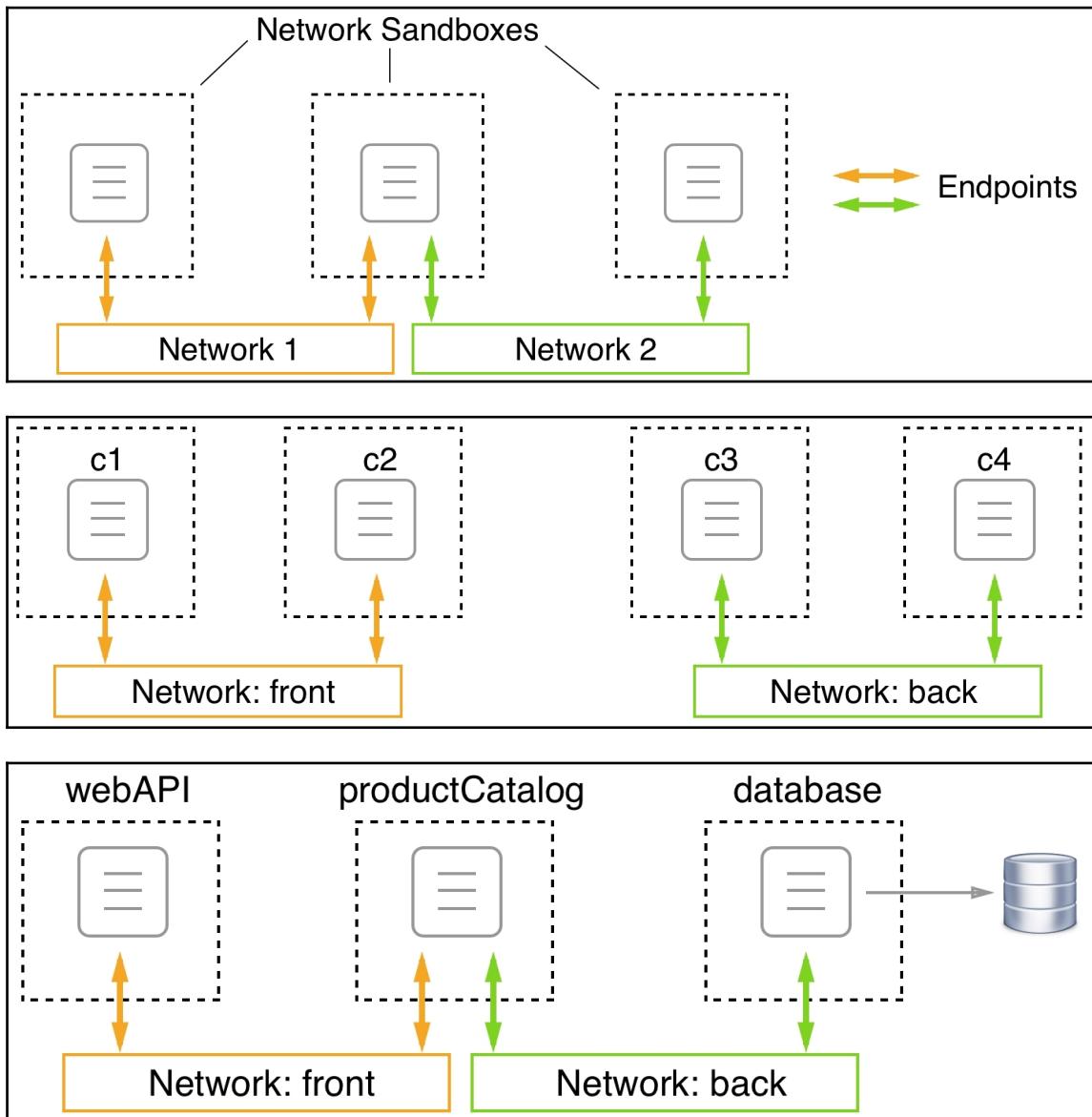








Chapter 7: Single-Host Networking

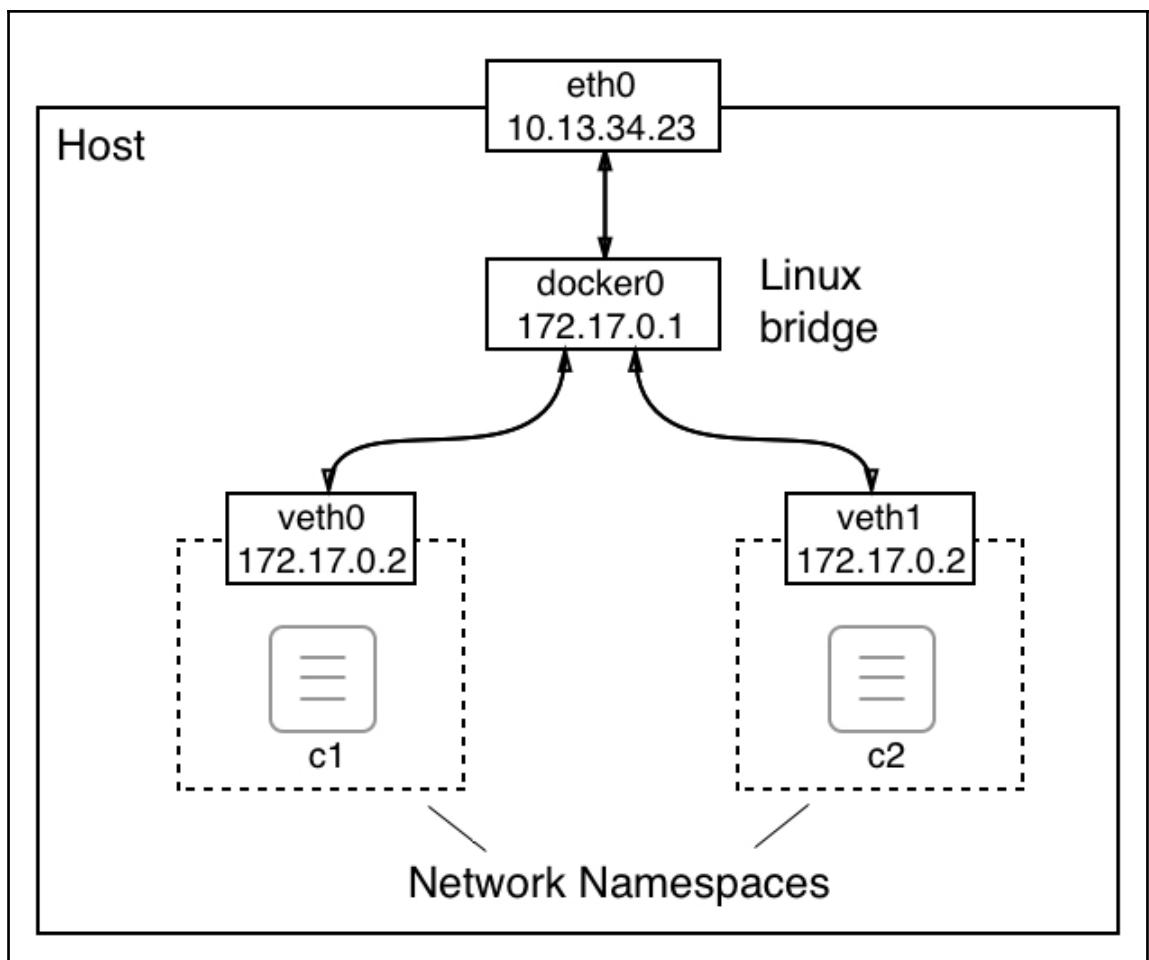
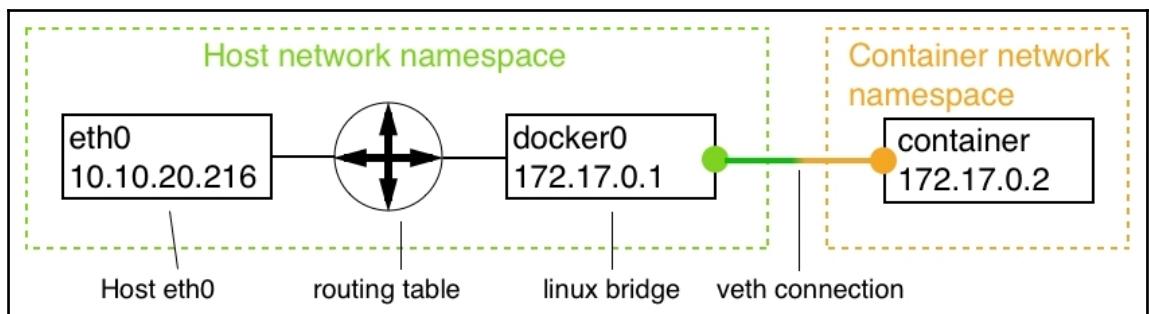


```
$ docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
928c8ce47bf2	bridge	bridge	local
bdb36adcf70c	host	host	local
af82006f2f2d	none	null	local

```
$ █
```

```
C:\Users\admin>docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "3b08c1c711ada84ae859c4bed48b5af1f45b68db89356ca5045dc7ee8672e946",
        "Created": "2018-04-09T09:47:29.9424652Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "com.docker.network.bridge.enable_ip_masquerade": "true",
            "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
            "com.docker.network.bridge.name": "docker0",
            "com.docker.network.driver.mtu": "1500"
        },
        "Labels": {}
    }
]
```

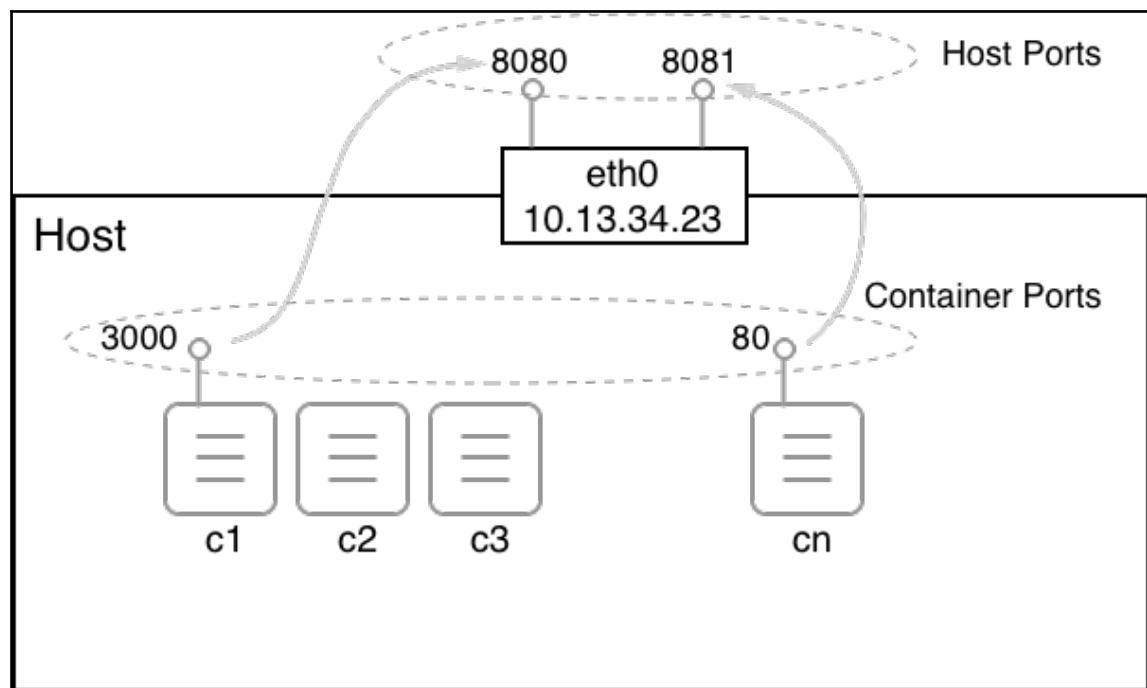
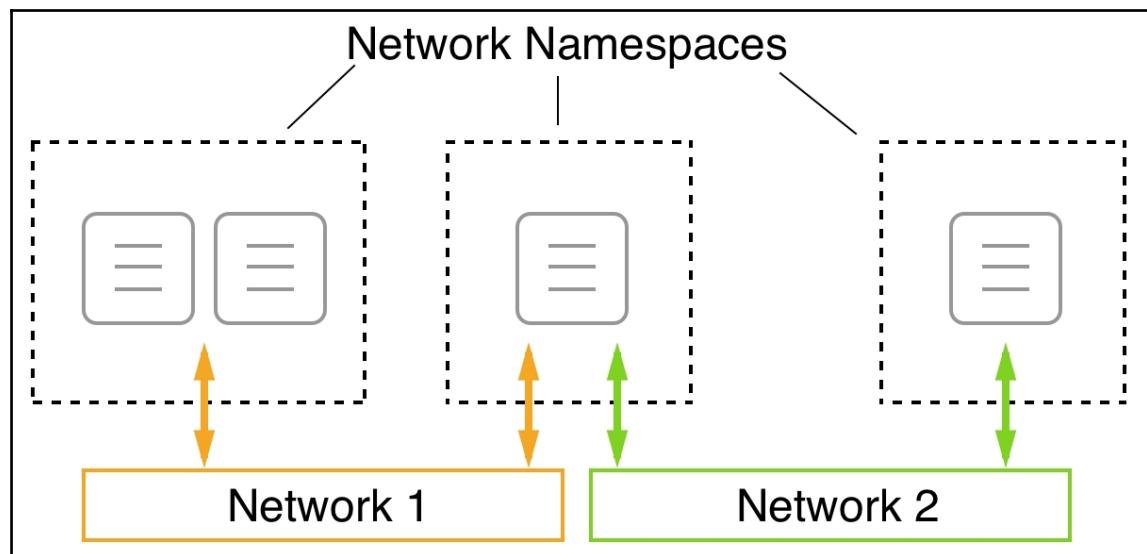


```
        },
        "NetworkSettings": {
            "Bridge": "",
            "SandboxID": "ae53496fba49de3d0a4727105cc0799b7fdb30746d76700238cb47c611f3eb68",
            "HairpinMode": false,
            "LinkLocalIPv6Address": "",
            "LinkLocalIPv6PrefixLen": 0,
            "Ports": {},
            "SandboxKey": "/var/run/docker/netns/ae53496fba49",
            "SecondaryIPAddresses": null,
            "SecondaryIPv6Addresses": null,
            "EndpointID": "c063a725d1f66e867b5769a80d1477cc88d07618860655fa3033a97478e55713",
            "Gateway": "172.17.0.1",
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.4",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "MacAddress": "02:42:ac:11:00:04",
            "Networks": {
                "bridge": {
                    "IPAMConfig": null,
                    "Links": null,
                    "Aliases": null,
                    "NetworkID": "026e653c2504e464748b4ce9b25cce69d29bc82a52105a25920f2b796663e635",
                    "EndpointID": "c063a725d1f66e867b5769a80d1477cc88d07618860655fa3033a97478e55713",
                    "Gateway": "172.17.0.1",
                    "IPAddress": "172.17.0.4",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
                    "MacAddress": "02:42:ac:11:00:04",
                    "DriverOpts": null
                }
            }
        }
    }
```

```
/ # ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: tunl0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN qlen 1
    link/pipe 0.0.0.0 brd 0.0.0.0
3: ip6tnl0@NONE: <NOARP> mtu 1452 qdisc noop state DOWN qlen 1
    link/tunnel6 00:00:00:00:00:00 brd 00:00:00:00:00:00
19: eth0@if20: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:ac:11:00:04 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.4/16 brd 172.17.255.255 scope global eth0
        valid_lft forever preferred_lft forever
/ #
```

```
"ConfigOnly": false,
"Containers": {
    "27b96de70b58cd918d35c235a7c180f56f71df58cf4cec50b8f0103dd529b95f": {
        "Name": "c2",
        "EndpointID": "8883649774c5c4c53063da02598c8d09fe7ee427145b348b1d1703f31213e9ca",
        "MacAddress": "02:42:ac:11:00:03",
        "IPv4Address": "172.17.0.3/16",
        "IPv6Address": ""
    },
    "35b8dd512acb985647833e1cc52625e129c15e903fd8a0c0ab247932bc910166": {
        "Name": "c1",
        "EndpointID": "28269a9cc630135ab287052fa69c72f28c57a10bd5e7523c451bf2d0976fd1b5",
        "MacAddress": "02:42:ac:11:00:02",
        "IPv4Address": "172.17.0.2/16",
        "IPv6Address": ""
    }
},
"Options": {
```

```
"Containers": {
    "134295caa6012df5dc7d541436954af1a5264c6f69d5b8012e88f9c12faf40f1": {
        "Name": "c3",
        "EndpointID": "5693cd9329437a9ec1d27f439887bb0258837b9342a1c32204fa4571298457",
        "MacAddress": "02:42:0a:01:00:02",
        "IPv4Address": "10.1.0.2/16",
        "IPv6Address": ""
    },
    "4a277d33ebfb74f00d31be272d2d74cbfec4b17666e44d88e26cfe83b0a790cc": {
        "Name": "c4",
        "EndpointID": "a1e9ecafebdcf816261883c171434273d9973832d43255b5aa224b081853ed0f",
        "MacAddress": "02:42:0a:01:00:03",
        "IPv4Address": "10.1.0.3/16",
        "IPv6Address": ""
    }
}
```





If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

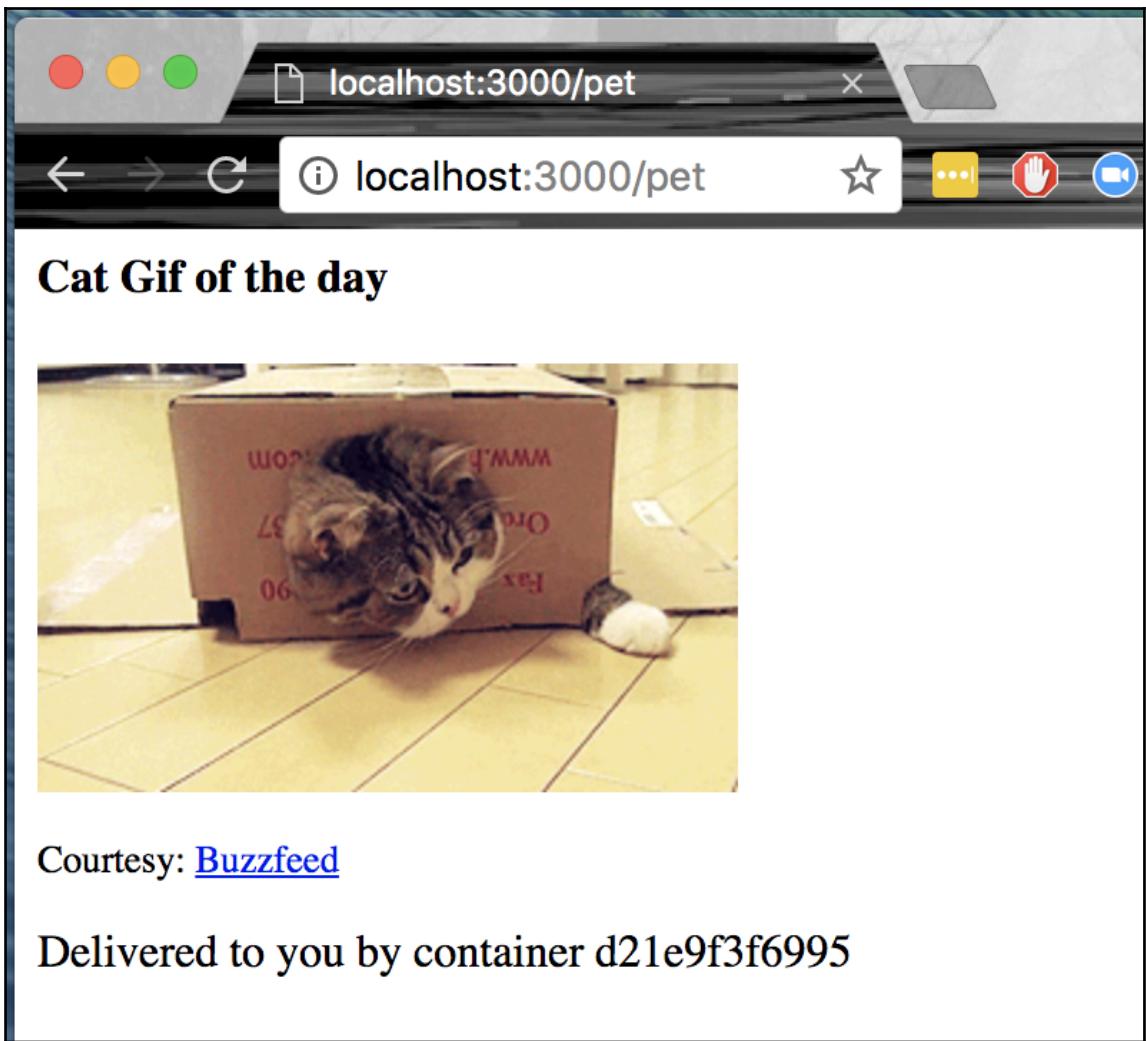
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Chapter 8: Docker Compose

