0 to Swarm

In An Automated Fashion

Mike Anderson
Chief Architect – Software Development
mmandersonii@securitymetrics.com

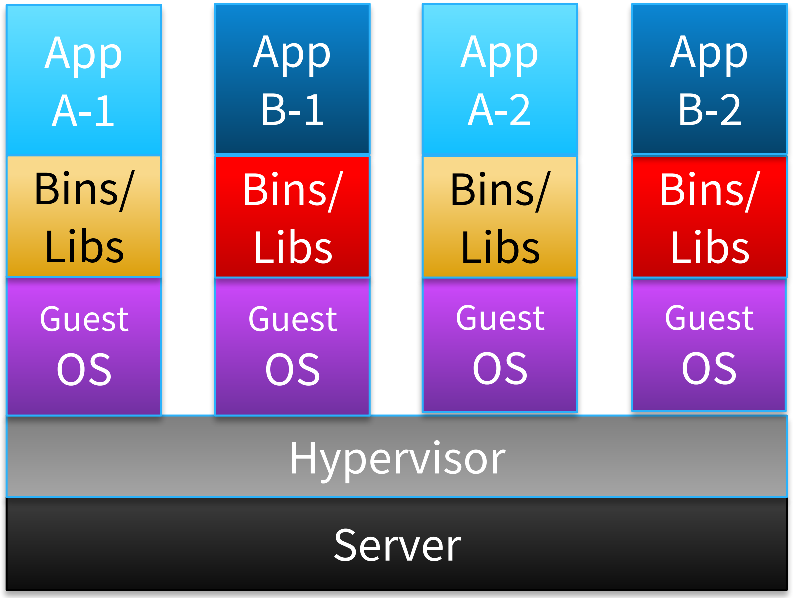
securityMETRICS®

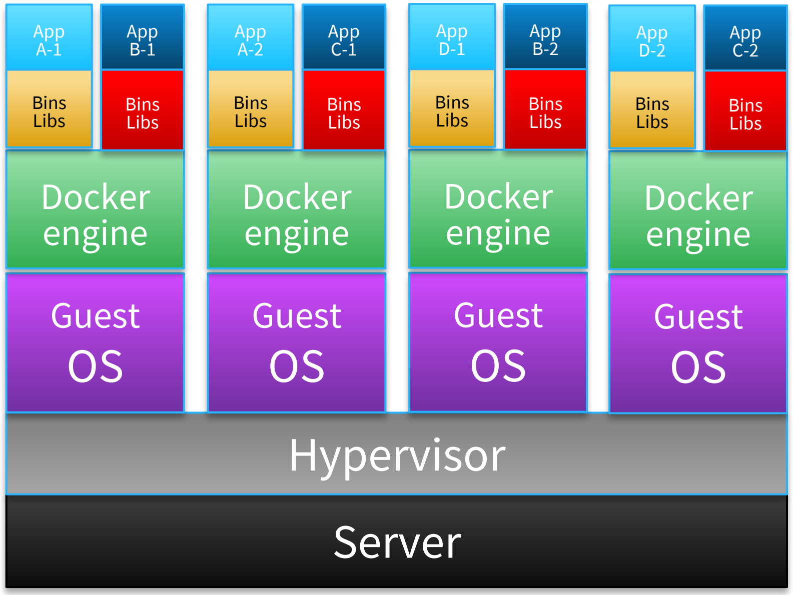
ABOUT SECURITYMETRICS

Helping organizations comply with mandates, avoid security breaches, and recover from data theft since 2000

