LXC Commands by CLOUDWWH

SN	LXC Command and Syntax	Use
1	lxc list	List containers
2	lxc image list	List Images
3	lxc launch ubuntu:16.04 container_name	Launching container from Remote
4	lxc exec ubuntu -- /bin/bash	Accessing container Bash Shell
5	lxc stop container_name	Stop the container
6	lxc delete container_name	Delete the stopped container
7	lxc delete --force container_name	Delete a running container
8	lxc image copy images:alpine/3.5 local: --alias image_name	Copy Images from remote to local
9	lxc launch image_name container_name	Launching container from Local
10	lxc exec container_name -- apk update	Command Execution - Repo update
11	lxc file edit container_name/etc/nginx/conf.d/default.conf	Command Execution - Edit File
12	lxc file push index.html container_name/var/www/index.html	File Push from to Container
13	lxc file pull container_name/var/www/index.html /var/www/index.html	Pull file from Container
14	lxc exec web -- rc-update add nginx default	Command Execution - Enable Service
15	lxc exec web -- /etc/init.d/nginx start	Command Execution - Start Service
16	lxc snapshot container_name snapshot_name	Taking Snapshot
17	lxc info container_name	Container Properties
18	lxc restore container_name snapshot_name	Restoring Snapshot
19	lxc copy container_name/snapshot_name new_container	Launch Container from Snapshot
20	lxc image delete image_name	Delete Image
21	lxc remote list	List Remote Public Images Store
22	Host2: (second) lxc config set core.https_address "[::]:8443" lxc config set core.trust_password secret Host1: lxc remote add second 172.31.18.81 --password=secret lxc list lxc list second	Managing second host from the first container host
23	lxc publish container_name/snapshot_name --alias new_imagename	Create new image from the snapshot
24	mkdir webimage lxc image export image_name webimage/ ls -> metadata.yaml rootfs template	Export and verify Image content
25	apt lxd lxd-client zfsutils-linux	Installing LXD and ZFS Utils
26	lxd init	Configuring LXD
27	dpkg-reconfigure -p medium lxd	Reconfigure Neworking part LXD
28	zpool list	List Storage Pools and devices

LXC Commands by CLOUDWWH

29	apt-key adv --recv-keys --keyserver hkp://keyserver.ubuntu.com:80 0xF1656F24C74CD1D8 add-apt-repository 'deb [arch=amd64,i386,ppc64el] http://ftp.utexas.edu/mariadb/repo/10.1/ubuntu xenial main' apt-get update apt-get install mariadb-server -y	Ubuntu key download to Install Maria DB

30	[mysqld]
	binlog_format=ROW
	default-storage-engine=innodb
	innodb_autoinc_lock_mode=2
	bind-address=0.0.0.0
	# Galera Provider Configuration
	wsrep_on=ON
	wsrep_provider=/usr/lib/galera/libgalera_smm.so
	# Galera Cluster Configuration
	wsrep_cluster_name="galera_cluster"
	wsrep_cluster_address="gcomm://x.x.x.x,y.y.y.y"
	
	# Galera Synchronization Configuration
	wsrep_sst_method=rsync
	
	# Galera Node Configuration
	wsrep_node_address="x.x.x.x"
	wsrep_node_name="g1"
		Galera Setup (/etc/mysql/conf.d/galera.cnf)

31	lxc exec g1 -- bash ufw enable ufw allow 3306/tcp ufw allow 4444/tcp ufw allow 4567/tcp ufw allow 4568/tcp ufw allow 4567/udp ufw status exit	Firewall Allow For the Container to run Galera
32	lxc exec g1 -- mysql -u root -p -e "show status like 'wsrep_cluster_size'"	Galera Cluster size check

33	server {
	listen 80 default_server;
	listen [::]:80 default_server;
	
	root /var/www/;
	
	}
		Nginx lxc file edit
		web1/etc/nginx/conf.d/default.conf

LXC Commands by CLOUDWWH

34	upstream lb-web {
	server web1;
	server web2;
	server web3;
	
	}
	
	server {
	listen 80 default_server;
	listen [::]:80 default_server;
	
	location / {
	proxy_pass http://lb-web;
	}
	
	}
		Load Balancer Nginx lxc file edit lb/etc/nginx/conf.d/default.conf

35	iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp --dport 80 -j DNAT --to 10.242.115.38:80	Preroute Host port to Container Port
36	Node1 & 2: apt lxd lxd-client zfsutils-linux bridge-utils Node1: lxd init (select block-device) Node1: lxc launch images:alpine/3.5 container_name Node1: ip link add contgre (GRE Tunnel Name) type gretap remote remote_ip local local_ip ttl 255 Node1: brctl add addif lxdbr0 contgre ((GRE Tunnel Name) Plug In Node1: ip link set contgre up Node2: brctl addbr multibr0 (Create new bridge interface do not use default) Node2: ifconfig multibr0 up Node2: ip link add contgre type gretap remote remote_ip local local_ip ttl 255 Node2: brctl addif multibr0 contgre Node2: ip link set contgre up Node2: lxd init (select block-device and use multibr0 and not lxbr0) Node1: lxc remote add node2 remote_ip:8443 --password=secret Node1: lxc list Node1: lxc list node2 Node1: lxc launch images:alpine/3.5 node2:container_name	Advanced Networking to make twonode containers ping each other and also to control all containers from first node
32	lxc stop c3 lxc move c3 node2:c3 lxc start node2:c3	Container Migration (Offline)