```
$ docker-compose up
Creating network "ch08_default" with the default driver
Creating volume "ch08_pets-data" with default driver
Pulling web (fundamentalsofdocker/ch08-web:1.0)...
1.0: Pulling from fundamentalsofdocker/ch08-web
605ce1bd3f31: Pull complete
d9c1bb40879c: Pull complete
d610e8516793: Pull complete
bf3a86e46185: Pull complete
f082b7c3a97c: Pull complete
188ade417c9f: Pull complete
ad8771290e5e: Pull complete
Digest: sha256:d7978627352813340e8f9bbf700ecb39bece12873956e5b77dc5e6431e9126a8
Status: Downloaded newer image for fundamentalsofdocker/ch08-web:1.0
Pulling db (fundamentalsofdocker/ch08-db:1.0)...
1.0: Pulling from fundamentalsofdocker/ch08-db
ff3a5c916c92: Pull complete
a503b44e1ce0: Pull complete
211706713093: Pull complete
8df57d533e71: Pull complete
7858f71c02fb: Pull complete
55a8ef17ba59: Pull complete
3fb44f23d323: Pull complete
65cad41156b3: Pull complete
5492a5bead70: Pull complete
ac3385cd756f: Pull complete
Digest: sha256:eb5364a418bf7072de3e992517cad4ce8c55725a1cdfcd18e1c04ea2ec2a7356
Status: Downloaded newer image for fundamentalsofdocker/ch08-db:1.0
Creating ch08_db_1 ... done
Creating ch08_web_1 ... done
Attaching to ch08_db_1, ch08_web_1
```

```
db_1 | done
db_1 | server started
web_1 | Listening at 0.0.0.0:3000
db_1 | CREATE DATABASE
db_1 |
db_1 | CREATE ROLE
db_1 |
db_1 |
db_1 | /usr/local/bin/docker-entrypoint.sh: running /docker-entrypoint-initdb.d/init-db.sql
db_1 | CREATE TABLE
db_1 | ALTER TABLE
db_1 | ALTER ROLE
db_1 | INSERT 0 1
db_1 | waiting for server to shut down....2018-03-21 12:52:40.709 UTC [34] LOG: received fast shutdown
request
db_1 | 2018-03-21 12:52:40.711 UTC [34] LOG: aborting any active transactions
db_1 | 2018-03-21 12:52:40.712 UTC [34] LOG: worker process: logical replication launcher (PID 41) exi
ted with exit code 1
db_1 | 2018-03-21 12:52:40.712 UTC [36] LOG: shutting down
db_1 | 2018-03-21 12:52:40.737 UTC [34] LOG: database system is shut down
db_1 | done
db_1 | server stopped
db_1 |
db_1 | PostgreSQL init process complete; ready for start up.
db_1 |
db_1 | 2018-03-21 12:52:40.817 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
db_1 | 2018-03-21 12:52:40.817 UTC [1] LOG: listening on IPv6 address "::", port 5432
db_1 | 2018-03-21 12:52:40.821 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.543
2"
db_1 | 2018-03-21 12:52:40.832 UTC [49] LOG: database system was shut down at 2018-03-21 12:52:40 UTC
db_1 | 2018-03-21 12:52:40.835 UTC [1] LOG: database system is ready to accept connections
```



Courtesy: [Buzzfeed](#)

Delivered to you by container d21e9f3f6995

```
$ docker-compose up
Creating network "ch08_default" with the default driver
Creating ch08_web_1 ... done
Creating ch08_db_1 ... done
Attaching to ch08_web_1, ch08_db_1
db_1  | 2018-03-02 01:25:35.874 UTC [1] LOG:  listening on IPv4 address "0.0.0.0", port 5432
db_1  | 2018-03-02 01:25:35.875 UTC [1] LOG:  listening on IPv6 address "::", port 5432
db_1  | 2018-03-02 01:25:35.877 UTC [1] LOG:  listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
db_1  | 2018-03-02 01:25:35.890 UTC [19] LOG:  database system was shut down at 2018-03-02 01:25:23 UTC
db_1  | 2018-03-02 01:25:35.894 UTC [1] LOG:  database system is ready to accept connections
web_1 | Listening at 0.0.0.0:3000
```

```
$ docker-compose ps
      Name          Command     State        Ports
-----  
ch08_db_1    docker-entrypoint.sh postgres   Up          5432/tcp  
ch08_web_1   /bin/sh -c node src/server.js  Up          0.0.0.0:3000->3000/tcp  
$ █
```

```
$ docker-compose up --scale web=3
WARNING: The "web" service specifies a port on the host. If multiple containers for this
service are created on a single host, the port will clash.
Starting ch08_web_3 ...
Starting ch08_web_3 ... error

ERROR: for ch08_web_3  Cannot start service web: driver failed programming external conne
ctivity on endpoint ch08_web_3 (534216cc36e0284b775e48c6450e25ff21fe90ff6d7b8b9716f421cb9
8560351): Bind for 0.0.0.0:3000 failed: port is already allocated

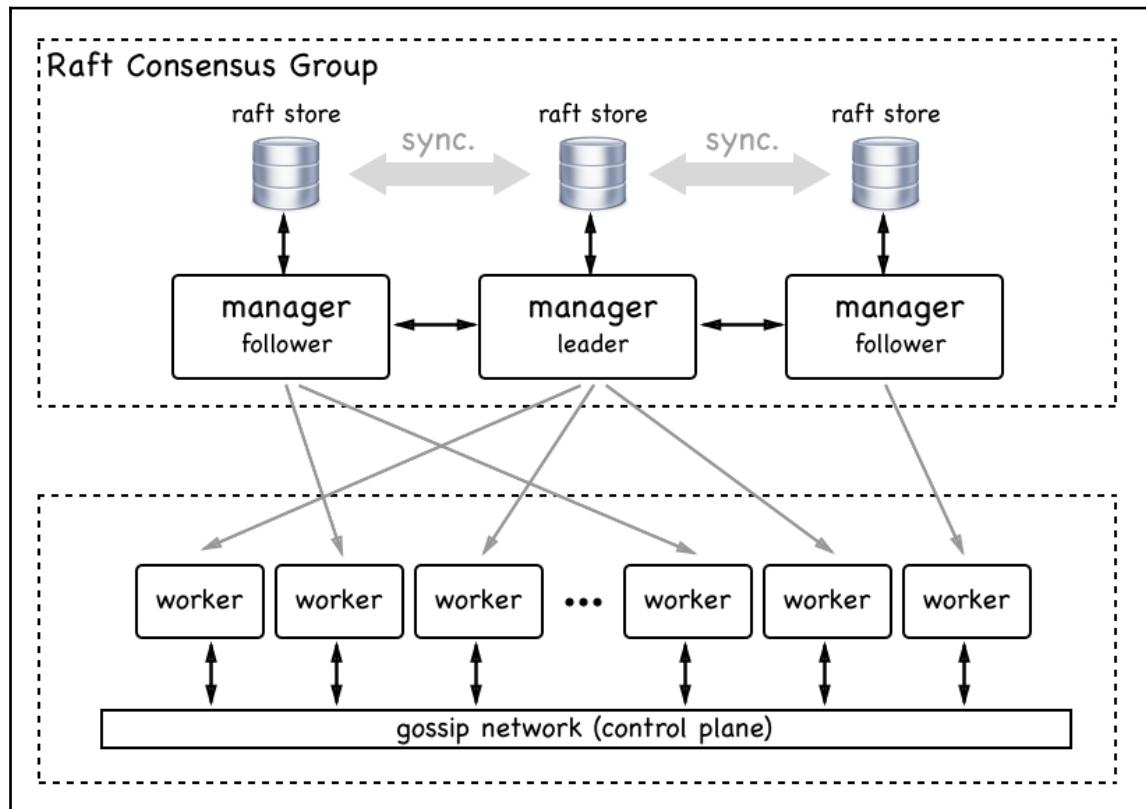
ERROR: for web  Cannot start service web: driver failed programming external connectivity
on endpoint ch08_web_3 (534216cc36e0284b775e48c6450e25ff21fe90ff6d7b8b9716f421cb98560351
): Bind for 0.0.0.0:3000 failed: port is already allocated
ERROR: Encountered errors while bringing up the project.
$ █
```

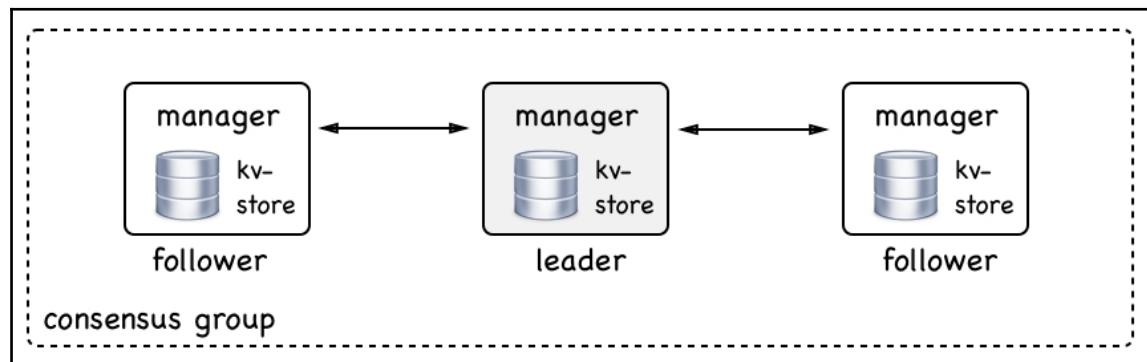
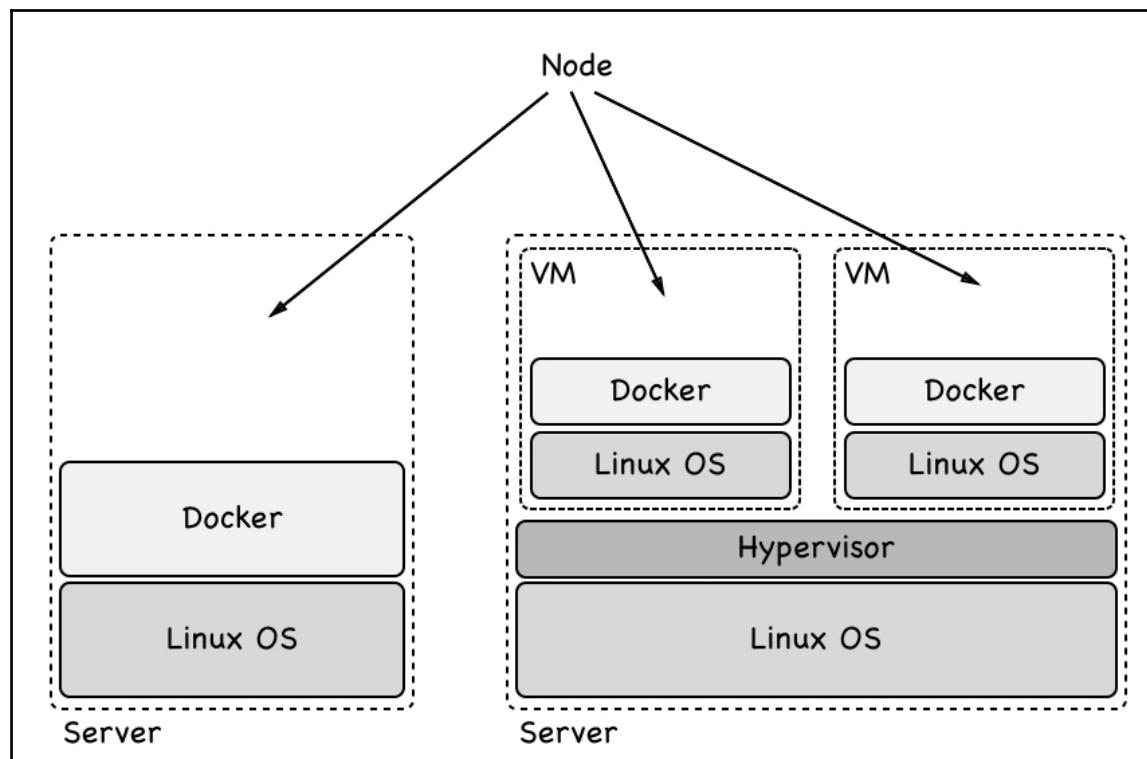
```
$ docker-compose ps
      Name          Command     State        Ports
-----  
ch08_db_1    docker-entrypoint.sh postgres   Up          5432/tcp  
ch08_web_1   /bin/sh -c node src/server.js  Up          0.0.0.0:32769->3000/tcp  
ch08_web_2   /bin/sh -c node src/server.js  Up          0.0.0.0:32771->3000/tcp  
ch08_web_3   /bin/sh -c node src/server.js  Up          0.0.0.0:32770->3000/tcp  
$ █
```

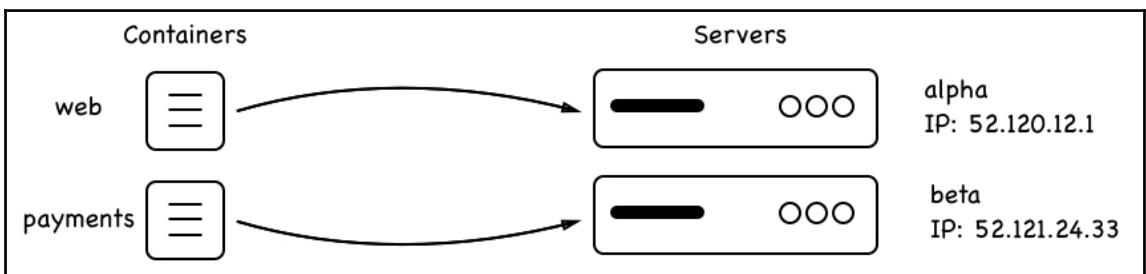
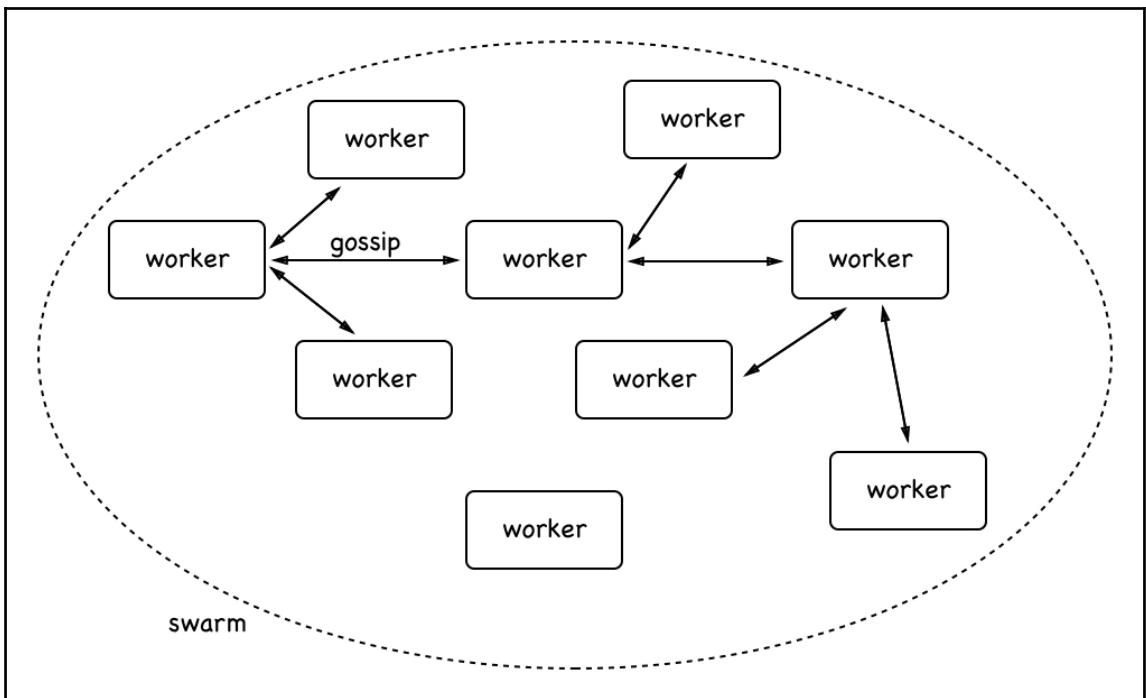
Chapter 9: Orchestrators

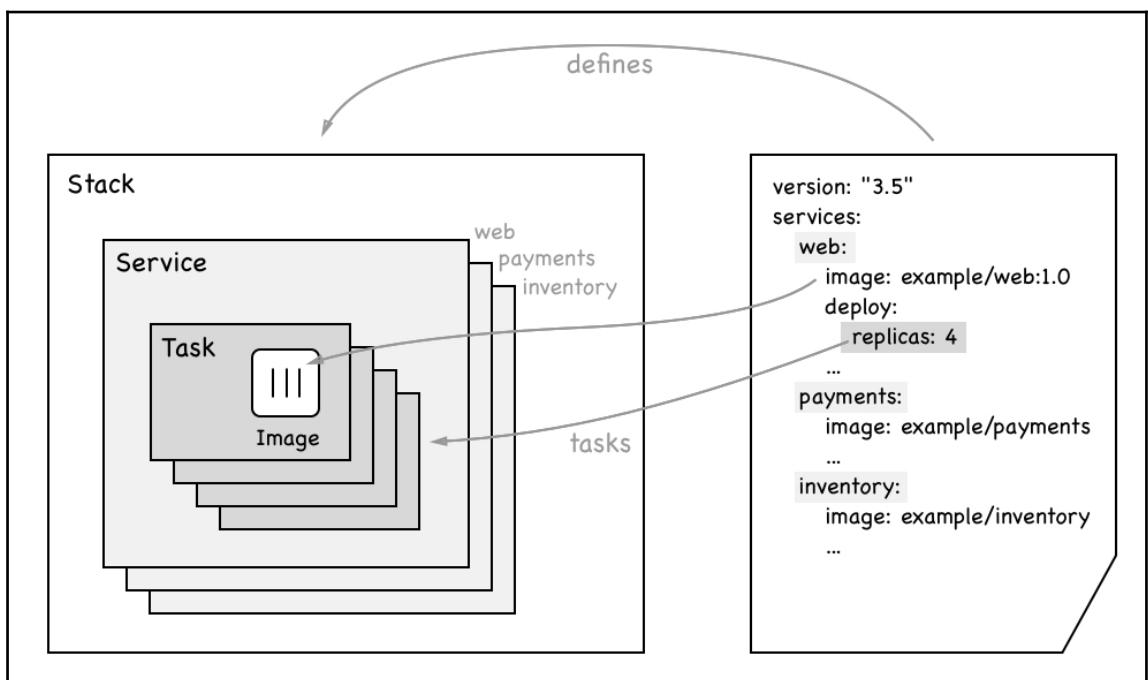
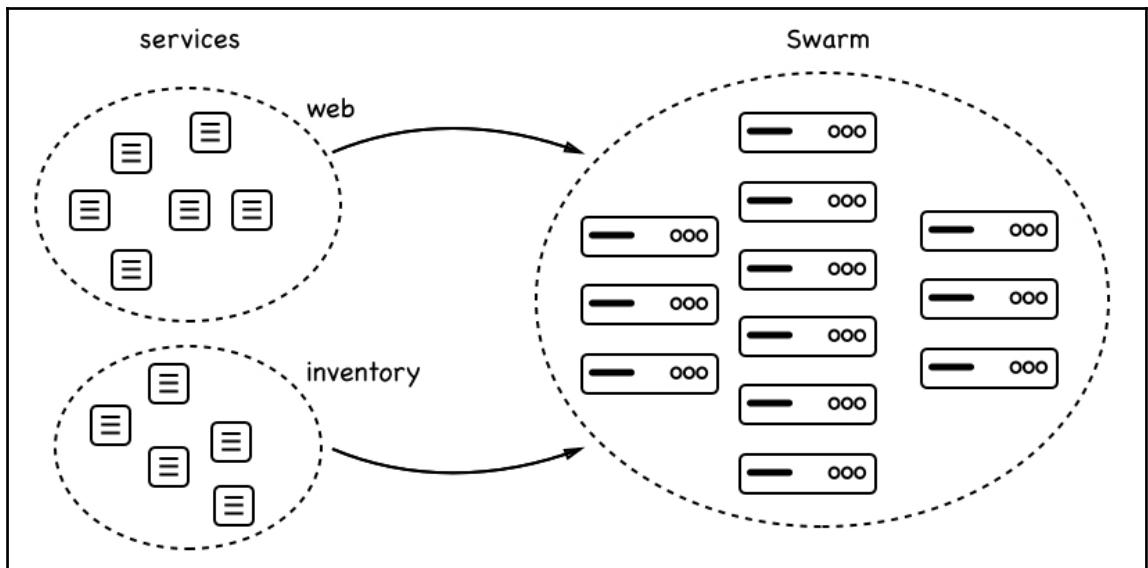


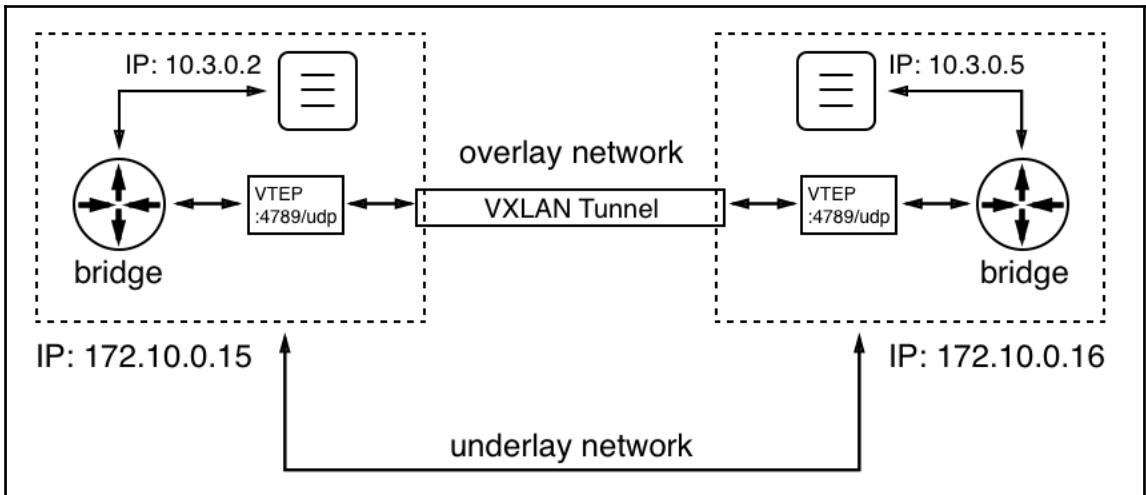
Chapter 10: Introduction to Docker Swarm











