Highlighted comment

HARDIK TILWANI 1 month ago (edited)

SUMMARY OF SESSION : 1

~> "Docker" Basics & It's Use Case's.
~> Provisioning (Platform) Of O.S & 4 ways for it.
~> Core Concept of "Containerization" & "Docker".
~> DevOps Principle - How Faster & Agile for "Business Growth".
~> Docker Public Registry (Repository) & "Private Registry".
~> Docker "Host" & "Engine".
~> Commands Used :
docker (ps -a, images, run -it, attach, --name, start, stop, --rm, --help, ls -a -q $( )), Ctrl+p+q, which, rpm -q -f .
~> "Virtualization" vs "Containerization".
~> "Downstream Project".
~> PreConfigured Yum Repository in our new O.S (Container) & installing any software on it.
~> "Platform As a Service".
~> Whatever we do Inside Docker is "Isolated".
~> Old Commands vs Management (New) Commands in Docker.
~> Life of O.S == Life of Program (According to our Requirements)
~> How To Create our own Image :
1- Commit(Copy/Clone to New/Our Image) & 2- Docker File.
~> And Creating Multiple Containers by our Own New Image.
#righteducation IN

Highlighted comment

HARDIK TILWANI 1 month ago (edited)

SUMMARY OF SESSION : 2

~> Configuring "Apache Web Server" inside "Docker".
~> Installing + Deploying Our own Index Page & then how to Start Services Permanently.
~> Docker "Inspect" & "JSON (Parser) Format".
~> "Process Id" == "httpd . pid"
~> "Cloning" Image by "Commit".
~> Multiple Use-Cases of Having "Image", like "Load Balance", "Collaboration", etc.
~> "Virtualization" vs "Containerization" (Multiple Environment).
~> How to Share Our "Image" :
1) Import/Export & 2) Push(Upload)/Pull from Registry.
~> Concept of "Shipping" & "Migration".
~> Concept of "Layering".
~> 3 Type's of Registry :
1) Local , 2) Public & 3)Private.
~> Commands Used -
Docker (container, image, inspect, --format, {Key-Variable}, kill all, commit, push, pull, save -o, load, tag, --network bridge/host), Vim, curl, netstat -tnlp, systemctl start/stop.
~> Created our Own Registry on "hub.docker. com"
#righteducation IN

Highlighted comment

HARDIK TILWANI 1 month ago (edited)

SUMMARY OF SESSION : 3

~> "Networking" In/With Docker.
~> "Docker Files/Logs".
~> IP, Netmask, Network Name, Gateway, Switch, Router, Bridge, Sub-Net.
~> Creating our Own "LAN" (Lab).
~>"DHCP"(DNS Server/Client, IPAM).
~> Network Driver (Default-Bridge Driver).
~> Creating our own Personal Network (Our own LAN) with Bridge Driver (As Default And Will Provide All Features) in Docker And Use Cases of it With it's Importance.
~> Use & Importance of "Isolation" & "Multi-Tenancy" with Containers.
~> "Linking" Concept & Importance of it with it's Availability.
~> Types of Different Network Driver's and there Uses.
~> How we Can Connect/Disconnect a Container from One Network to Another.
~> As an Example of Real Use Case in Industry - Testing Environment (Network) to Production Environment (Network).
~> "Load Balancing" with DNS Server.
~> Example by Creating our own Web Server and Deployed our own Web Page (In PHP) by Enabling Services with our Containers, as a server and client to check.
~> And Cloned(Commit) That Complete Configured Container to our New Image & used it to Create Multiple Containers with Same Data (Same Web Page).
~> With Real Scenario Example of Google, Configured our own Network and Used "Load Balancing" through creating Multiple Containers as Server and Client with DNS.
~> "Alias" Concept by Giving a Unique Name to Our Network (Containers), for Easy Connectivity and Load Balancing.
~> Concept of "LookUp / Resolve".
~> Command's Used -
Docker (logs, -f, run, exec, --name, network, inspect, create, --driver, -dit, route -n, --link, commit, --network alias), PHP(Language), Curl (IP), nslookup, host.
#righteducation IN

Highlighted comment

HARDIK TILWANI 1 month ago (edited)

SUMMARY OF SESSION : 4

~> "Architecture of Docker".
~> Journey of "Docker Technology" - From Bare Metal to Virtualization to "Containerization".
~>"Hypervisor" in Virtualization.
~> Importance of "Container Engine" & Containerization Technology.
~> Concept of Boot Sequence, Kernal, Drivers, Resources, etc.
~> "systemd" Parent (First) Process (pid-1) to start all other Processes.
~> Types of "Network Driver's" -
1) Bridge, 2) Null & 3) Host with Multiple Examples.
~> Concept on Switch, Router, DNS, ISP, LAN,WAN, Packets, etc with Network Driver Examples.
~> Connectivity Rule for Public Network & Private Network.
~> "NAT", "SNAT", "DNAT & "PAT" Concept with its Importance & Use-Cases with Examples.
~> Concept of "Firewalld" & "IpTables".
~> Command's Used -
Docker (uname -r, container, run, -it, --network, -p (port in Router Enable):(port to send Traffic), Bridge, Host), ping (IP), free -m, rpm -q kernal, ps -aux, ps tree.
#righteducation IN

HARDIK TILWANI 3 weeks ago

SUMMARY OF SESSION : 5

~> " IP FORWARDING ".

~> " NAT / PAT " With Successful Practical, "Expose / Exposing".

~> Concept's Such As - "CHAIN"(Forward Chain), " IP TABLES ".

~> "Alias" & "Permanent Alias".

~> "STORAGE" & It's Type's -

1) Ephemeral & 2) Persistent.

~> "VOLUME" - How to Create New Volume? (Volume == Storage).

~> "Centralized Storage".

~> Creating Our "Local YUM".

~> Command's Used -

uname -r, sysctl -a | grep ip_forward, ip tables (-nvL, -F, -p FORWARD ACCEPT), netstat - tnlp, vim, alias, rm -f *.repo, docker (info, inspect, volume(ls, create, inspect,), -v (volume name / path).

#righteducation IN

Highlighted comment

HARDIK TILWANI 1 week ago

SUMMARY OF SESSION : 6

~> Small Project For WebApp /Website Using "WordPress Image".
~> And For Dynamic WebApp We Have To Use Database / Store The Data using "mysql".
~> "Multi-Tier Architecture".
~> Adding "Environment Variable" In Container.
~> "Entry Point".
~> "Linking".
~> Using Variables In WordPress & mysql Containers.
~> "Auto Scaling", "Load Balancer", "Rectify".
~> Done Multi-Tier Setup Manually & Can Be Done By Automation.
~> "Docker Compose" - "Infrastructure As Code".
~> In "docker-compose. yml" File.
~> "Key-Value Pair", "list", "Indent" (For Block Of Statement), "Internal Keywords".
~> "Complete Infrastructure" Created Can Start / Stop / Launch / Remove Perfectly.
~> Command's Used -
docker (pull, run -e, volume, --link, compose, start, stop, up, rm, ps).
#righteducation IN