Docker Notes

|  |  |
| --- | --- |
| Install Docker on Windows 10 Pro | |
| Pre Req : Windows 10 Pro | |
| Enable Hyper-V | Go to windows features  Enable Hyper-V  Restart |
| Download Docker for windows | Verify after msi using  PS : docker version  PS : docker info  PS : docker images |

|  |  |
| --- | --- |
| Install Docker on Windows Server 2016 | |
| Pre Req : Windows Server 2016 | |
| Install container features | PS : Install-WindowsFeature containers PS : Restart-Computer |
| Create Folder | C:\Program Files\docker |
| Download Docker Binaries | DL : <https://aka.ms/tp5/b/dockerd>  DL : <https://aka.ms/tp5/b/docker>  place binaries in C:\Program Files\docker |
| Add docker folder to System Path |  |
| Restart PS |  |
| Run script | PS : dockerd --register-service  PS : Start-Service docker  Verify via PS : Get-Service docker |
| Before starting container, need to download base OS image | PS : Install-PackageProvier ContainerImage -Force PS : Find-ContainerImage PS : Install-ContainerImage windowServerCore PS : Restart-Service docker  verify PS : docker images //should see windowsServerCore |
| Tag image as latest | Docker tag {image id} windowservercore:latest  verify PS : docker images // should see windowsServerCore as latest |
| Run script | Docker run -it windowsservercore cmd //-it attach powershell terminal to docker terminal  run PS : hostname in PS and CMD to see different environment  CTRL + PQ to exit  run PS : docker ps //see container running |

|  |  |
| --- | --- |
| Notes | |
| Containers vs Virtual Machines | Containers vs. VMs |
| Docker hubs | <https://hub.docker.com> |
| Contain and images | Images - Stopped containers Containers – running images |
| Container Lifecycle | Stop -> Run -> Stop -> Run… docker rm will wipe it, otherwise it’s persistent |
| Top Level Images | \*Official images ie. Ubuntu, Redis, alpine, nginx |
| Second Level Images | \*contributed images dockercould/haproxy cockpit/ws |

|  |  |
| --- | --- |
| Docker Hello World | |
| Run Script | Docker run hello-world |

|  |  |
| --- | --- |
| Commands | Info |
| docker {cmd} --help | List more info about command |
| docker run | Run container |
| docker run -d --name web -p 80:8080 {image} | -d detached -p 80:8080 80 = docker host  8080 = docker container port |
| docker run -it –name temp {ubuntu:latest} /bin/bash | Run interactive terminal of ubuntu:latest in current terminal \*terminal is very lightweight |
| docker ps | List containers |
| docker ps -a | Show all containers |
| docker pull {image} | Pull image |
| docker pull {image}:{version} | Specify specific version |
| docker rmi {name:tag} | Remove image |
| docker stop $(docker ps -aq) | Stop all containers -a lists all -q quiet mode, shows id only |
| docker rm $(docker ps -aq) | Remove all containers |
| docker rmi $(docker images -q) | Remove all images |

Docker Swarm

|  |  |
| --- | --- |
| Swarm Mode Theory (Docker 1.12 and later) | |
| A cluster = a swarm |  |
| Engines in a swarm run in swarm mode |  |
| Manager nodes | Maintain the swarm -> H/A – recommended 3 or 5 -> only one is leader  The Raft Consensus Algorithm |
| Worker nodes | Execute tasks.  Managers are also workers |
| Services | Declarative & Scalable  Ie. docker service create --name web-frontend --replicas 5 |
| Tasks | Atomic unit of work  Assigned to workers  Task can be thought of as a container |
| IANA | Docker engine port : 2375 Secure Engine port : 2376 Swarm port : 2377 |

|  |  |
| --- | --- |
| Building a Swarm | |
| docker swarm init  --advertise-addr {172.31.12.161:2377}  --listen-addr {172.31.12.161:2377} | Best practice to specify IP and listener IP |
| docker swarm join-token manager |  |
| docker swarm join-token worker |  |
| docker node ls | on a manager node list all nodes |
| docker swarm leave | Leaves the swarm |
| docker node promote | Promotes a node to manager |
| docker node demote | Demotes a node to worker |

|  |  |
| --- | --- |
| Services | |
| docker service ps {name} | List services |
| docker service scale {name}=7 | Scale up to 7 shortcut to docker service update --replicas 10 {name} |
|  |  |