```
$ docker swarm init
Swarm initialized: current node (mc07c43kp8v8d4ofnl5i9skb2) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-1ynzcy7z2tze0zhrbw7h855biywspmg9mjewknn5hwg6g10b5m-7h98rot6dfi5723ftkitsb1vt 192.168.65.3:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS	ENGINE VERSION
mc07c43kp8v8d4ofnl5i9skb2 *	linuxkit-025000000001	Ready	Active	Leader	18.03.0-ce-rc1

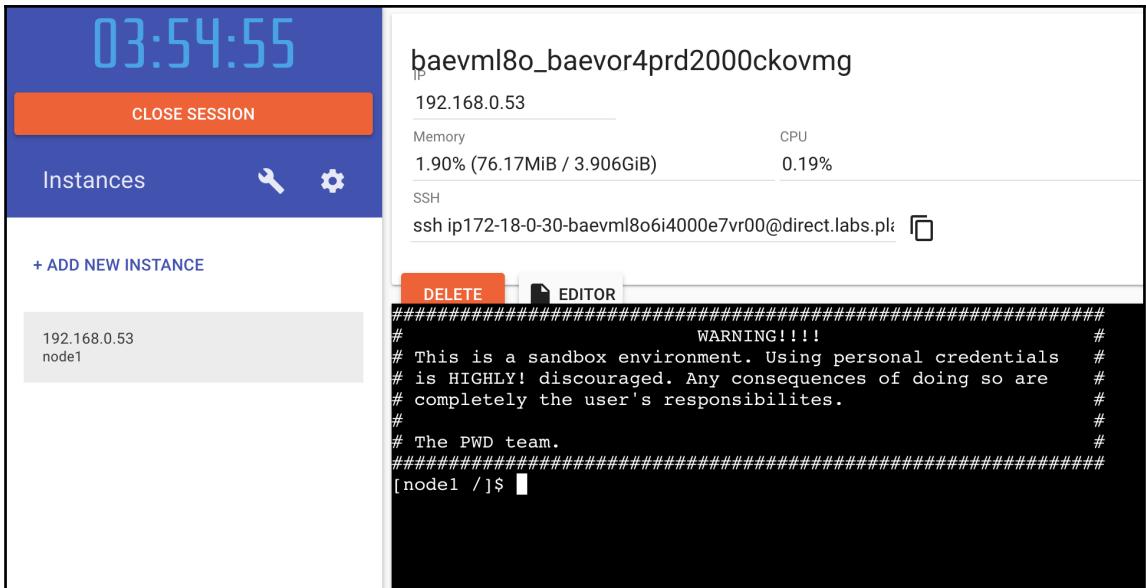
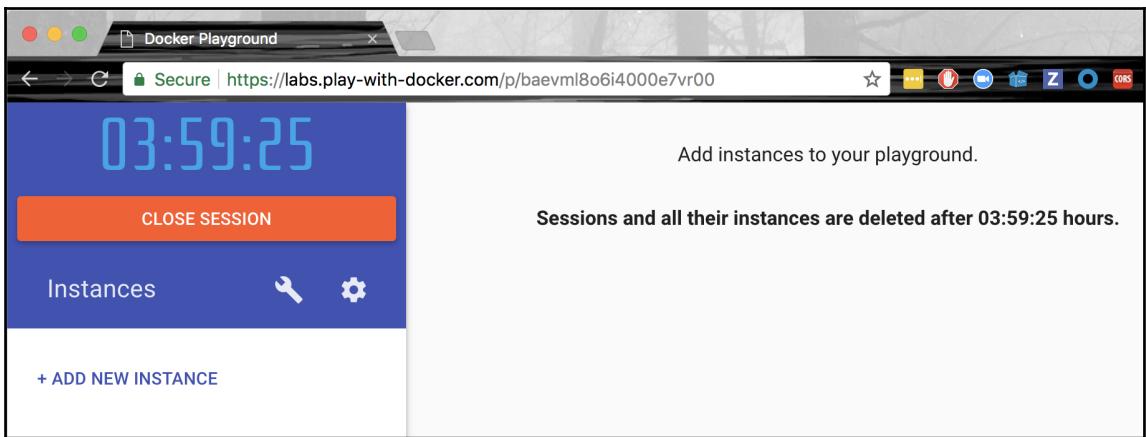
```
$ docker node inspect mc07c43kp8v8d4ofnl5i9skb2
[
  {
    "ID": "mc07c43kp8v8d4ofnl5i9skb2",
    "Version": {
      "Index": 9
    },
    "CreatedAt": "2018-03-06T01:48:57.62500232Z",
    "UpdatedAt": "2018-03-06T01:48:58.235847341Z",
    "Spec": {
      "Labels": {},
      "Role": "manager",
      "Availability": "active"
    },
    "Description": {
      "Hostname": "linuxkit-025000000001",
      "Platform": {
        "Architecture": "x86_64",
        "OS": "linux"
      },
      "Resources": {
        "NanoCPUs": 4000000000,
        "MemoryBytes": 2095788032
      },
      "Engine": {
        "EngineVersion": "18.03.0-ce-rc1",
        "Plugins": [
          {
            "Type": "Log",
            "Name": "awslogs"
          },
          {
            "Type": "Log",
            "Name": "awslogs"
          }
        ]
      }
    }
  }
]
```

```
$ docker-machine ls
NAME      ACTIVE   DRIVER      STATE      URL
default    -        virtualbox  Running    tcp://192.168.99.100:2376
$
```

```
$ docker-machine create --driver virtualbox default
Running pre-create checks...
Creating machine...
(default) Copying /Users/gabriel/.docker/machine/cache/boot2docker.iso to /Users/gabriel/.docker/machines/default/
(default) Creating VirtualBox VM...
(default) Creating SSH key...
(default) Starting the VM...
(default) Check network to re-create if needed...
(default) Waiting for an IP...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with boot2docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run
$
```

```
$ docker-machine ls
NAME      ACTIVE   DRIVER      STATE      URL
default    -        virtualbox  Running    tcp://192.168.99.100:2376
node-1     -        virtualbox  Running    tcp://192.168.99.101:2376
node-2     -        virtualbox  Running    tcp://192.168.99.102:2376
node-3     -        virtualbox  Running    tcp://192.168.99.103:2376
node-4     -        virtualbox  Running    tcp://192.168.99.104:2376
node-5     -        virtualbox  Running    tcp://192.168.99.105:2376
$
```

```
$ docker-machine ssh node-1 docker node ls
ID          HOSTNAME      STATUS      AVAILABILITY      MANAGER STATUS
kgvj80vupkw9ucdkxw853dejt *  node-1      Ready      Active      Leader
m2bel1iye6szjqs5nghfpcmzv   node-2      Ready      Active      Reachable
ij0yjt1rd7mzr4jq5mn9fwyky   node-3      Ready      Active      Reachable
ys3cg84p1fu6krz4pskebhcg7   node-4      Ready      Active
esm46efplk769rel3q2tebz8b   node-5      Ready      Active
$
```



The screenshot shows a Docker session interface. On the left, there's a sidebar with a clock at 03:37:36, a "CLOSE SESSION" button, and a "Instances" section listing several nodes:

- 192.168.0.53 node1
- 192.168.0.52 node2
- 192.168.0.51 node3** (highlighted)
- 192.168.0.50 node4
- 192.168.0.49 node5

On the right, a terminal window displays the output of a "docker swarm join" command on node3:

```
#####
#           WARNING!!!!
# This is a sandbox environment. Using personal credentials
# IS HIGHLY! discouraged. Any consequences of doing so are
# completely the user's responsibilites.
#
# The PWD team.
#####
[node3 /]$ docker swarm join --token SWMTKN-1-67c59yj4dbawmyn8mdbz7
263rprxm7u9a6k7527cy8unf6khjx-36ubstajb7ly9754af8mtvllg 192.168.0.5
3:2377
This node joined a swarm as a worker.
[node3 /]$
```

ID	HOSTNAME	STATUS	AVAILABILITY	MANAGER STATUS
8mwcohdn2khf936s0xadkjlc	node4	Ready	Active	
fmr85luxuoryn6i5kgvqqsgqi	node5	Ready	Active	
gtwxqt6j6l4khlyixvfsckrra	node3	Ready	Active	
n0xrvrq2wjbdbgm2nsbjqka06	node2	Ready	Active	
rkmkkjt90j4xd9y3m4xyhk32f *	node1	Ready	Active	Leader

```
→ ~ docker-machine create --driver amazonec2 aws-node-1
Running pre-create checks...
Creating machine...
(~aws-node-1) Launching instance...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with ubuntu(systemd)...
Installing Docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: docker-machine env aws-node-1
→ ~
```

```
↳ ~ export | grep DOCKER
DOCKER_CERT_PATH=/Users/gabriel/.docker/machine/machines/aws-node-1
DOCKER_HOST=tcp://35.172.240.127:2376
DOCKER_MACHINE_NAME=aws-node-1
DOCKER_TLS_VERIFY=1
↳ ~
```

```
↳ ~ curl -4 35.172.240.127:8000
curl: (7) Failed to connect to 35.172.240.127 port 8000: Operation timed out
↳ ~
```

The screenshot shows the AWS EC2 Instances page. On the left sidebar, under the 'INSTANCES' section, the 'Instances' link is highlighted with a red box. In the main content area, a search bar at the top contains the text 'search : aws-node-1'. Below the search bar, a table lists one instance: 'aws-node-1' (Instance ID: i-0d4d1eb363211f1e4, Instance Type: t2.micro, Availability Zone: us-east-1a). To the right of the table, detailed information for the instance is shown, including its elastic IP (us-east-1a), security group (docker-machine, with a link to view inbound rules), scheduled events (No scheduled events), AMI ID (ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170619.1 (ami-d15a75c7)), platform (-), and IAM role (-).

Screenshot of the AWS CloudFormation console showing the creation of a new security group named "sg-c14f4db3". The security group is associated with a VPC ID "vpc-f3723f96" and is described as "Docker Machine". The Inbound tab is selected, showing three rules: "All traffic" (Protocol All, Port Range All, Source 70.113.114.234/32, Description personal access), "SSH" (Protocol TCP, Port Range 22, Source 0.0.0.0/0), and "Custom TCP Rule" (Protocol TCP, Port Range 2376, Source 0.0.0.0/0). The "All traffic" rule is highlighted with a red box.

Type	Protocol	Port Range	Source	Description
All traffic	All	All	70.113.114.234/32	personal access
All traffic	All	All	sg-c14f4db3 (docker-machine)	intra swarm commun...
SSH	TCP	22	0.0.0.0/0	
Custom TCP Rule	TCP	2376	0.0.0.0/0	

```
➜ ~ docker-machine ls
NAME      ACTIVE   DRIVER      STATE      URL
aws-node-1 *      amazonec2  Running    tcp://35.172.240.127:2376
aws-node-2 -       amazonec2  Running    tcp://54.236.40.1:2376
aws-node-3 -       amazonec2  Running    tcp://34.205.171.56:2376
aws-node-4 -       amazonec2  Running    tcp://34.239.93.22:2376
aws-node-5 -       amazonec2  Running    tcp://52.205.26.218:2376
node-1     -       virtualbox  Running    tcp://192.168.99.100:2376
node-2     -       virtualbox  Running    tcp://192.168.99.101:2376
node-3     -       virtualbox  Running    tcp://192.168.99.102:2376
node-4     -       virtualbox  Running    tcp://192.168.99.103:2376
node-5     -       virtualbox  Running    tcp://192.168.99.104:2376

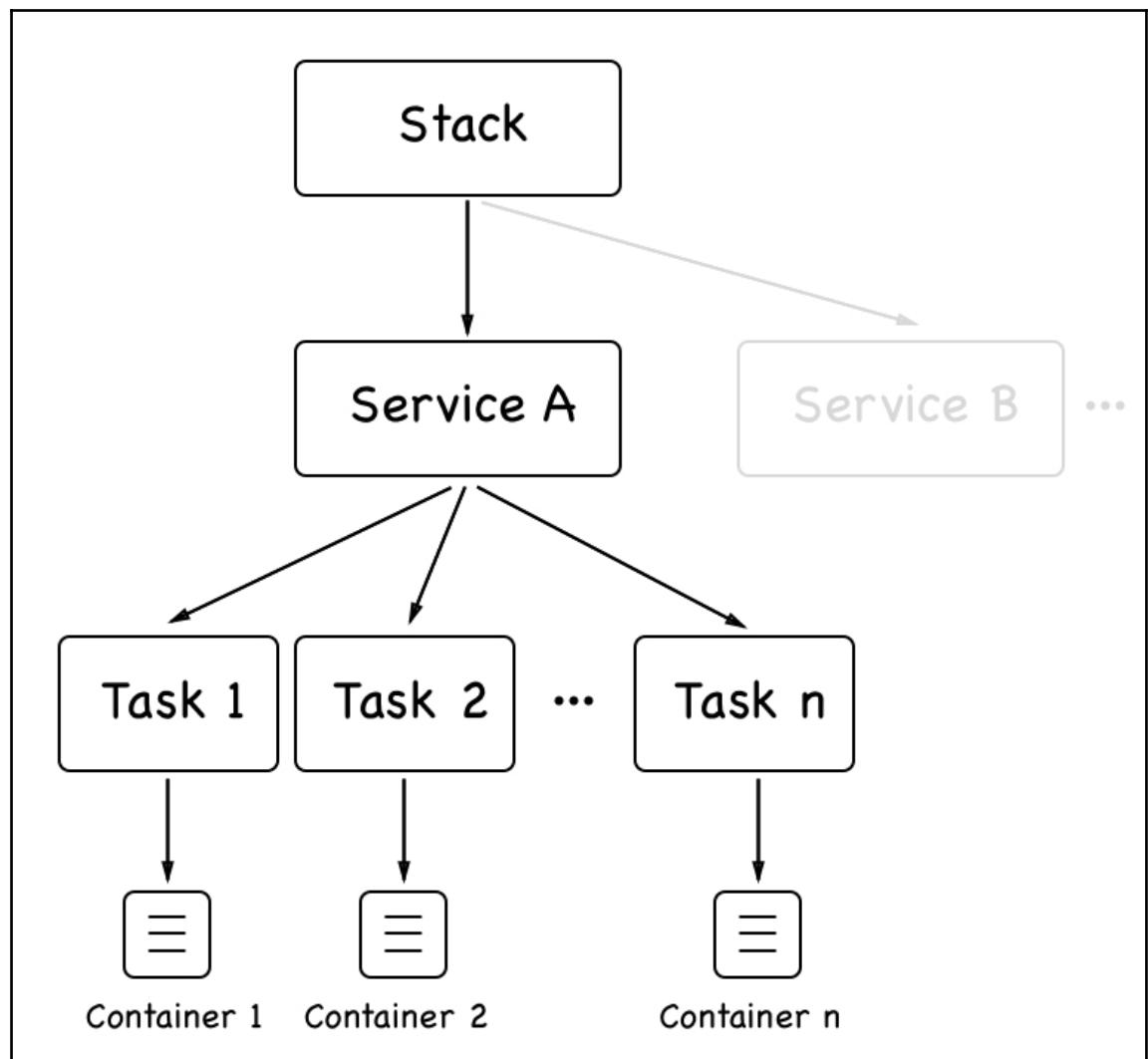
```

```
$ docker node ls
ID          HOSTNAME   STATUS      AVAILABILITY  MANAGER STATUS  ENGINE VERSION
jcjv5id54tp6laevnvjdqk1t6 *  aws-node-1  Ready      Active        Leader        18.04.0-ce
i9y6iiyc1v9o23x86i5qfyh2a  aws-node-2  Ready      Active        Reachable    18.04.0-ce
zetk7ntzqyf75mk8n39y2o72v  aws-node-3  Ready      Active        Reachable    18.04.0-ce
h38drhf7fg50h2n4tc1vc49ko  aws-node-4  Ready      Active        Active       18.04.0-ce
ofbxw724dvvu84dak974s64ocv  aws-node-5  Ready      Active        Active       18.04.0-ce
$
```

```
$ docker service ls
ID          NAME      MODE      REPLICAS      IMAGE      PORTS
lqyymm2u2mry  sample-stack_whoami  replicated  6/6        training/whoami:latest  *:81->8000/tcp
$
```

```
docker@node-1:~$ docker service ps sample-stack_whoami
ID          NAME      IMAGE      NODE      DESIRED STATE  CURRENT STATE      ERROR      PORTS
mtvunqieacg  sample-stack_whoami.1  training/whoami:latest  node-5    Running   Running 26 seconds ago
n21e7ktyvo4b  sample-stack_whoami.2  training/whoami:latest  node-1    Running   Running 26 seconds ago
lozzitfydlad  sample-stack_whoami.3  training/whoami:latest  node-2    Running   Running 27 seconds ago
xymlohw68639  sample-stack_whoami.4  training/whoami:latest  node-2    Running   Running 27 seconds ago
yn84l8fc83el  sample-stack_whoami.5  training/whoami:latest  node-3    Running   Running 28 seconds ago
3hv4qu10dzs  sample-stack_whoami.6  training/whoami:latest  node-4    Running   Running 27 seconds ago
docker@node-1:~$
```

```
docker@node-1:~$ docker container ls
CONTAINER ID      IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
d11a82869bf0  training/whoami:latest  "/app/http"  7 minutes ago  Up 7 minutes  8000/tcp      sample-stack_whoami.2.n21e7ktyvo4b2ufalk0aibzy
docker@node-1:~$
```



```
docker@node-1:~$ docker service logs sample-stack_whoami
sample-stack_whoami.2.n21e7ktyvo4b@node-1      | Listening on :8000
sample-stack_whoami.1.mtvvunqieacg@node-5      | Listening on :8000
sample-stack_whoami.6.3hv4ql0dzs@node-4        | Listening on :8000
sample-stack_whoami.4.xymlohw68639@node-2      | Listening on :8000
sample-stack_whoami.3.lozzitfydlad@node-2       | Listening on :8000
sample-stack_whoami.5.yn84l8fc83el@node-3      | Listening on :8000
docker@node-1:~$
```

```
docker@node-1:~$ docker service logs n21e7ktyvo4b
sample-stack_whoami.2.n21e7ktyvo4b@node-1      | Listening on :8000
docker@node-1:~$
```