securityMETRICS®

Why Docker?
VMs vs Docker Images / Containers



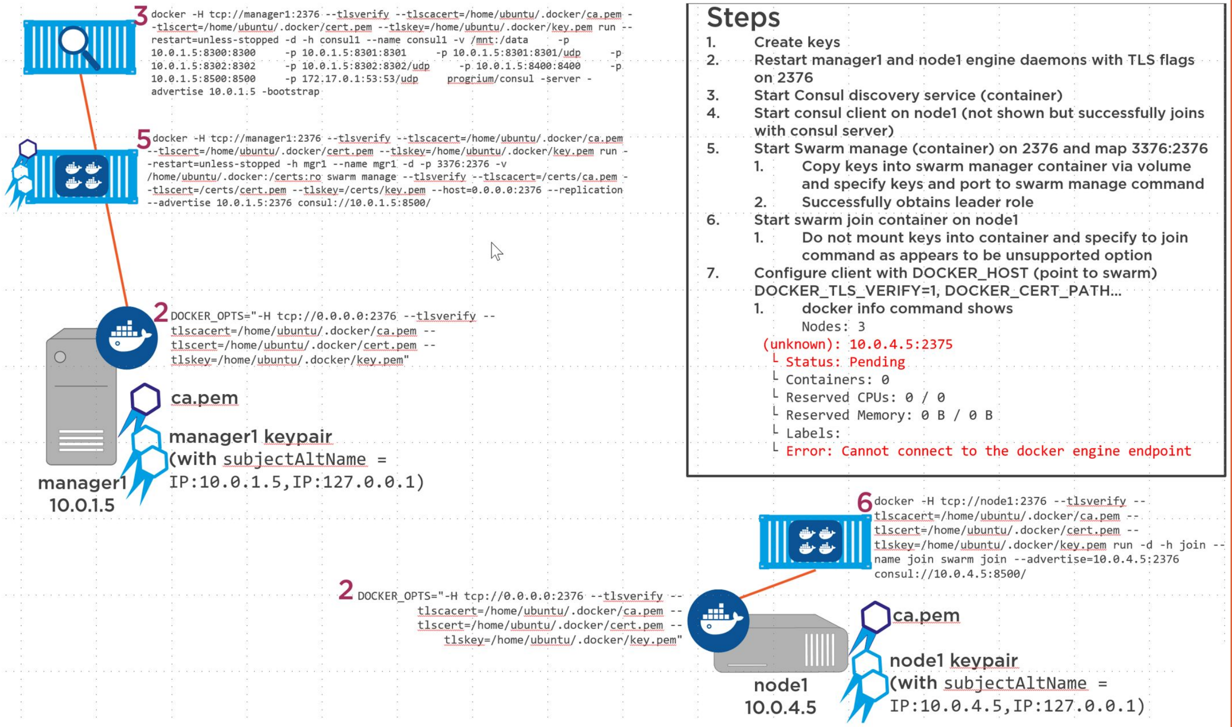


securityMETRICS®

Setting up a Docker Swarm

The old way

(many steps and commands not shown)



3
```
docker -H tcp://manager1:2376 --tlsverify --tlscacert=/home/ubuntu/.docker/ca.pem -
-tlscert=/home/ubuntu/.docker/cert.pem --tlskey=/home/ubuntu/.docker/key.pem run --
restart=unless-stopped -d -h consul1 --name consul1 -v /mnt:/data     -p
10.0.1.5:8300:8300     -p 10.0.1.5:8301:8301     -p 10.0.1.5:8301:8301/udp     -p
10.0.1.5:8302:8302     -p 10.0.1.5:8302:8302/udp     -p 10.0.1.5:8400:8400     -p
10.0.1.5:8500:8500     -p 172.17.0.1:53:53/udp     progrium/consul -server -
advertise 10.0.1.5 -bootstrap
```

5
```
docker -H tcp://manager1:2376 --tlsverify --tlscacert=/home/ubuntu/.docker/ca.pem
--tlscert=/home/ubuntu/.docker/cert.pem --tlskey=/home/ubuntu/.docker/key.pem run -
-restart=unless-stopped -h mgr1 --name mgr1 -d -p 3376:2376 -v
/home/ubuntu/.docker:/certs:ro swarm manage --tlsverify --tlscacert=/certs/ca.pem -
-tlscert=/certs/cert.pem --tlskey=/certs/key.pem --host=0.0.0.0:2376 --replication
--advertise 10.0.1.5:2376 consul://10.0.1.5:8500/
```