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
mtvvunqieacg	sample-stack_whoami.1	training/whoami:latest	node-5	Running	Running 7 hours ago		
i8reh311h03z	sample-stack_whoami.2	training/whoami:latest	node-1	Running	Running 12 seconds ago		
n21e7ktyvo4b	\ sample-stack_whoami.2	training/whoami:latest	node-1	Shutdown	Failed 17 seconds ago	"task: non-zero exit (137)"	
lozzitfydlad	sample-stack_whoami.3	training/whoami:latest	node-2	Running	Running 7 hours ago		
xymlohw68639	sample-stack_whoami.4	training/whoami:latest	node-2	Running	Running 7 hours ago		
yn84l8fc83el	sample-stack_whoami.5	training/whoami:latest	node-3	Running	Running 7 hours ago		
3hv4ql0dzs	sample-stack_whoami.6	training/whoami:latest	node-4	Running	Running 7 hours ago		

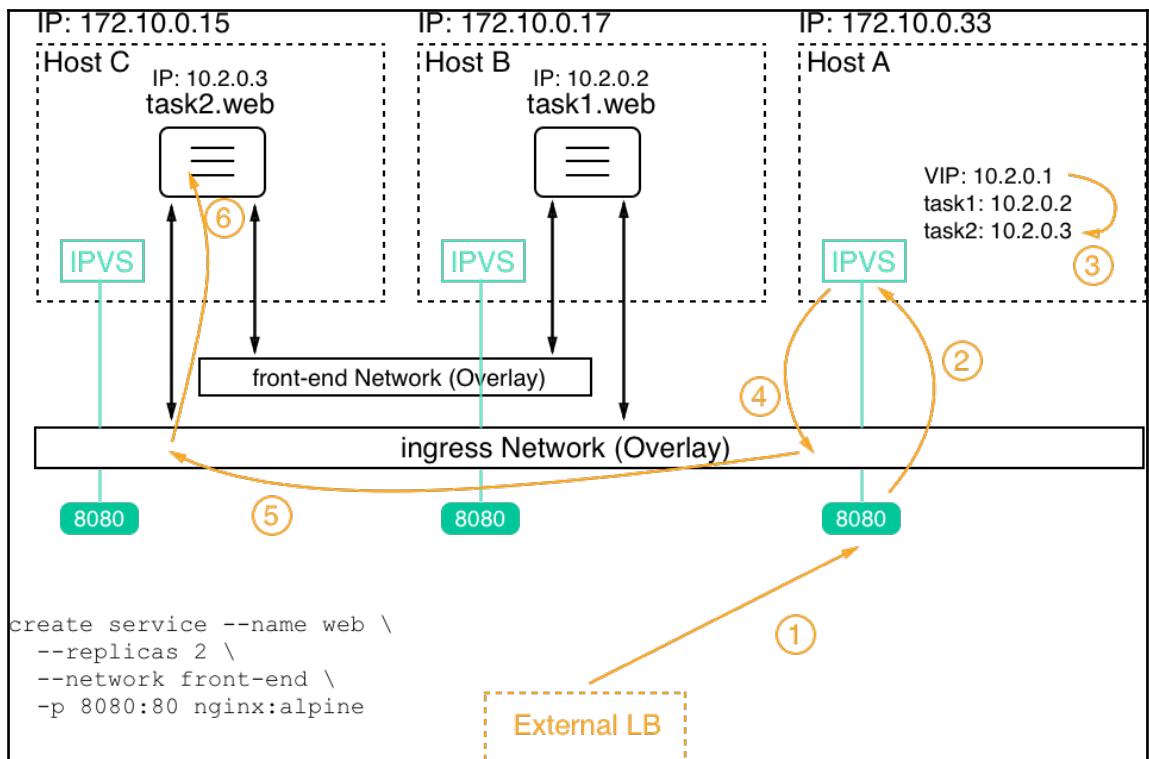
ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
mtvvunqieacg	sample-stack_whoami.1	training/whoami:latest	node-5	Running	Running 7 hours ago		
i8reh311h03z	sample-stack_whoami.2	training/whoami:latest	node-1	Running	Running 11 minutes ago		
n21e7ktyvo4b	\ sample-stack_whoami.2	training/whoami:latest	node-1	Shutdown	Failed 11 minutes ago	"task: non-zero exit (137)"	
so43kd4d03h5	sample-stack_whoami.3	training/whoami:latest	node-1	Ready	Ready 1 second ago		
lozzitfydlad	\ sample-stack_whoami.3	training/whoami:latest	node-2	Shutdown	Running 9 seconds ago		
jhlolhnkgzt7	sample-stack_whoami.4	training/whoami:latest	node-3	Ready	Ready 1 second ago		
xymlohw68639	\ sample-stack_whoami.4	training/whoami:latest	node-2	Shutdown	Running 9 seconds ago		
yn84l8fc83el	sample-stack_whoami.5	training/whoami:latest	node-3	Running	Running 7 hours ago		
3hv4ql0dzs	sample-stack_whoami.6	training/whoami:latest	node-4	Running	Running 7 hours ago		

```
docker@node-1:~$ docker stack rm sample-stack
Removing service sample-stack_whoami
Removing network sample-stack_test-net
docker@node-1:~$
```

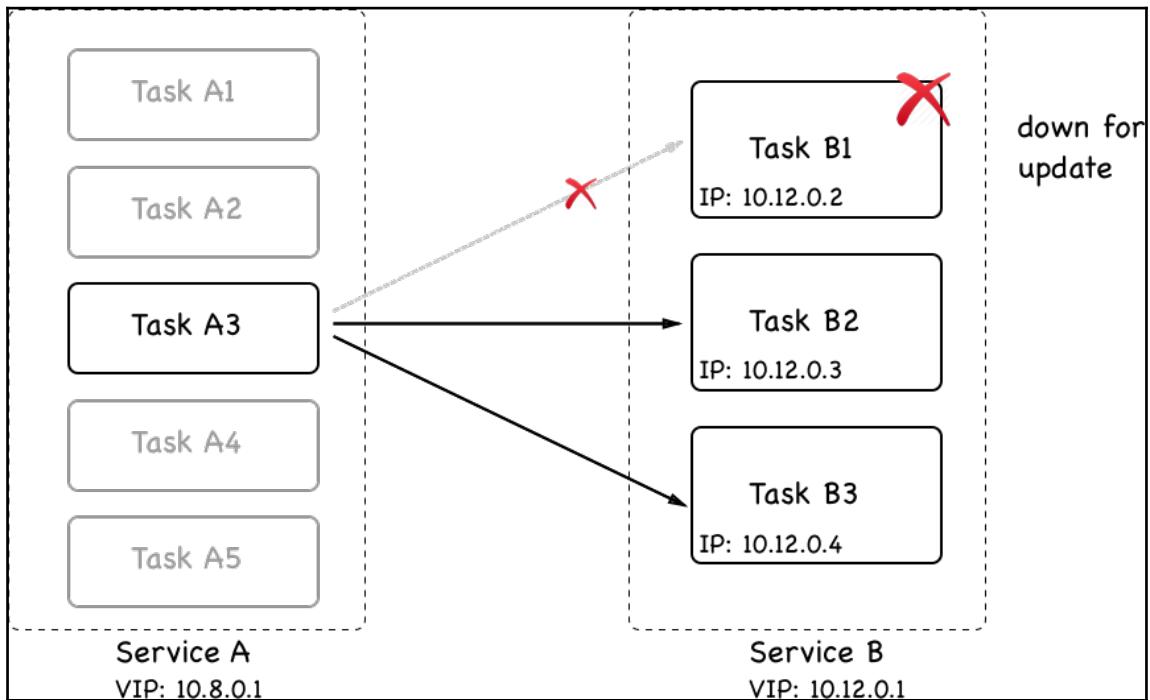
```
docker@node-1:~$ docker stack deploy -c pet-stack.yaml pets
Creating network pets_pets-net
Creating service pets_db
Creating service pets_web
docker@node-1:~$ 4
```

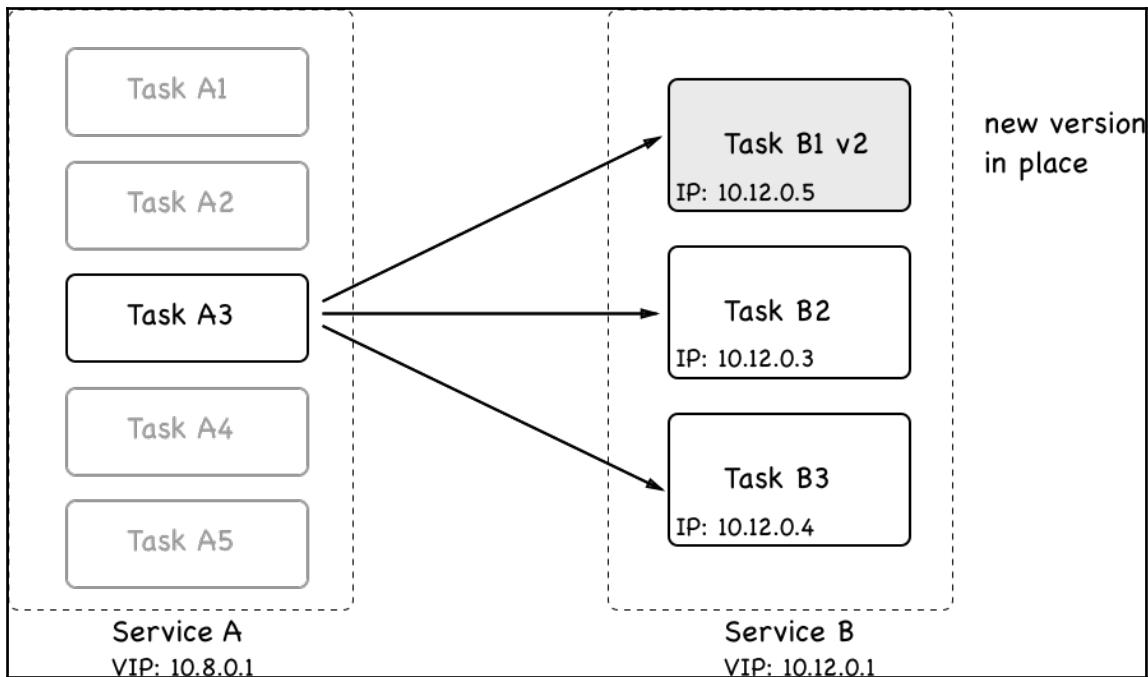
ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR
08f14cawm4if	pets_db.1	fundamentalsofdocker/ch08-db:1.0	node-2	Running	Running 2 seconds ago	
qbhefgj97ijp	pets_web.1	fundamentalsofdocker/ch08-web:1.0	node-4	Running	Running 8 seconds ago	
o5fb1zf86588	pets_web.2	fundamentalsofdocker/ch08-web:1.0	node-3	Running	Running 8 seconds ago	
nfrez66ssvix	pets_web.3	fundamentalsofdocker/ch08-web:1.0	node-5	Running	Running 8 seconds ago	

```
docker@node-1:~$ curl localhost:3000/pet
<html>
<head>
    <link rel="stylesheet" href="main.css">
</head>
<body>
    <div class="container">
        <h4>Cat Gif of the day</h4>
        <small>Courtesy: <a href="http://www.buzzfeed.com/copyranter/the-best-cat-gif-post-"
        <p>Delivered to you by container c9aa9dacd9b2<p>
    </div>
</body>
</html>docker@node-1:~$
```



Chapter 11: Zero Downtime Deployments and Secrets





```
$ docker stack deploy -c stack.yaml web
Creating network web_default
Creating service web_web
$
```

Every 2.0s: docker stack ps web							
ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
ze29yvu4jyy	web_web.1	nginx:1.12-alpine	node-2	Running	Running 3 minutes ago		
i1cv5v4o9ld3	web_web.2	nginx:1.12-alpine	node-2	Running	Running 3 minutes ago		
kzpq1cub4q49	web_web.3	nginx:1.12-alpine	node-1	Running	Running 3 minutes ago		
ynt8n4ke8y1d	web_web.4	nginx:1.12-alpine	node-3	Running	Running 3 minutes ago		
qa18xv1u9v1d	web_web.5	nginx:1.12-alpine	node-5	Running	Running 3 minutes ago		
5in9mkx1pkv	web_web.6	nginx:1.12-alpine	node-4	Running	Running 3 minutes ago		
iyjntpgy6cwe	web_web.7	nginx:1.12-alpine	node-1	Running	Running 3 minutes ago		
q230v16rlwr	web_web.8	nginx:1.12-alpine	node-5	Running	Running 3 minutes ago		
nh6jm2fyzwre	web_web.9	nginx:1.12-alpine	node-3	Running	Running 3 minutes ago		
iuu56iot6dxm	web_web.10	nginx:1.12-alpine	node-4	Running	Running 3 minutes ago		

overall progress: 4 out of 10 tasks

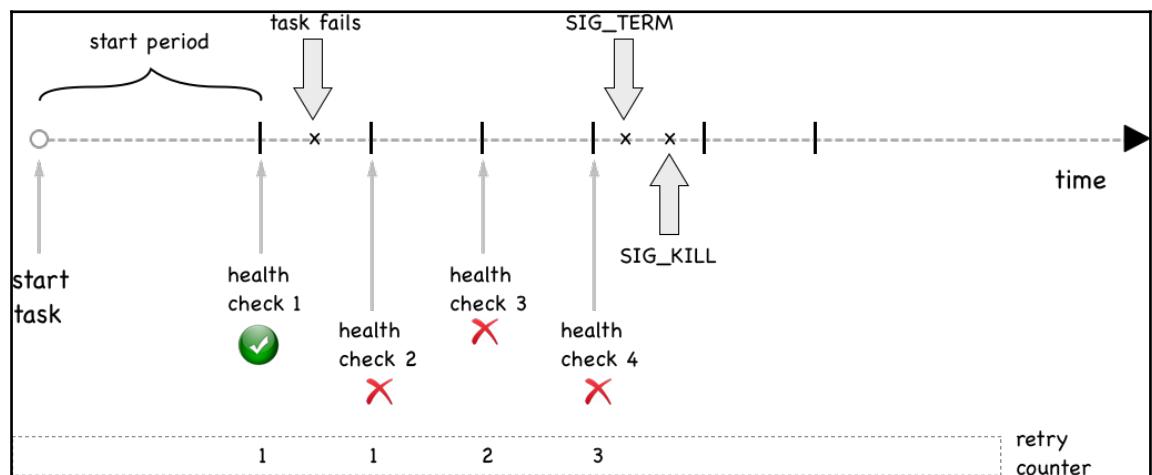
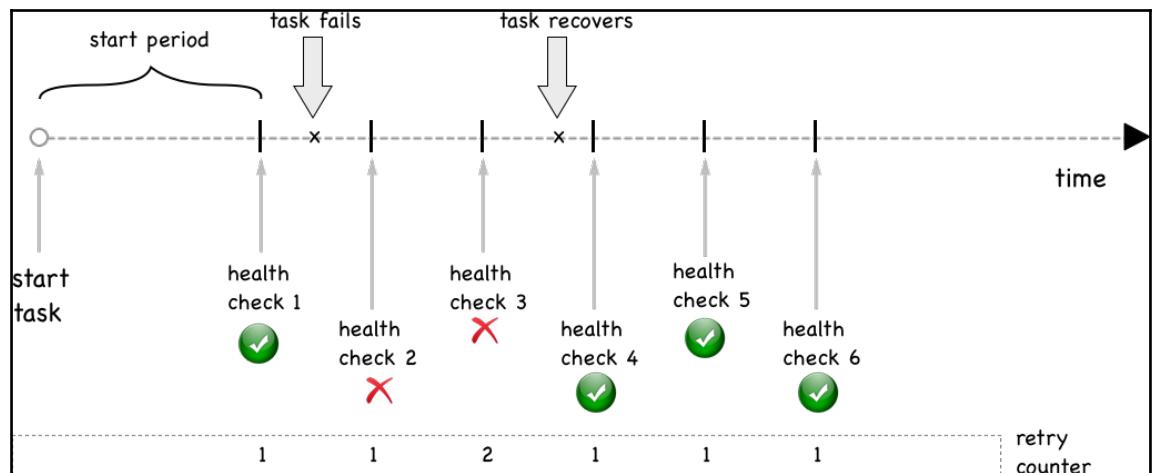
1/10: running [=====>]
2/10: running [=====>]
3/10: running [=====>]
4/10: running [=====>]
5/10: preparing [=====>]
6/10: preparing [=====>]
7/10:
8/10:
9/10:
10/10:

Every 2.0s: docker stack ps web

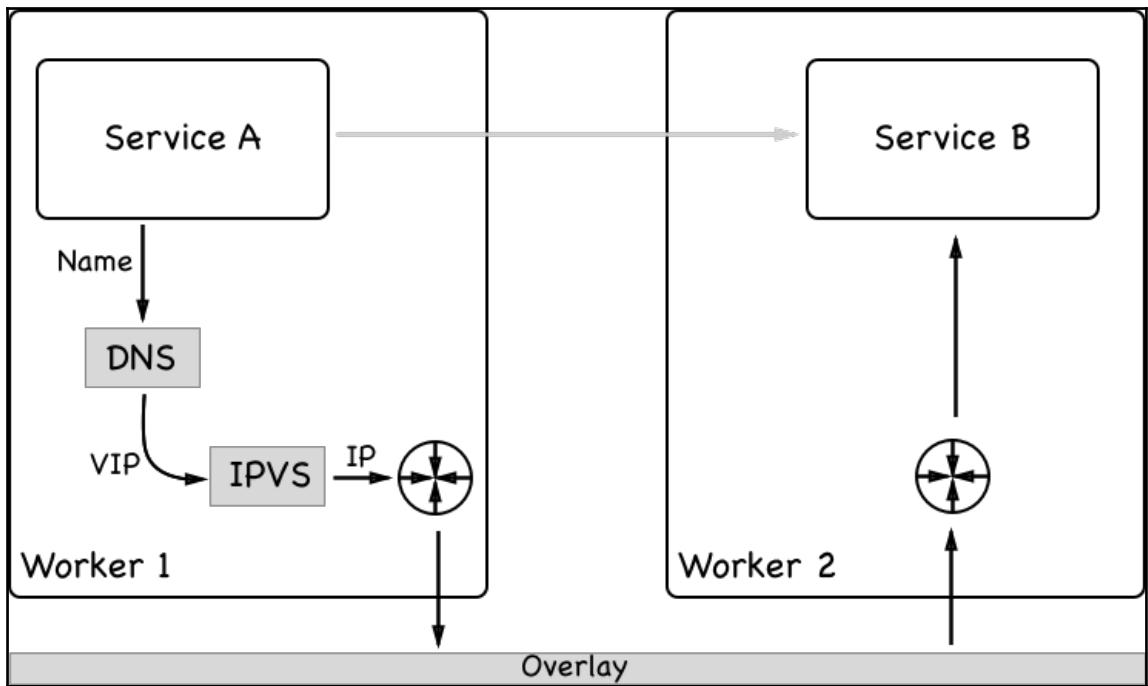
ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
ze29yvu4jyy	web_web.1	nginx:1.12-alpine	node-2	Running	Running 7 minutes ago		
i1cy5v4o9ld3	web_web.2	nginx:1.12-alpine	node-2	Running	Running 7 minutes ago		
kzqylcub4a9	web_web.3	nginx:1.12-alpine	node-1	Running	Running 7 minutes ago		
ynt8n4ke8yld	web_web.4	nginx:1.12-alpine	node-3	Running	Running 7 minutes ago		
qa18xviu9v1d	web_web.5	nginx:1.12-alpine	node-5	Running	Running 7 minutes ago		
5in9mkxlpkv	web_web.6	nginx:1.12-alpine	node-4	Running	Running 7 minutes ago		
iyintppg6cwe	web_web.7	nginx:1.12-alpine	node-1	Running	Running 7 minutes ago		
goxtziaseswi	web_web.8	nginx:1.13-alpine	node-5	Running	Running 7 seconds ago		
q230v16rlwrw	__ web_web.8	nginx:1.12-alpine	node-5	Shutdown	Shutdown 8 seconds ago		
y3q1e0mk20wn	web_web.9	nginx:1.13-alpine	node-3	Running	Running 9 seconds ago		
nh6jm2fyzwre	__ web_web.9	nginx:1.12-alpine	node-3	Shutdown	Shutdown 10 seconds ago		
iuu56iot6dxm	web_web.10	nginx:1.12-alpine	node-4	Running	Running 7 minutes ago		