2
```
DOCKER_OPTS="-H tcp://0.0.0.0:2376 --tlsverify --
tlscacert=/home/ubuntu/.docker/ca.pem --
tlscert=/home/ubuntu/.docker/cert.pem --
tlskey=/home/ubuntu/.docker/key.pem"
```

ca.pem

manager1 keypair
(with subjectAltName =
IP:10.0.1.5,IP:127.0.0.1)

manager1
10.0.1.5

Steps

1. Create keys
2. Restart manager1 and node1 engine daemons with TLS flags on 2376
3. Start Consul discovery service (container)
4. Start consul client on node1 (not shown but successfully joins with consul server)
5. Start Swarm manage (container) on 2376 and map 3376:2376
   1. Copy keys into swarm manager container via volume and specify keys and port to swarm manage command
   2. Successfully obtains leader role
6. Start swarm join container on node1
   1. Do not mount keys into container and specify to join command as appears to be unsupported option
7. Configure client with DOCKER_HOST (point to swarm) DOCKER_TLS_VERIFY=1, DOCKER_CERT_PATH…
   1. docker info command shows

```
        Nodes: 3
 (unknown): 10.0.4.5:2375
  └ Status: Pending
  └ Containers: 0
  └ Reserved CPUs: 0 / 0
  └ Reserved Memory: 0 B / 0 B
  └ Labels:
  └ Error: Cannot connect to the docker engine endpoint
```

6
```
docker -H tcp://node1:2376 --tlsverify --
tlscacert=/home/ubuntu/.docker/ca.pem --
tlscert=/home/ubuntu/.docker/cert.pem --
tlskey=/home/ubuntu/.docker/key.pem run -d -h join --
name join swarm join --advertise=10.0.4.5:2376
consul://10.0.4.5:8500/
```

2
```
DOCKER_OPTS="-H tcp://0.0.0.0:2376 --tlsverify --
    tlscacert=/home/ubuntu/.docker/ca.pem --
    tlscert=/home/ubuntu/.docker/cert.pem --
        tlskey=/home/ubuntu/.docker/key.pem"
```

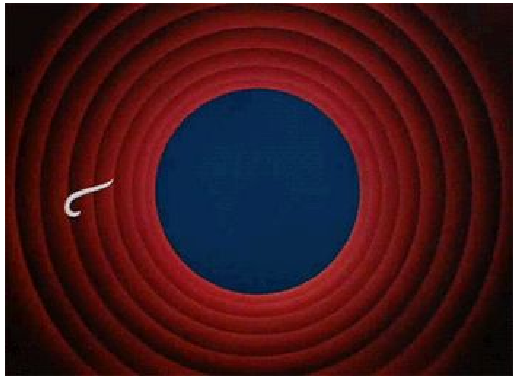
ca.pem

node1 keypair
(with subjectAltName =
IP:10.0.4.5,IP:127.0.0.1)

node1
10.0.4.5

The new way

```
$ docker swarm init
$ docker swarm join
```



securityMETRICS®

Quick and easy

- Make sure you have Docker installed on each node
  - Need version 1.12 or later
  - Version 17.05 CE recommended since it has some additional features that make life much easier
- Initialize the swarm on one node
  - docker swarm init --advertise-addr eth0
- Using the worker or the manager join-token, add additional nodes

```
TOKEN=$(docker swarm join-token -q manager)
for N in $(seq 2 5); do
        DOCKER_HOST=tcp://node$N:2375 docker swarm join --token $TOKEN node1:2377
done
```

Play-with-docker

- https://github.com/play-with-docker/play-with-docker
  - source code, written in go
- http://play-with-docker.com/
  - live environment that lets you try all docker features in a fast environment since it is running in Amazon
- Presented during the Moby's Cool Hack session at DockerCon this year.
  - DockerCon 2017 Moby's Cool Hack Session
- http://training.play-with-docker.com/
  - Labs and tutorials on Docker topics

securityMETRICS®

play-with-docker demo

What did we just do?