Every 2.0s: docker stack ps web

ID	NAME	IMAGE	NODE	DESIRED STATE	CURRENT STATE	ERROR	PORTS
v99yet3urtyf	web_web.1	nginx:1.13-alpine	node-2	Running	Running 2 minutes ago		
ze29yvu4jyy	__ web_web.1	nginx:1.12-alpine	node-2	Shutdown	Shutdown 2 minutes ago		
s6was0b36hoa	web_web.2	nginx:1.13-alpine	node-2	Running	Running 2 minutes ago		
i1cy5v4o9ld3	__ web_web.2	nginx:1.12-alpine	node-2	Shutdown	Shutdown 2 minutes ago		
m2Fvxq7yxqfc	web_web.3	nginx:1.13-alpine	node-1	Running	Running 2 minutes ago		
kzqylcub4a9	__ web_web.3	nginx:1.12-alpine	node-1	Shutdown	Shutdown 2 minutes ago		
xsjjudndb7jm	web_web.4	nginx:1.13-alpine	node-3	Running	Running 2 minutes ago		
ynt8n4ke8yld	__ web_web.4	nginx:1.12-alpine	node-3	Shutdown	Shutdown 2 minutes ago		
fuk4xpb5g5un	web_web.5	nginx:1.13-alpine	node-5	Running	Running about a minute ago		
qa18xviu9v1d	__ web_web.5	nginx:1.12-alpine	node-5	Shutdown	Shutdown about a minute ago		
ipa1scShe7d	web_web.6	nginx:1.13-alpine	node-4	Running	Running about a minute ago		
5in9mkxlpkv	__ web_web.6	nginx:1.12-alpine	node-4	Shutdown	Shutdown about a minute ago		
feyu3l2ufjgr	web_web.7	nginx:1.13-alpine	node-1	Running	Running 2 minutes ago		
iyintppg6cwe	__ web_web.7	nginx:1.12-alpine	node-1	Shutdown	Shutdown 2 minutes ago		
goxtziaseswi	web_web.8	nginx:1.13-alpine	node-5	Running	Running 2 minutes ago		
q230v16rlwrw	__ web_web.8	nginx:1.12-alpine	node-5	Shutdown	Shutdown 2 minutes ago		
y3q1e0mk20wn	web_web.9	nginx:1.13-alpine	node-3	Running	Running 2 minutes ago		
nh6jm2fyzwre	__ web_web.9	nginx:1.12-alpine	node-3	Shutdown	Shutdown 2 minutes ago		
7r93m02hnizg	web_web.10	nginx:1.13-alpine	node-4	Running	Running 2 minutes ago		
iuu56iot6dxm	__ web_web.10	nginx:1.12-alpine	node-4	Shutdown	Shutdown 2 minutes ago		



```
$ docker-machine ssh node-3 docker container ls
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
93c998264c10        nginx:alpine       "nginx -g 'daemon of..."   22 seconds ago    Up 21 seconds (healthy)   80/tcp
$
```



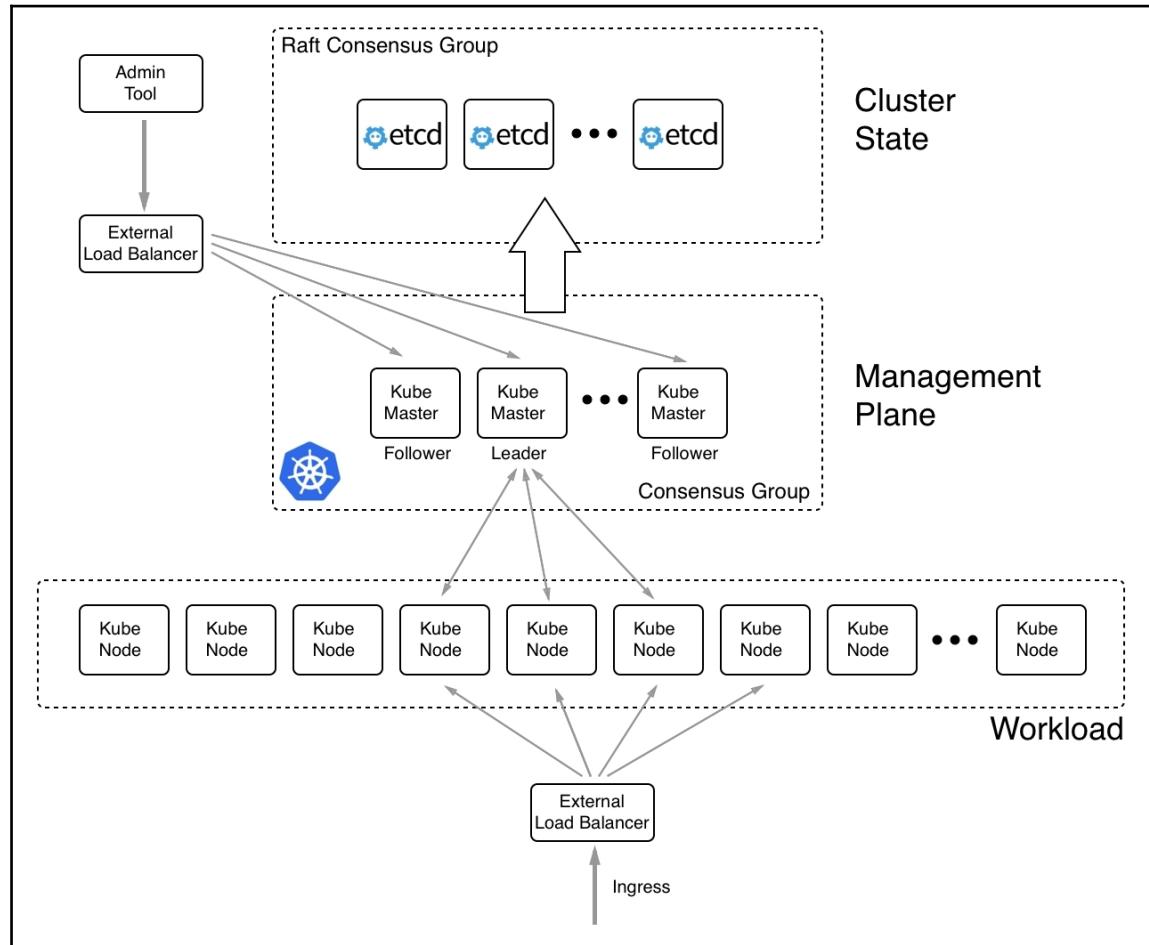
```
$ docker secret ls
ID          NAME      DRIVER      CREATED      UPDATED
axykb7msipit1g5so63ef02it  other-secret
puns1op5wr5hi21st5h3wj64  sample-secret
$
```

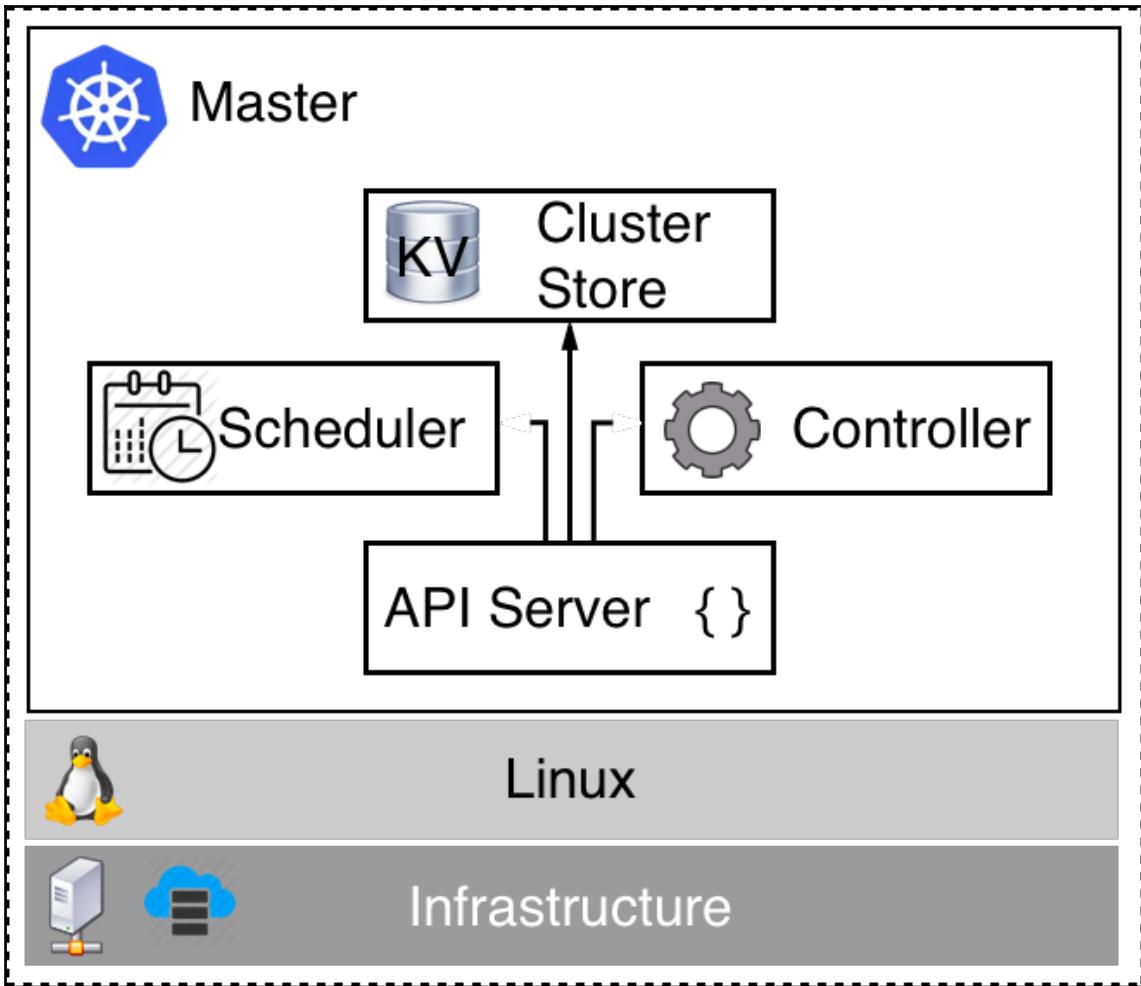
```
$ docker secret inspect other-secret
[
  {
    "ID": "axykb7msipit1g5so63ef02it",
    "Version": {
      "Index": 135
    },
    "CreatedAt": "2018-03-16T01:29:14.367872931Z",
    "UpdatedAt": "2018-03-16T01:29:14.367872931Z",
    "Spec": {
      "Name": "other-secret",
      "Labels": {}
    }
  }
]
$
```

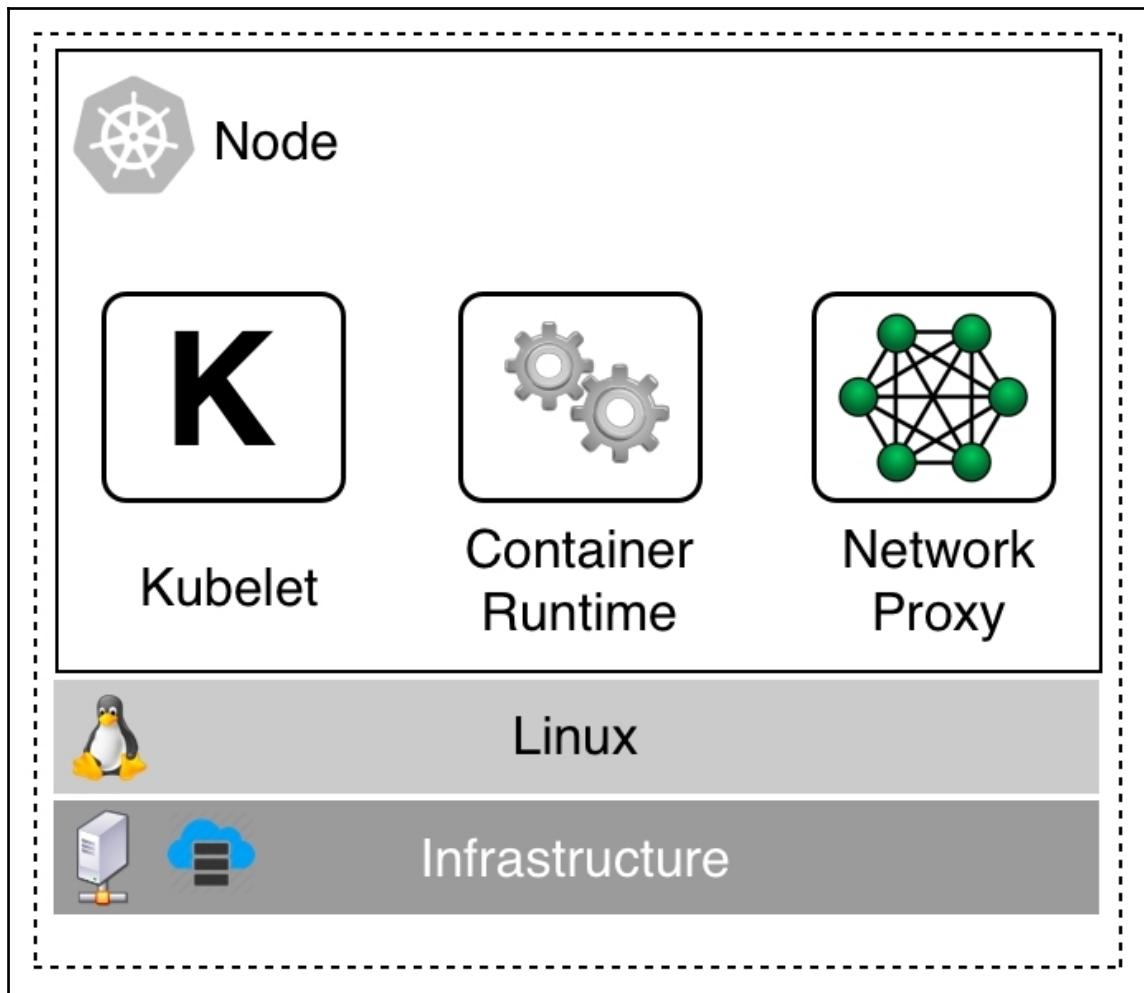
```
$ docker service create --name web \
>   --secret api-secret-key \
>   --publish 8000:8000 \
>   fundamentalsofdocker/whoami:latest
dzxxme8kmo0bg1r2ufwrhcztm
overall progress: 1 out of 1 tasks
1/1: running  [=====>]
verify: Service converged
$
```

```
docker@node-1:~$ docker container exec -it d5133b0e3eb3 cat /run/secrets/api-secret-key
my secret key
docker@node-1:~$
```

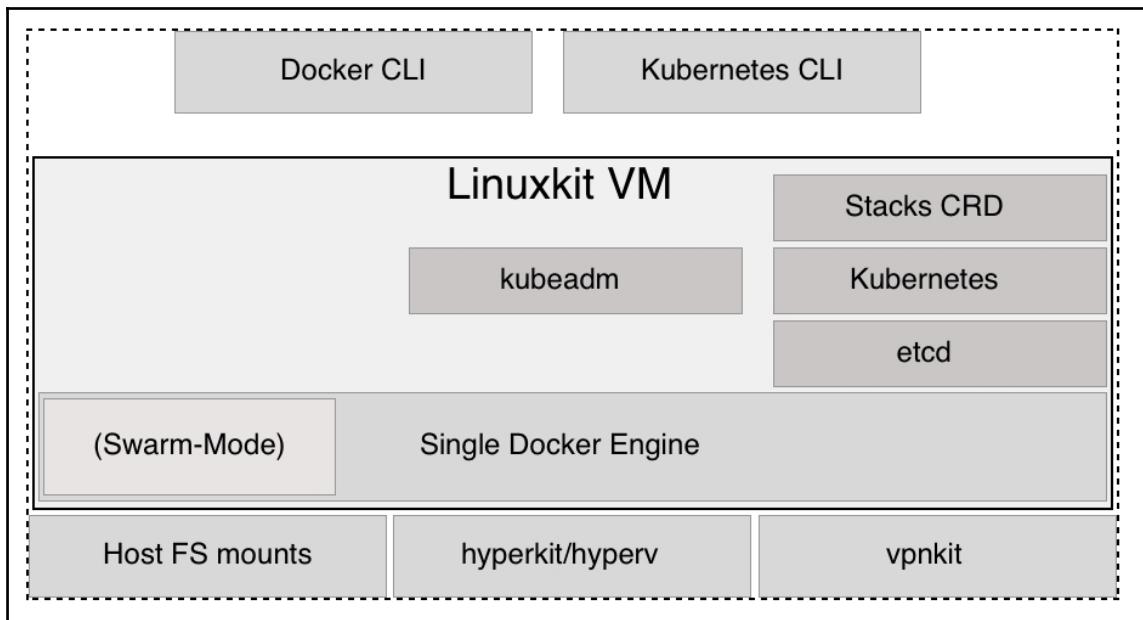
Chapter 12: Introduction to Kubernetes







```
$ kubectl get nodes
NAME      STATUS    ROLES   AGE      VERSION
minikube  Ready     <none>  2d      v1.9.0
$
```



Kubernetes

General File Sharing Disk Advanced Proxies Daemon Kubernetes Reset

Enable Kubernetes
Start a Kubernetes single-node cluster when starting Docker and set Kubernetes as Docker orchestrator.

Show system containers (advanced)
Show Kubernetes internal containers when using Docker commands.

Apply

● Docker is running

The initial Kubernetes cluster installation takes a few minutes and requires an Internet connection.

Install the Kubernetes cluster now?

Install Cancel

```
$ kubectl config get-contexts
CURRENT   NAME                 CLUSTER          AUTHINFO           NAMESPACE
          docker-for-desktop   docker-for-desktop-cluster
*         minikube            minikube          docker-for-desktop
$ [REDACTED]
```

```
$ kubectl config use-context docker-for-desktop
Switched to context "docker-for-desktop".
$ [REDACTED]
```

```
$ kubectl get nodes
NAME                  STATUS    ROLES      AGE      VERSION
docker-for-desktop   Ready     master     15m     v1.9.2
$ [REDACTED]
```

```
$ docker container ls --format "table {{.ID}}\t{{.Names}}"
CONTAINER ID        NAMES
0cddff5e5a86        k8s_compose_compose-5d4f4d67b6-gbrjh_docker_c3f07e06-2e3b-11e8-860f-025000000001_0
0edd323f6ccb        k8s_compose_compose-api-7bb7b5968f-f98jk_docker_c3e89385-2e3b-11e8-860f-025000000001_0
218514b4fc00        k8s_POD_compose-5d4f4d67b6-gbrjh_docker_c3f07e06-2e3b-11e8-860f-025000000001_0
af8ab4fc9f7e        k8s_POD_compose-api-7bb7b5968f-f98jk_docker_c3e89385-2e3b-11e8-860f-025000000001_0
f64fbcd5070c        k8s_sidecar_kube-dns-6f4fd4bdf-9b8kn_kube-system_ab66aab5-2e3b-11e8-860f-025000000001_0
4b6138bd34e7        k8s_dnsmasq_kube-dns-6f4fd4bdf-9b8kn_kube-system_ab66aab5-2e3b-11e8-860f-025000000001_0
bf1394d8e48a        k8s_kubedns_kube-dns-6f4fd4bdf-9b8kn_kube-system_ab66aab5-2e3b-11e8-860f-025000000001_0
a16b63a08f614       k8s_kube-proxy_kube-proxy-p4cf8_kube-system_ab6e9881-2e3b-11e8-860f-025000000001_0
655f8dc4a01c        k8s_POD_kube-proxy-p4cf8_kube-system_ab6e9881-2e3b-11e8-860f-025000000001_0
108b5a2fe05c        k8s_POD_kube-dns-6f4fd4bdf-9b8kn_kube-system_ab66aab5-2e3b-11e8-860f-025000000001_0
23f1808a6f8a        k8s_kube-scheduler_kube-scheduler-docker-for-desktop_kube-system_3a369b3ba7d6d3b6fa014295eab94925_0
89a1032beee7       k8s_kube-controller-manager_kube-controller-manager-docker-for-desktop_kube-system_b098d9f7b8b45512f23cb04fe3f64f5_0
bb25965301d8        k8s_etcd_etcd-docker-for-desktop_kube-system_7278f85057e8bf5cb81c9f96d3b25320_0
126d0edc29f0        k8s_kube-apiserver_kube-apiserver-docker-for-desktop_kube-system_8d19d05a3d7b137bafa35348cb849dd5_0
d87992c3ea6e        k8s_POD_kube-scheduler-docker-for-desktop_kube-system_3a369b3ba7d6d3b6fa014295eab94925_0
063fdf120ea5       k8s_POD_kube-controller-manager-docker-for-desktop_kube-system_b098d9f7b8b45512f23cb04fe3f64f5_0
22a1c70f6c4e        k8s_POD_kube-apiserver-docker-for-desktop_kube-system_8d19d05a3d7b137bafa35348cb849dd5_0
91ec502f1467       k8s_POD_etcd-docker-for-desktop_kube-system_7278f85057e8bf5cb81c9f96d3b25320_0
$ [REDACTED]
```

```
$ docker stack deploy -c docker-compose.yml app
Stack app was created
Waiting for the stack to be stable and running...
  - Service db has one container running
  - Service web has one container running
Stack app is stable and running
```

```
$ curl localhost:3000/pet
<html>
<head>
  <link rel="stylesheet" href="main.css">
</head>
<body>
  <div class="container">
    <h4>Cat Gif of the day</h4>
    </a></small>" alt="A cat gif" />
    <p>Delivered to you by container web-5c5964c9b8-b5jq9</p>
  </div>
</body>
$ █
```

```
$ kubectl get all
NAME      DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
deploy/web  1        1        1           1          9m

NAME            DESIRED  CURRENT  READY     AGE
rs/web-5c5964c9b8  1        1        1         9m

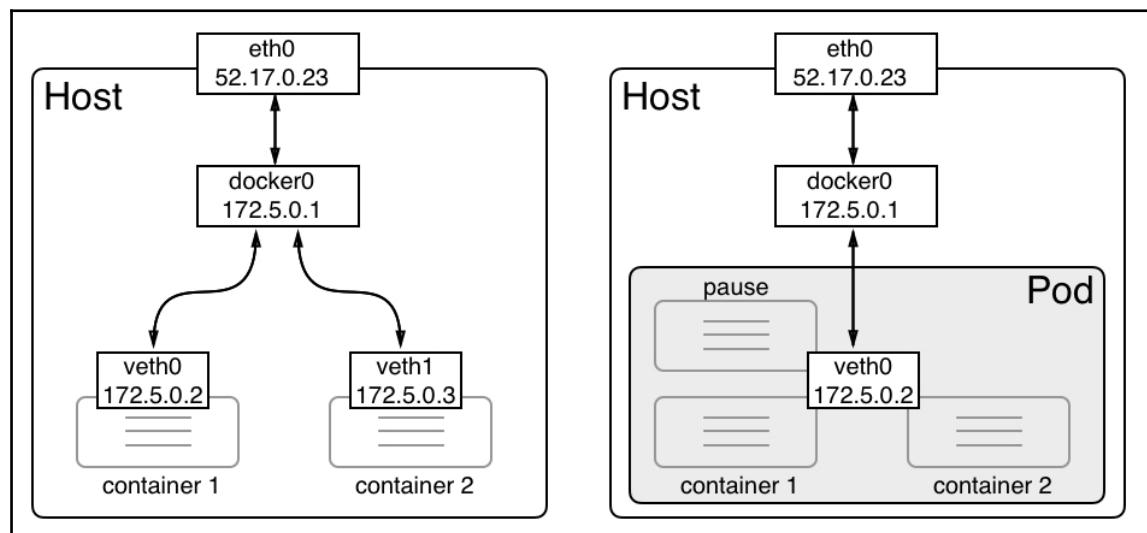
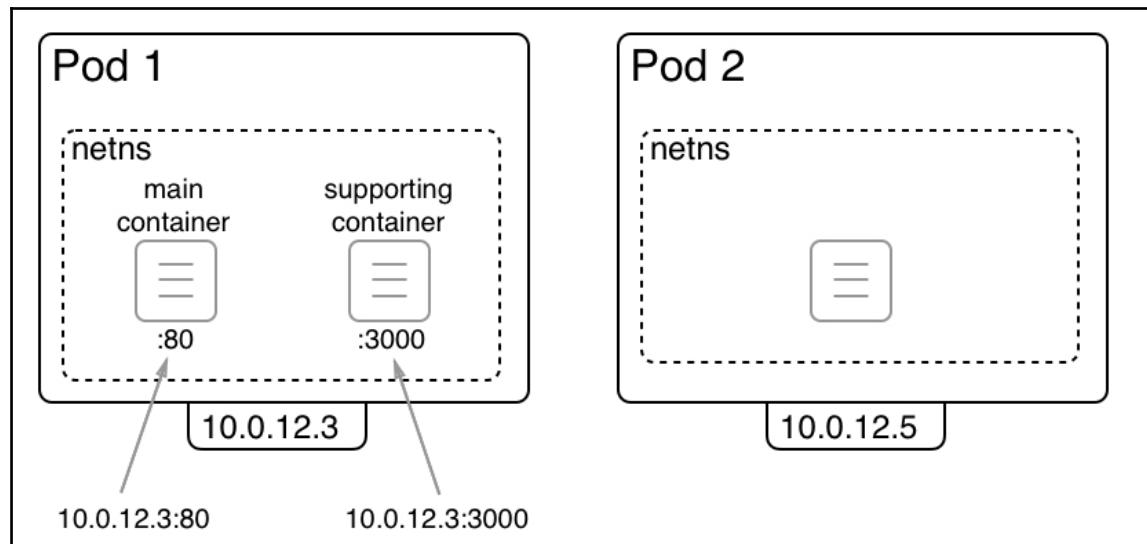
NAME      DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
deploy/web  1        1        1           1          9m

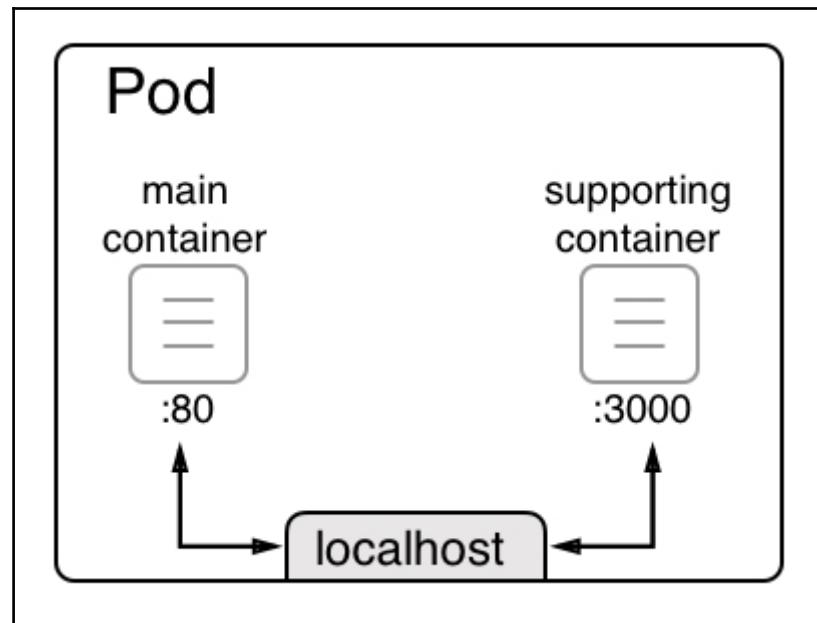
NAME            DESIRED  CURRENT  READY     AGE
rs/web-5c5964c9b8  1        1        1         9m

NAME      DESIRED  CURRENT  AGE
statefulsets/db  1        1        9m

NAME            READY   STATUS    RESTARTS  AGE
po/db-0          1/1     Running   0          9m
po/web-5c5964c9b8-b5jq9  1/1     Running   0          9m

NAME      TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
svc/db    ClusterIP   None           <none>        55555/TCP    9m
svc/kubernetes  ClusterIP   10.96.0.1      <none>        443/TCP     45m
svc/web    ClusterIP   None           <none>        55555/TCP    9m
svc/webPublished LoadBalancer  10.111.43.147  localhost    3000:32590/TCP  9m
$
```





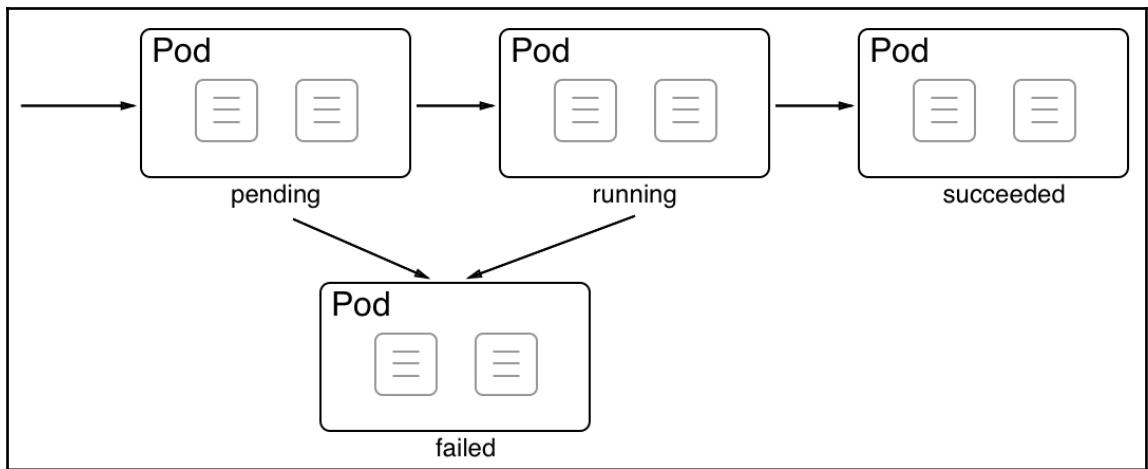
```
/ # wget -qO - localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
}
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
/ #
```

```
/ # ip a show eth0
11: eth0@if12: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
            valid_lft forever preferred_lft forever
/ #
```

```
$ docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "41909c08794041cab3a9d2e034426f2344f5310bd1cbfcbae65c5f25a05f541",
    "Created": "2018-03-26T22:16:44.790966007Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "172.17.0.0/16",
          "Gateway": "172.17.0.1"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {
      "8965ec65ca4a1de1f1d9c987b68e888c1115cf64f44ba3842953d29a2b9a0ea8": {
        "Name": "pause",
        "EndpointID": "890fc0527f7cb6484d24b7886772db23bb5a0502fe34269fc306277ea7a6f95e",
        "MacAddress": "02:42:ac:11:00:02",
        "IPv4Address": "172.17.0.2/16",
        "IPv6Address": ""
      }
    },
    "Options": {
      "com.docker.network.bridge.default_bridge": "true",
      "com.docker.network.bridge.enable_icc": "true",
      "com.docker.network.bridge.enable_ip_masquerade": "true",
      "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
      "com.docker.network.bridge.name": "docker0",
      "com.docker.network.driver.mtu": "1500"
    },
    "Labels": {}
  }
]
```



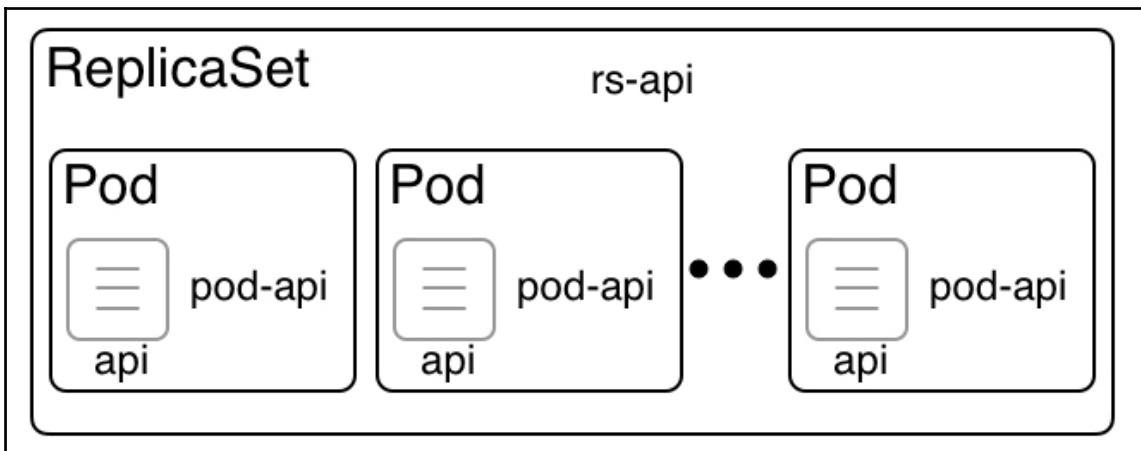
```

$ kubectl describe pod/web-pod
Name:           web-pod
Namespace:      default
Node:          minikube/192.168.99.105
Start Time:    Sun, 25 Mar 2018 22:47:49 -0500
Labels:         <none>
Annotations:   <none>
Status:        Running
IP:            172.17.0.3
Containers:
  web:
    Container ID:  docker://e8784dfc2e3fcf1de4fb9ab1508176799b6024b96d9447126e1db5dd5e2201f
    Image:          nginx:alpine
    Image ID:      docker-pullable://nginx@sha256:17c4704e19a11cd47545fa3c17e6903fc88672021f7f907f212d6663baf6ab57
    Port:          80/TCP
    State:         Running
      Started:    Sun, 25 Mar 2018 22:47:50 -0500
    Ready:         True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-fhdsm (ro)
Conditions:
  Type  Status
  Initialized  True
  Ready  True
  PodScheduled  True
Volumes:
  default-token-fhdsm:
    Type:     Secret (a volume populated by a Secret)
    SecretName: default-token-fhdsm
    Optional:  false
  QoS Class:  BestEffort
  Node-Selectors:  <none>
  Tolerations:  <none>
Events:
  Type  Reason          Age  From          Message
  ----  ----          --  --           --
  Normal Scheduled       5m  default-scheduler  Successfully assigned web-pod to minikube
  Normal SuccessfulMountVolume  5m  kubelet, minikube  MountVolume.SetUp succeeded for volume "default-token-fhdsm"
  Normal Pulled          5m  kubelet, minikube  Container image "nginx:alpine" already present on machine
  Normal Created          5m  kubelet, minikube  Created container
  Normal Started          5m  kubelet, minikube  Started container
$ 

```

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
my-data-claim	Bound	pvc-aac3bb2c-3224-11e8-a07f-080027c10823	2Gi	RWO	standard	14m

```
$ kubectl delete po/web-pod  
pod "web-pod" deleted  
$ kubectl create -f pod-with-vol.yaml  
pod "web-pod" created  
$ kubectl exec -it web-pod -- /bin/sh  
/ # cat /data/sample.txt  
Hello world!  
/ #
```



```
$ kubectl get pods  
NAME        READY   STATUS    RESTARTS   AGE  
rs-web-frj2m 1/1     Running   0          22h  
rs-web-q6cr7 1/1     Running   0          41s  
rs-web-zd2kt 1/1     Running   0          22h  
$
```

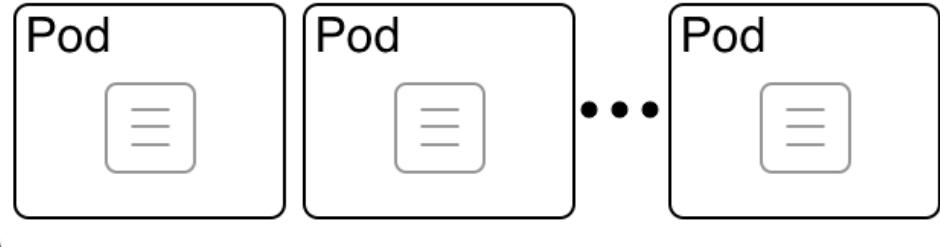
```
$ kubectl describe rs/rs-web
Name:           rs-web
Namespace:      default
Selector:       app=web
Labels:         app=web
Annotations:    <none>
Replicas:       3 current / 3 desired
Pods Status:   3 Running / 0 Waiting / 0 Succeeded / 0 Failed
Pod Template:
  Labels:  app=web
  Containers:
    nginx:
      Image:      nginx:alpine
      Port:       80/TCP
      Environment: <none>
      Mounts:     <none>
      Volumes:    <none>
Events:
  Type      Reason          Age   From            Message
  ----      ----          ----  ----            -----
  Normal   SuccessfulCreate  4m    replicaset-controller  Created pod: rs-web-q6cr7
```