- Since version 1.12, Docker Engine embeds SwarmKit
  - SwarmKit is an open source toolkit to build multi-node systems. Similar to libcontainer, libnetwork, vpnkit, etc.
  - See https://github.com/docker/swarmkit
- SwarmKit is “asleep” until you ”enable” Swarm Mode
  - docker swarm init
- Other SwarmKit commands
  - docker swarm (initialize, join, manage cluster parameters)
  - docker node (view, manage, promote, demote nodes)
  - docker service (create and manage services)

securityMETRICS®

play-with-docker is cool but I want more

- Vagrant – stand up local vms to demonstrate/test swarm and tools
  - Only really needed if there isn't another way to set up the necessary VMs
- Ansible – define inventories of "nodes" and roles for them and reproducibly provision them
  - To create a new environment, create an inventory file that describes the different machines available and what their roles are
  - Only requires ssh access and python installed with the ability to use sudo unless logging in as root

Show me the goodies

- https://github.com/mmanderson/docker-swarm-init
  - Set up a local swarm to play with
  - Can be used to provision other swarms with different inventory files
- https://github.com/jpetazzo/orchestration-workshop
  - Amazing repo that has additional information and a full workshop to explore additional history and samples
  - We will deploy the DockerCoins solution in our swarm for our demo
  - Jérôme Petazzoni works for Docker and is a great presenter and all around nice guy. He regularly presents this workshop and encourages others to use it as well
- https://github.com/dockersamples/docker-swarm-visualizer
  - Lets you see what is running in your swarm

securityMETRICS®

What will it give me?

- A virtual machine running ElasticStack for logging
- A 4 node docker swarm with 3 managers and 1 worker (just to show some of the differences)
  - Each docker machine is configured to log to ElasticStack by default. This can be overridden but it is nice to go to one place for logs

securityMETRICS®

Local swarm demo

Additional topics

- Swarm overlay networks
  - The playbook that I've demonstrated creates one for you
  - Each service gets a Virtual IP address in the swarm and the swarm load balances across the service instances
- Stack files
  - Easiest way to stand up a suite of services
- Docker secrets
  - only deployed to swarm nodes that need them.
- Metrics
  - What's happening in the swarm

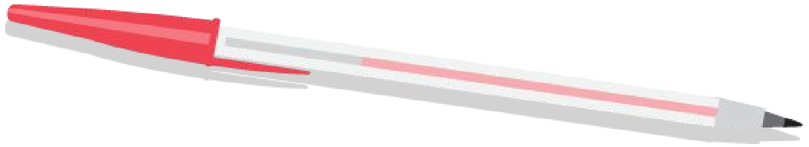
securityMETRICS®

Stacks and Metrics

Questions?

- Github repo: https://github.com/mmanderson/docker-swarm-init
- My email: mmandersonii@securitymetrics.com

www.securitymetrics.com

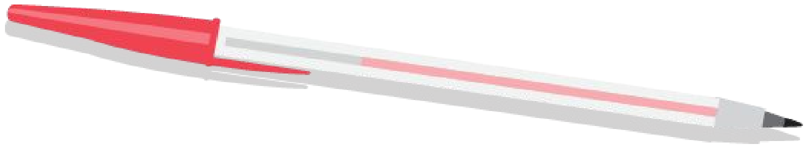


securityMETRICS®

Thank you!

- Github repo: https://github.com/mmanderson/docker-swarm-init
- My email: mmandersonii@securitymetrics.com

www.securitymetrics.com



securityMETRICS®