Deployment

Updates and Rollback

ReplicaSet

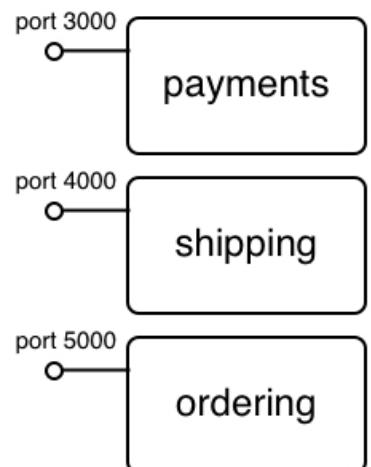
Self-healing, scalable, desired state

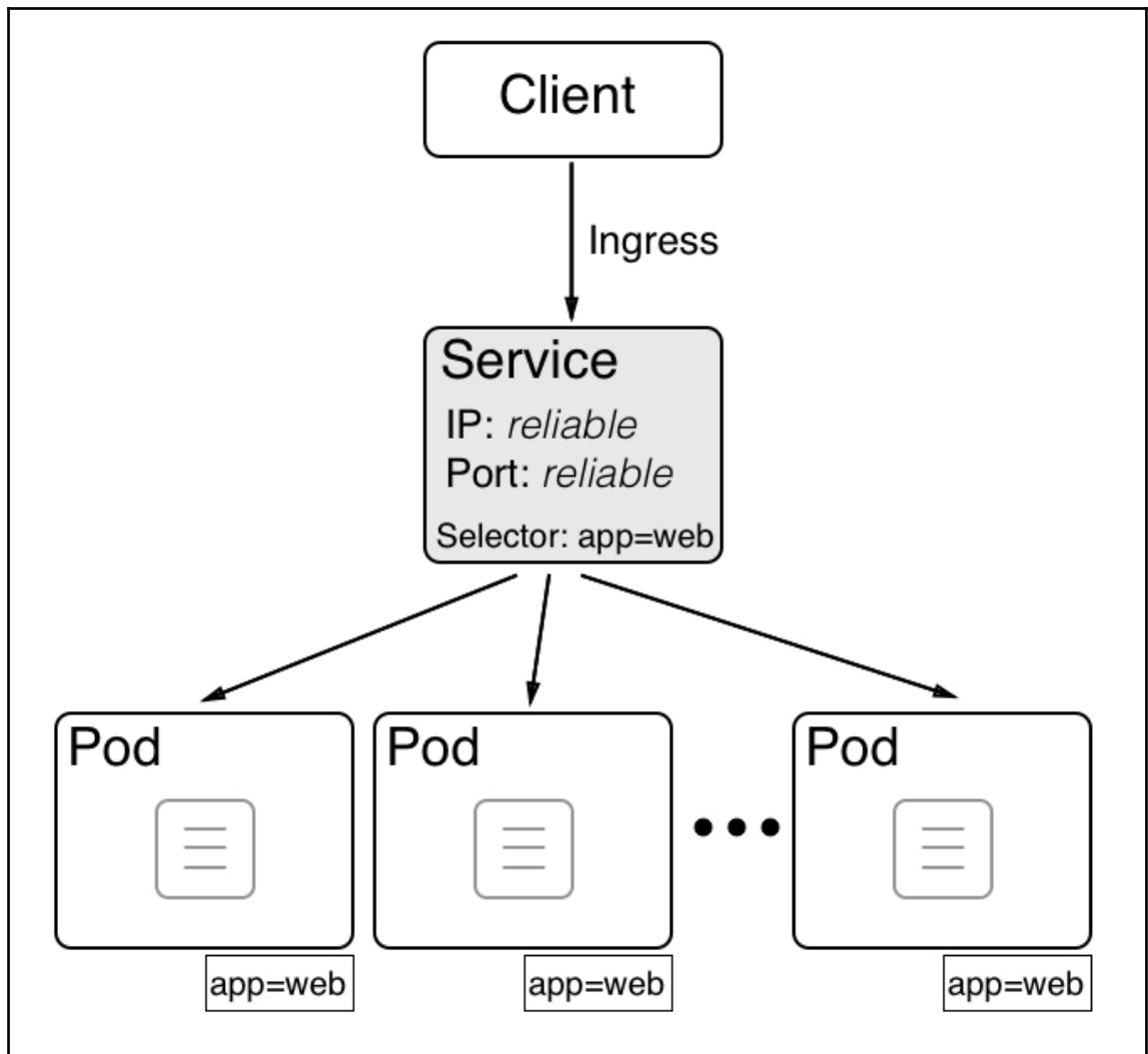


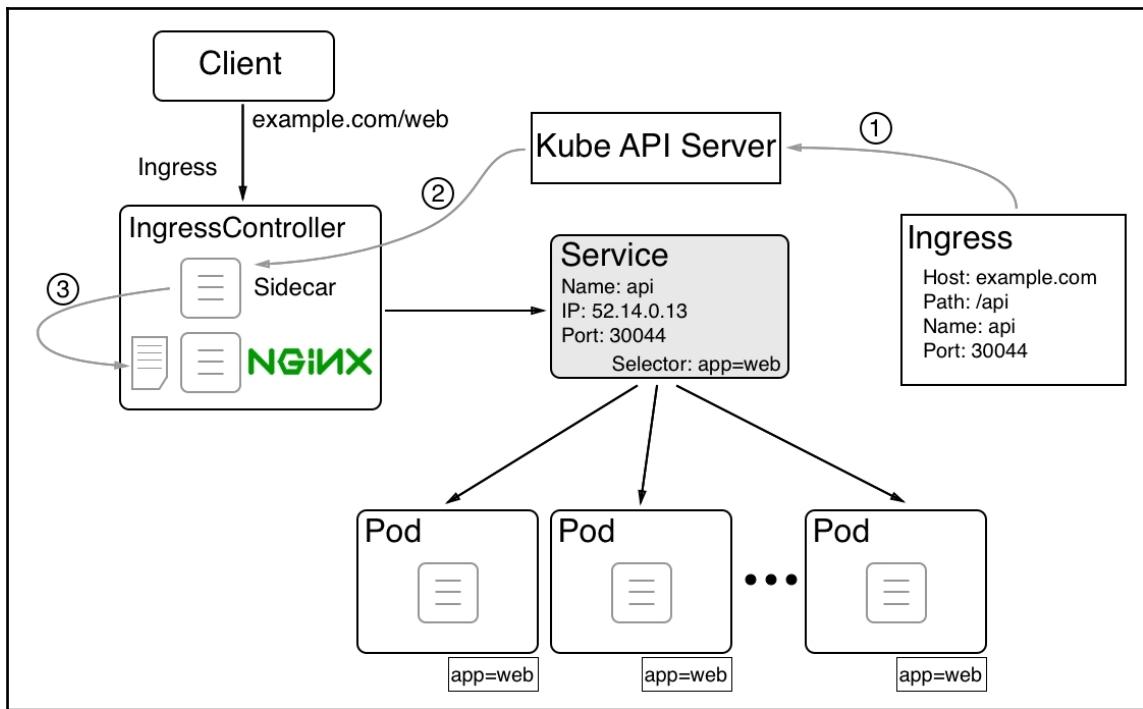
Web API

http://payments:3000
http://shipping:4000
http://ordering:5000

?







Chapter 13: Deploying, Updating, and Securing an Application with Kubernetes

```
! web-deployment.yaml ✘
```

```
1 apiVersion: extensions/v1beta1
2 kind: Deployment
3 metadata:
4   name: web
5 spec:
6   replicas: 1
7   selector:
8     matchLabels:
9       app: pets
10      service: web
11   template:
12     metadata:
13       labels:
14         app: pets
15         service: web
16   spec:
17     containers:
18       - image: fundamentalsofdocker/ch08-web:1.0
19         name: web
20         ports:
21           - containerPort: 3000
22             protocol: TCP
```

```
$ kubectl get all
NAME          DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
deploy/web    1         1         1           1          5m

NAME          DESIRED  CURRENT  READY       AGE
rs/web-769b88f67  1         1         1          5m

NAME          DESIRED  CURRENT  READY       AGE
deploy/web    1         1         1           1          5m

NAME          DESIRED  CURRENT  READY       AGE
rs/web-769b88f67  1         1         1          5m

NAME          READY     STATUS    RESTARTS  AGE
po/web-769b88f67-4fccx  1/1      Running   0          5m

NAME          TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)  AGE
svc/kubernetes  ClusterIP  10.96.0.1    <none>      443/TCP  8d
$
```

! web-service.yaml ×

```
1   apiVersion: v1
2   kind: Service
3   metadata:
4     name: web
5   spec:
6     type: NodePort
7     ports:
8       - port: 3000
9         protocol: TCP
10        selector:
11          app: pets
12          service: web
13
```

```
$ kubectl get services
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
kubernetes  ClusterIP  10.96.0.1      <none>        443/TCP      9d
web       NodePort   10.103.113.40  <none>        3000:30125/TCP  3m
$
```

```
! db-stateful-set.yaml ✘
```

```
1  apiVersion: apps/v1
2  kind: StatefulSet
3  metadata:
4    name: db
5  spec:
6    selector:
7      matchLabels:
8        app: pets
9        service: db
10   serviceName: db
11   template:
12     metadata:
13       labels:
14         app: pets
15         service: db
16   spec:
17     containers:
18       - image: fundamentalsofdocker/ch08-db:1.0
19         name: db
20         ports:
21           - containerPort: 5432
22         volumeMounts:
23           - mountPath: /var/lib/postgresql/data
24             name: pets-data
25   volumeClaimTemplates:
26     - metadata:
27       name: pets-data
28     spec:
29       accessModes:
30         - ReadWriteOnce
31       resources:
32         requests:
33           storage: 100Mi
34
```

```
$ kubectl get all
NAME          DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
deploy/web    1         1         1           1          27m

NAME          DESIRED  CURRENT  READY       AGE
rs/web-769b88f67  1         1         1          27m

NAME          DESIRED  CURRENT  UP-TO-DATE  AVAILABLE  AGE
deploy/web    1         1         1           1          27m

NAME          DESIRED  CURRENT  READY       AGE
rs/web-769b88f67  1         1         1          27m

NAME          DESIRED  CURRENT  AGE
statefulsets/db  1         1         49s

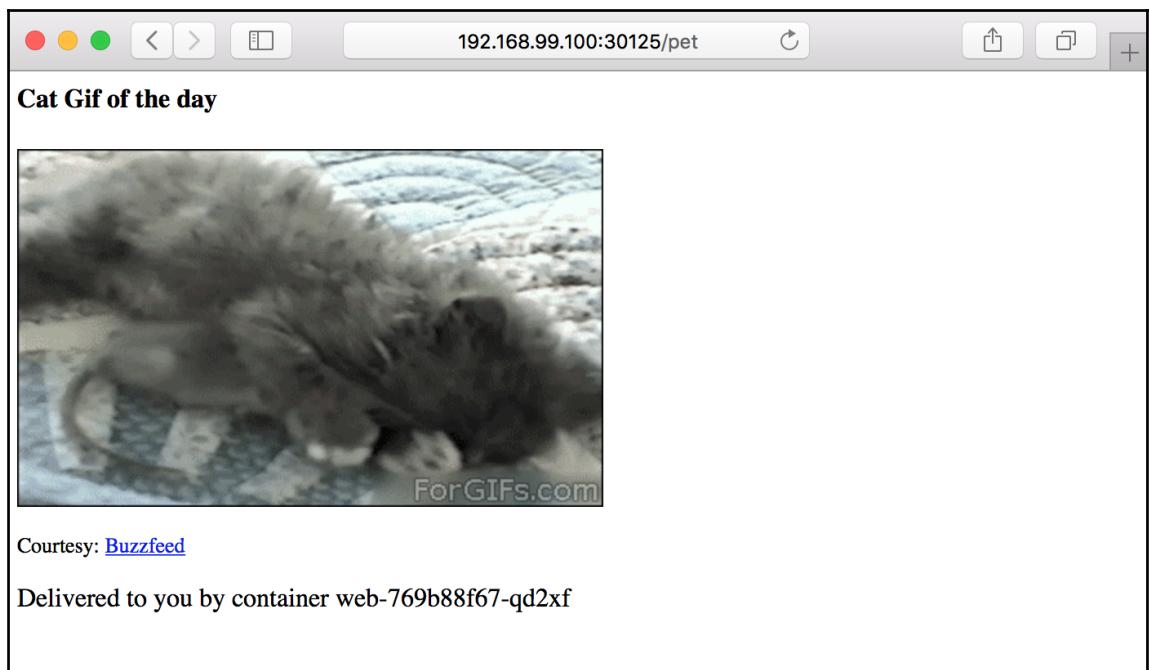
NAME          READY     STATUS    RESTARTS  AGE
po/db-0        1/1      Running   0          49s
po/web-769b88f67-qd2xf  1/1      Running   0          27m

NAME          TYPE        CLUSTER-IP      EXTERNAL-IP  PORT(S)      AGE
svc/kubernetes ClusterIP  10.96.0.1      <none>      443/TCP      10d
svc/web        NodePort    10.103.113.40  <none>      3000:30125/TCP 27m
$
```

! db-service.yaml ✘

Gabriel Schenker, 2 days ago | 1 comment

```
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: db
5 spec:
6   type: ClusterIP
7   ports:
8     - port: 5432
9       protocol: TCP
10  selector:
11    app: pets
12    service: db
```



```
$ kubectl create -f pets.yaml
deployment "web" created
service "web" created
statefulset "db" created
service "db" created
$ █
```

```
$ ./remove-pets.sh
deployment "web" deleted
service "web" deleted
statefulset "db" deleted
service "db" deleted
$
```

```
9     app.set('views', __dirname);
10
11    app.get('/',function(req,res){
12        res.status(200).send('Pets Demo Application v2\n');
13    });
14
```

Events:				
Type	Reason	Age	From	Message
Normal	ScalingReplicaSet	12m	deployment-controller	Scaled up replica set web-769b88f67 to 5
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled up replica set web-55cdf67cd to 1
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled down replica set web-769b88f67 to 4
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled up replica set web-55cdf67cd to 2
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled down replica set web-769b88f67 to 3
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled up replica set web-55cdf67cd to 3
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled down replica set web-769b88f67 to 2
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled up replica set web-55cdf67cd to 4
Normal	ScalingReplicaSet	3m	deployment-controller	Scaled down replica set web-769b88f67 to 1
Normal	ScalingReplicaSet	3m (x2 over 3m)	deployment-controller	(Combined from similar events): Scaled down replica set web-769b88f67 to 0

```
$ kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
web-55cdf67cd 5         5         5       27m
web-769b88f67 0         0         0       36m
$ █
```

```
$ kubectl get rs
NAME          DESIRED   CURRENT   READY   AGE
web-55cdf67cd 0         0         0       36m
web-769b88f67 5         5         5       45m
$ █
```

! web-deploy-blue.yaml ×

```
1  apiVersion: extensions/v1beta1
2  kind: Deployment
3  metadata:
4    name: web-blue
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: pets
10       service: web
11       color: blue
12    template:
13      metadata:
14        labels:
15          app: pets
16          service: web
17          color: blue
18    spec:
19      containers:
20        - image: fundamentalsofdocker/ch08-web:1.0
21          name: web
22          ports:
23            - containerPort: 3000
24              protocol: TCP
```

! web-svc-blue-green.yaml ×

```
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: web
5  spec:
6    type: NodePort
7    ports:
8      - port: 3000
9        protocol: TCP
10   selector:
11     app: pets
12     service: web
13     color: blue
14
```

! web-deploy-green.yaml ×

```
1  apiVersion: extensions/v1beta1
2  kind: Deployment
3  metadata:
4    name: web-green
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: pets
10       service: web
11       color: green
12    template:
13      metadata:
14        labels:
15          app: pets
16          service: web
17          color: green
18    spec:
19      containers:
20        - image: fundamentalsofdocker/ch13-web:2.0
21          name: web
22          ports:
23            - containerPort: 3000
24              protocol: TCP
25
```

```
$ kubectl get deploy
NAME      DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
web-blue   1          1          1            1           23h
web-green  1          1          1            1           3s
$ █
```

```
$ echo "john.doe" | base64
am9obi5kb2UK
$ echo "sEcReT-paSSwOrD" | base64
c0VjcmV0LXBhc1N3MHJECg==
$ █
```

```
$ kubectl create -f pets-secret.yaml
secret "pets-secret" created
$ kubectl describe secrets/pets-secret
Name:           pets-secret
Namespace:      default
Labels:         <none>
Annotations:    <none>

Type:  Opaque

Data
=====
password:  16 bytes
username:  9 bytes
$ █
```

```
$ kubectl get secrets/pets-secret -o yaml
apiVersion: v1
data:
  password: c0VjcmV0LXBhc1N3MHJECg==
  username: am9obi5kb2UK
kind: Secret
metadata:
  creationTimestamp: 2018-03-31T20:36:05Z
  name: pets-secret
  namespace: default
  resourceVersion: "154786"
  selfLink: /api/v1/namespaces/default/secrets/pets-secret
  uid: 22d818bd-3523-11e8-a3cb-080027c10823
type: Opaque
$
```

! web-deploy-secret.yaml ✘

```
1  apiVersion: extensions/v1beta1
2  kind: Deployment
3  metadata:
4    name: web
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: pets
10       service: web
11    template:
12      metadata:
13        labels:
14          app: pets
15          service: web
16    spec:
17      containers:
18        - image: fundamentalsofdocker/ch08-web:1.0
19          name: web
20          ports:
21            - containerPort: 3000
22              protocol: TCP
23          volumeMounts:
24            - name: secrets
25              mountPath: "/etc/secrets"
26              readOnly: true
27          volumes:
28            - name: secrets
29              secret:
30                secretName: pets-secret
```

```
$ kubectl exec -it web-597b7f7749-87mq5 -- /bin/sh
/app # cd /etc/secrets/
/etc/secrets # ls -l
total 0
lrwxrwxrwx    1 root      root          15 Apr  2 01:26 password -> ..data/password
lrwxrwxrwx    1 root      root          15 Apr  2 01:26 username -> ..data/username
/etc/secrets # cat username && cat password
john.doe
sEcret-pasSw0rD
/etc/secrets #
```

```
! web-deploy-secret-env.yaml ✘

1  apiVersion: extensions/v1beta1
2  kind: Deployment
3  metadata:
4    name: web
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: pets
10       service: web
11    template:
12      metadata:
13        labels:
14          app: pets
15          service: web
16    spec:
17      containers:
18        - image: fundamentalsofdocker/ch08-web:1.0
19          name: web
20          ports:
21            - containerPort: 3000
22              protocol: TCP
23
24      env:
25        - name: PETS_USERNAME
26          valueFrom:
27            secretKeyRef:
28              name: pets-secret
29              key: username
30        - name: PETS_PASSWORD
31          valueFrom:
32            secretKeyRef:
33              name: pets-secret
34              key: password
```

```
$ kubectl exec -it web-694f958cd4-6zq89 -- /bin/sh  
/app # echo $PETS_USERNAME && echo $PETS_PASSWORD  
john.doe  
sEcret-pasSw0rD  
/app # █
```

Chapter 14: Running a Containerized App in the Cloud

Add container

▼ Standard

Container name* ⓘ

Image* ⓘ

Custom image format: [registry-url]/[namespace]/[image]:[tag]

Memory Limits (MiB) ⓘ

Soft limit ▾

+ Add Hard limit

Define hard and/or soft memory limits in MiB for your container. Hard and soft limits correspond to the `memory` and `memoryReservation` parameters, respectively, in task definitions.

ECS recommends 300-500 MiB as a starting point for web applications.

Port mappings ⓘ

Container port **Protocol** ⓘ

+ Add port mapping

Host port mappings are not valid when the network mode for a task definition is host or awsvpc. To specify different host and container port mappings, choose the Bridge network mode.

Clusters Run Task

Task Definitions

Amazon ECR

Repositories

Select the cluster to run your task definition on and the number of copies of that task to run. To apply Advanced Options.

Launch type FARGATE EC2 i

Task Definition

Platform version i

Cluster i

Number of tasks

Task Group i

VPC and security groups

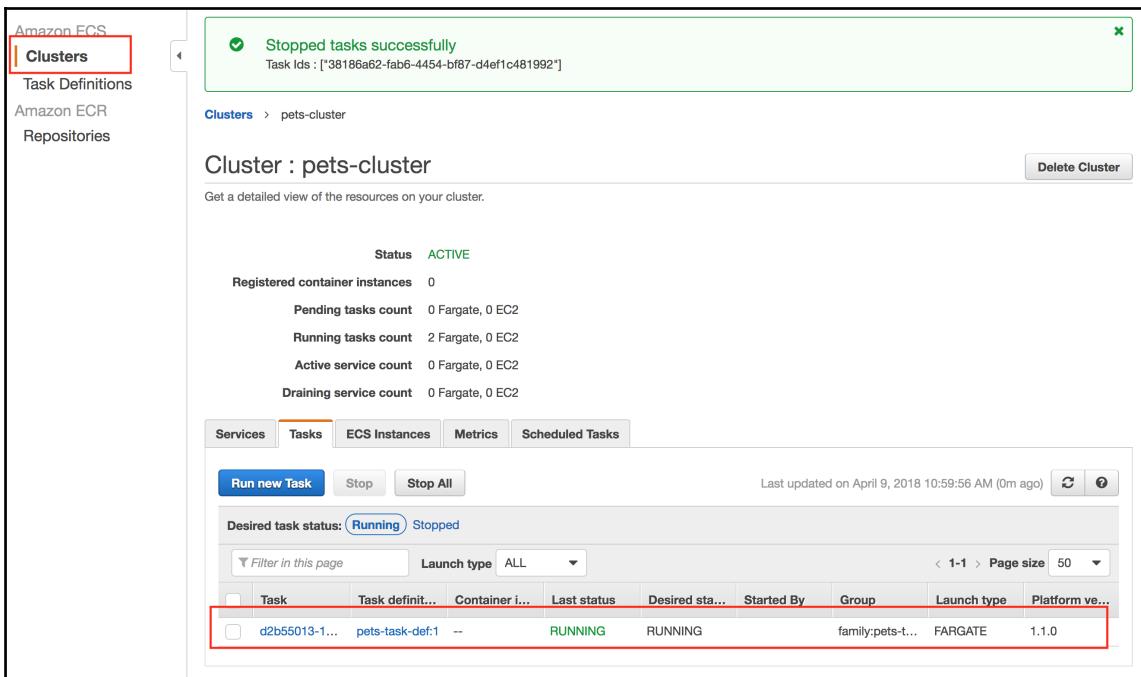
VPC and security groups are configurable when your task definition uses the awsvpc network mode.

Cluster VPC* i

Subnets* x i

Security groups* Edit i

Auto-assign public IP i



The screenshot shows the Amazon ECS console interface. The left sidebar has 'Clusters' selected, highlighted with a red box. The main content area shows a success message: 'Stopped tasks successfully' with Task IDs: ['38186a62-fab6-4454-bf87-d4ef1c481992']. Below this, the 'Cluster : pets-cluster' page is displayed. It shows cluster statistics: Status ACTIVE, Registered container instances 0, Pending tasks count 0 Fargate, 0 EC2, Running tasks count 2 Fargate, 0 EC2, Active service count 0 Fargate, 0 EC2, and Draining service count 0 Fargate, 0 EC2. A navigation bar at the top of the cluster page includes 'Delete Cluster'. Below the stats, there are tabs for Services, Tasks (selected), ECS Instances, Metrics, and Scheduled Tasks. Under the Tasks tab, there are buttons for 'Run new Task', 'Stop', and 'Stop All'. A note says 'Last updated on April 9, 2018 10:59:56 AM (0m ago)'. A table lists tasks, with one row highlighted by a red box: Task d2b55013-1..., Task definition pets-task-def:1, Container i. --, Last status RUNNING, Desired sta... RUNNING, Started By family:pets-t..., Group FARGATE, Launch type 1.1.0.



The screenshot shows a web browser window displaying the nginx welcome page. The address bar shows the IP address 54.204.209.77. The main content is a large bold heading 'Welcome to nginx!'. Below it, text reads: 'If you see this page, the nginx web server is successfully installed and working. Further configuration is required.' Another line of text says: 'For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com'. At the bottom, a thank you message reads: 'Thank you for using nginx.'



STORAGE AND LOGGING

Read only root file system i

Mount points

Source	Container path	Read only
pets-data	/var/lib/postgresql/data	<input type="checkbox"/> x

Add mount point



Courtesy: [Buzzfeed](#)

Delivered to you by container f0dd8a1c4495

Create Auto Scaling group Actions

Filter: pets 1 to 1 of 1 Auto Scaling Groups

Name	Launch Configuration	Instances	Desired	Min	Max	Availability Zones	Default Cooldown	Health Check
pets-group	pet-asg	3	3	3	3	us-east-1e	300	300

Auto Scaling Group: pets-group

Details Activity History Scaling Policies Instances Monitoring Notifications Tags Scheduled Actions Lifecycle Hooks Actions

Filter: Any Health Status Any Lifecycle State Filter instances... 1 to 3 of 3 Instances

Instance ID	Lifecycle	Launch Configuration Name	Availability Zone	Health Status	Protected from
i-0a4697b10a1180de3	InService	pet-asg	us-east-1e	Healthy	
i-0b71beb770f665e4a	InService	pet-asg	us-east-1e	Healthy	
i-0c3d2741065073ed6	InService	pet-asg	us-east-1e	Healthy	

Create Security Group Actions

Group ID : sg-945e9bdd Add filter 1 to 1 of 1

Name	Group ID	Group Name	VPC ID	Description
sg-945e9bdd	pets-sg	vpc-f3723f96	Pets Security Group	

Security Group: sg-945e9bdd

Description Inbound Outbound Tags

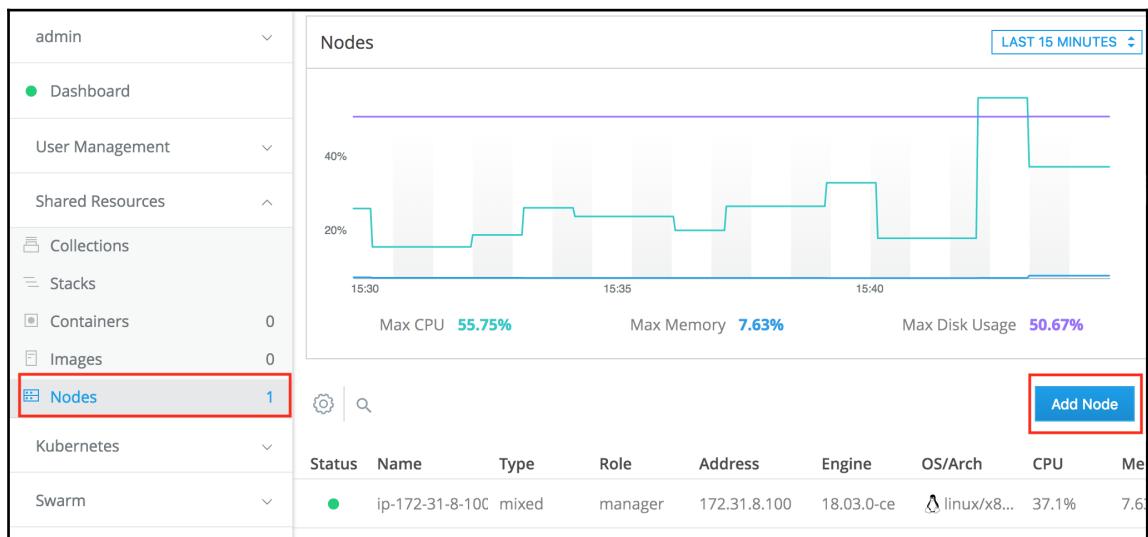
Edit

Type	Protocol	Port Range	Source	Description
All traffic	All	All	70.113.114.234/32	
All traffic	All	All	sg-945e9bdd (pets-sg)	
SSH	TCP	22	0.0.0.0/0	

```

ubuntu@ip-172-31-8-100:~$ docker container run --rm -it --name ucp \
>   -v /var/run/docker.sock:/var/run/docker.sock \
>   docker/ucp:${UCP_VERSION} install \
>   --admin-username admin \
>   --admin-password adminadmin \
>   --san ${UCP_IP} \
>   --san ${UCP_FQDN}
INFO[0000] Verifying your system is compatible with UCP 3.0.0-beta2 (4f665c3)
INFO[0000] Your engine version 18.03.0-ce, build 0520e24 (4.4.0-1052-aws) is compatible
INFO[0000] All required images are present
INFO[0000] Initializing a new swarm at 172.31.8.100
INFO[0005] Establishing mutual Cluster Root CA with Swarm
INFO[0008] Installing UCP with host address 172.31.8.100 - If this is incorrect, please sp
INFO[0008] Generating UCP Client Root CA
INFO[0008] Deploying UCP Service
INFO[0049] Installation completed on ip-172-31-8-100 (node jatip5ocsvhighii1o55ho41v)
INFO[0049] UCP Instance ID: 803f54eedvsdlc2wvfju0iv47
INFO[0049] UCP Server SSL: SHA-256 Fingerprint=51:E8:13:FF:5F:2C:89:CC:E8:53:46:5C:D9:2F:3
INFO[0049] Login to UCP at https://172.31.8.100:443
INFO[0049] Username: admin
INFO[0049] Password: (your admin password)
ubuntu@ip-172-31-8-100:~$ 

```



```

$ ssh -i pets.pem ubuntu@54.208.149.247
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1052-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

30 packages can be updated.
0 updates are security updates.

*** System restart required ***
Last login: Sun Apr  8 20:58:32 2018 from 70.113.114.234
ubuntu@ip-172-31-6-57:~$ docker swarm join --token SWMTK... 172.31.8.100:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-6-57:~$ 

```

Status	Name	Type	Role	Address	Engine	OS/Arch	CPU	Memory	Disk	Details
●	ip-172-31-8-100	mixed	manager	172.31.8.100	18.03.0-ce	🐧 linux/x8...	31.95%	7.71%	50.73%	Healthy UCP ...
●	ip-172-31-15-11	swarm	worker	172.31.15.110	18.03.0-ce	🐧 linux/x8...	3.5%	1.05%	31.36%	Healthy UCP ...
●	ip-172-31-6-57	swarm	worker	172.31.6.57	18.03.0-ce	🐧 linux/x8...	3.27%	1.07%	31.36%	Healthy UCP ...

Profile

- [Client Bundles](#)
- [Default Collection](#)
- [All Roles](#)
- [My Grants](#)
- [Security](#)

New Client Bundle ^

Generate Client Bundle

Add Existing Client Bundle

21:34:09 UTC

PUBLIC KEY

```
-----BEGIN
MFkwEwYHKoZ
CESN5GODNgG
-----END PU
```

```
$ docker node ls
ID          HOSTNAME   STATUS  AVAILABILITY  MANAGER STATUS  ENGINE VERSION
wougljiphzk4vmmmbm1tlqi2kg  ip-172-31-6-57  Ready  Active        Leader           18.03.0-ce
jatip5ccsvhighiilo55ho4iv * ip-172-31-8-100  Ready  Active
tlkaeww3idlt90ko5zr8xkeu   ip-172-31-15-110 Ready  Active
$
```

```
$ docker stack deploy -c stack.yml pets
Creating network pets_pets-net
Creating service pets_db
Creating service pets_web
$
```

The screenshot shows the Docker Desktop interface with the 'Stacks' section highlighted. On the left, there's a sidebar with 'admin' at the top, followed by 'Dashboard' (green dot), 'User Management', 'Shared Resources', 'Collections' (with 'Stacks' highlighted in red), 'Containers', and 'Images'. The main area has a title '2 Stacks' and a search bar. Below it is a table with columns 'Type', 'Name', 'Services/Containers', and 'Networks'. One row is selected and highlighted with a red border: 'Swarm Services' for 'pets', with 2 services and 1 network. Another row for 'Docker Universal Control Plane' is visible below it.

Type	Name	Services/Containers	Networks
Swarm Services	pets	2	1
Basic Containers	Docker Universal Control Plane 803f54eedvsdlc2wvfju0...	9	0

The screenshot shows the Docker Swarm interface. On the left, a sidebar lists 'Dashboard', 'User Management', 'Shared Resources', 'Kubernetes', and 'Swarm'. Under 'Swarm', 'Services' is selected and highlighted with a red box, showing a count of 2. Other options include 'Volumes' (4), 'Networks' (14), and 'Secrets' (0). The main panel displays '2 Services' with a table:

Status	Name	Image	Mode	Updated At	Last Error
● 1/1	pets_web	fundamen...	Replicated	6 minutes ago	No errors
● 1/1	pets_db	fundamen...	Replicated	6 minutes ago	No errors

On the right, a detailed view for the 'pets_web' service is shown, also with a red box around its endpoint information:

- pets_web
- Last updated 6 minutes ago
- Inspect Resource | Configure
- CONFIGURATION
- ID: 0h54r4sbnc9wom0yaiwjx0lt
- Name: pets_web
- Created: Today at 4:46 PM
- Updated: Today at 4:46 PM
- Update Status: Never Updated
- Published Endpoints:
http://34.232.53.86:3000
- Collection: /

```
$ kubectl get nodes
NAME           STATUS    ROLES      AGE   VERSION
ip-172-31-15-110  Ready    <none>    1h    v1.8.2-docker.128+56ab40b2f3e9b9
ip-172-31-6-57   Ready    <none>    1h    v1.8.2-docker.128+56ab40b2f3e9b9
ip-172-31-8-100  Ready    master    22h   v1.8.2-docker.128+56ab40b2f3e9b9
$ █
```

```
$ kubectl create -f pets.yaml
deployment "web" created
service "web" created
deployment "db" created
service "db" created
$ █
```

```
bash-4.3# az group create --name pets-rg --location eastus
{
  "id": "/subscriptions/186760ad-9152-4499-b317-c9bff441fb9d/resourceGroups/pets-rg",
  "location": "eastus",
  "managedBy": null,
  "name": "pets-rg",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}
bash-4.3#
```

```
bash-4.3# az acs create -g pets-rg -n pets-cluster --orchestrator-type swarm --generate-ssh-keys
{
  "id": "/subscriptions/186760ad-9152-4499-b317-c9bff441fb9d/resourceGroups/pets-rg/providers/MicrosoftContainerServices/pets-cluster",
  "name": "azurecli1523325303.63794370404",
  "properties": {
    "additionalProperties": {
      "duration": "PT10M22.7071402S",
      "outputResources": [
        {
          "id": "/subscriptions/186760ad-9152-4499-b317-c9bff441fb9d/resourceGroups/pets-rg/providers/MicrosoftContainerServices/pets-cluster",
          "resourceGroup": "pets-rg"
        }
      ],
      "templateHash": "14213326594306665848"
    },
    "correlationId": "94aebdbd-b8d0-40fc-b492-e285e1a364bb",
    "debugSetting": null,
    "dependencies": [],
    "mode": "Incremental",
    "outputs": {
      "agentFQDN": {
        "type": "String",
        "value": "pets-clust-pets-rg-186760agent.eastus.cloudapp.azure.com"
      },
      "masterFQDN": {
        "type": "String",
        "value": "pets-clust-pets-rg-186760mgmt.eastus.cloudapp.azure.com"
      },
      "sshMaster0": {
```

```

bash-4.3# az network public-ip list --resource-group pets-rg \
>     --query "[*].{Name:name,IPAddress:ipAddress}" \
>     -o table
Name                                IPAddress
-----
swarm-agent-ip-pets-clust-pets-rg-186760agent-CE7D8170 13.92.172.89
swarm-master-ip-pets-clust-pets-rg-186760mgmt-CE7D8170 13.90.151.207
bash-4.3# █

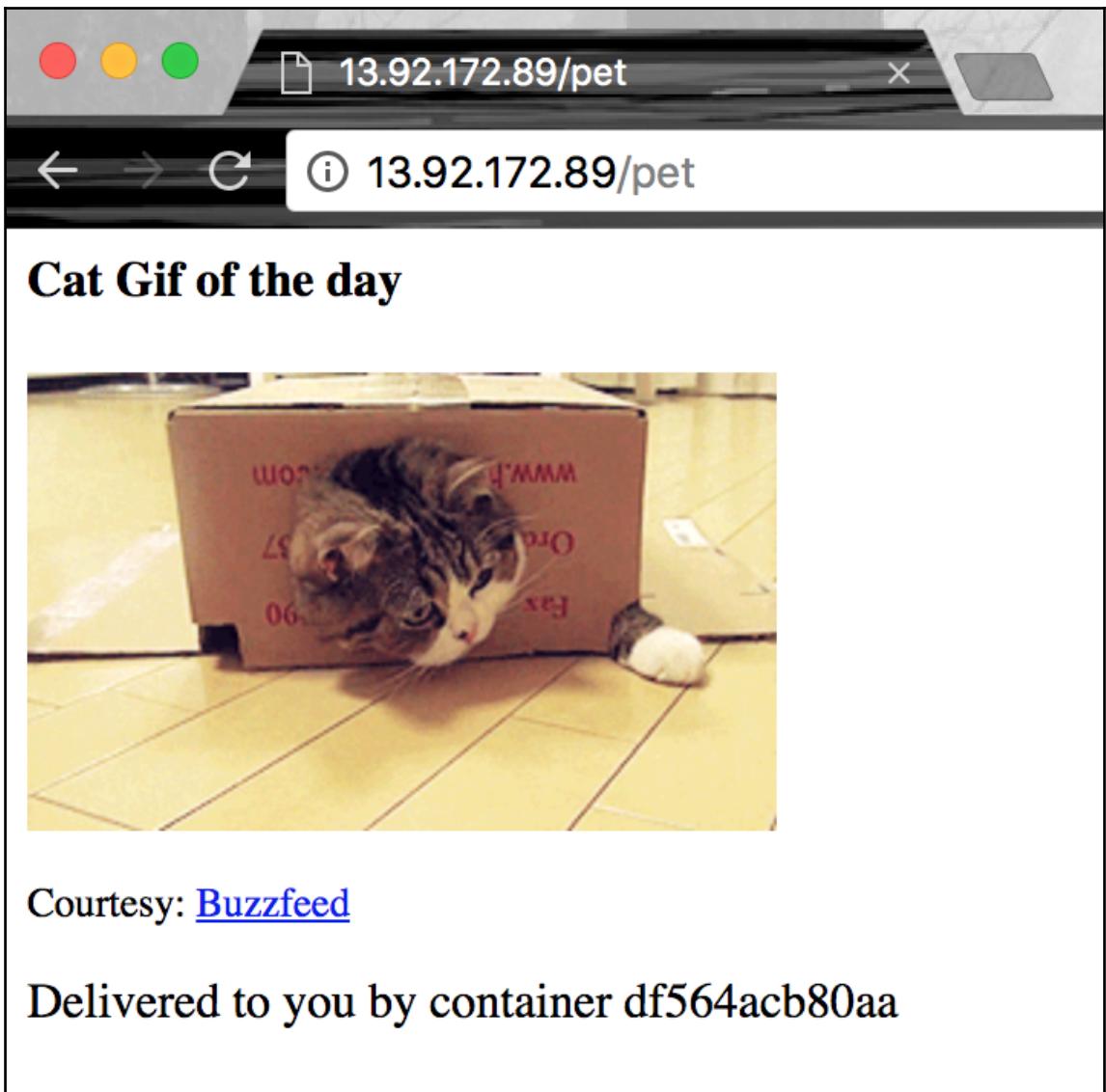
```

```

$ docker info
Containers: 1
Running: 1
Paused: 0
Stopped: 0
Images: 1
Role: primary
Strategy: spread
Filters: health, port, dependency, affinity, constraint
Nodes: 3
swarm-agent-CE7D8170000000: 10.0.0.4:2375
└ Status: Healthy
└ Containers: 0
└ Reserved CPUs: 0 / 2
└ Reserved Memory: 0 B / 7.137 GiB
└ Labels: executiondriver=<not supported>, kernelversion=3.19.0-65-generic, operatingsystem=Ubuntu 14.04.4 LTS, storagedriver=overlay
└ Error: (none)
└ UpdatedAt: 2018-04-07T20:25:56Z
swarm-agent-CE7D8170000001: 10.0.0.5:2375
└ Status: Healthy
└ Containers: 0
└ Reserved CPUs: 0 / 2
└ Reserved Memory: 0 B / 7.137 GiB
└ Labels: executiondriver=<not supported>, kernelversion=3.19.0-65-generic, operatingsystem=Ubuntu 14.04.4 LTS, storagedriver=overlay
└ Error: (none)
└ UpdatedAt: 2018-04-07T20:26:22Z
swarm-agent-CE7D8170000002: 10.0.0.6:2375
└ Status: Healthy
└ Containers: 1
└ Reserved CPUs: 0 / 2
└ Reserved Memory: 0 B / 7.137 GiB
└ Labels: executiondriver=<not supported>, kernelversion=3.19.0-65-generic, operatingsystem=Ubuntu 14.04.4 LTS, storagedriver=overlay
└ Error: (none)
└ UpdatedAt: 2018-04-07T20:25:59Z
Plugins:
Volume:
Network:
Log:
Swarm:

```

```
$ docker-compose up
Creating network "azure_default" with the default driver
Creating volume "azure_pets-data" with default driver
Pulling web (fundamentalsofdocker/ch08-web:1.0)...
swarm-agent-CE7D8170000001: Pulling fundamentalsofdocker/ch08-web:1.0... : downloaded
swarm-agent-CE7D8170000002: Pulling fundamentalsofdocker/ch08-web:1.0... : downloaded
swarm-agent-CE7D8170000000: Pulling fundamentalsofdocker/ch08-web:1.0... : downloaded
Pulling db (fundamentalsofdocker/ch08-db:1.0)...
swarm-agent-CE7D8170000000: Pulling fundamentalsofdocker/ch08-db:1.0... : downloaded
swarm-agent-CE7D8170000001: Pulling fundamentalsofdocker/ch08-db:1.0... : downloaded
swarm-agent-CE7D8170000002: Pulling fundamentalsofdocker/ch08-db:1.0... : downloaded
Creating azure_db_1 ... done
Creating azure_web_1 ... done
Attaching to azure_web_1, azure_db_1
web_1  | Listening at 0.0.0.0:3000
db_1   | The files belonging to this database system will be owned by user "postgres".
db_1   | This user must also own the server process.
db_1   |
db_1   | The database cluster will be initialized with locale "en_US.utf8".
db_1   | The default database encoding has accordingly been set to "UTF8".
db_1   | The default text search configuration will be set to "english".
db_1   |
db_1   | Data page checksums are disabled.
db_1   |
db_1   | fixing permissions on existing directory /var/lib/postgresql/data ... ok
db_1   | creating subdirectories ... ok
db_1   | selecting default max_connections ... 100
db_1   | selecting default shared_buffers ... 128MB
db_1   | selecting dynamic shared memory implementation ... posix
db_1   | creating configuration files ... ok
db_1   | running bootstrap script ... ok
db_1   | performing post-bootstrap initialization ... sh: locale: not found
db_1   | 2018-04-07 20:34:10.714 UTC [27] WARNING: no usable system locales were found
db_1   | ok
db_1   | syncing data to disk ...
db_1   | WARNING: enabling "trust" authentication for local connections
db_1   | You can change this by editing pg_hba.conf or using the option -A, or
```



Google Cloud Platform My First Project

Kubernetes clusters

CREATE CLUSTER REFRESH DELETE SHOW INFO PANEL

Filter by label or name

Kubernetes clusters

Name	Location	Cluster size	Total cores	Total memory	Notifications	Labels
<input checked="" type="checkbox"/> pets-cluster	us-central1-a	3	3 vCPUs	11.25 GB		

electric-clone-200421 x +

Welcome to Cloud Shell! Type "help" to get started.
schenker_gabriel@electric-clone-200421:~\$

This screenshot shows the Google Cloud Platform interface. At the top, it displays 'Google Cloud Platform' and 'My First Project'. Below this is a navigation bar with icons for Kubernetes clusters, Compute Engine, Storage, and more. The main area is titled 'Kubernetes clusters' and shows a table with one entry: 'pets-cluster' located in 'us-central1-a' with 3 vCPUs and 11.25 GB of memory. A 'Connect' button is available for this cluster. Below this is a 'Cloud Shell' window titled 'electric-clone-200421'. It displays a welcome message and a command prompt: 'Welcome to Cloud Shell! Type "help" to get started.' followed by 'schenker_gabriel@electric-clone-200421:~\$'. The bottom part of the shell window shows a command being run: 'kubectl get svc/web --watch'. The output of this command is shown in a table:

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
web	LoadBalancer	10.43.244.195	35.184.142.67	3000:32086/TCP	25m

```
schenker_gabriel@electric-clone-200421:~/labs/ch14/gce$ kubectl get svc/web --watch
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
web      LoadBalancer      10.43.244.195      35.184.142.67      3000:32086/TCP      25m
```