|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Docker** | Independently deployable services into container | | | |
| Containers, isolated environment for application | | | |
| Automated Building and deploying application , Docker Hub - repository | | | |
|  | | | |
| **Command** | Docker run -it -p 8080:80 nginx | | Interactive start the docker |
| Docker run - -p 8080:80 nginx | | Background start the docker |
| docker ps | | list containers |
| docker ps -a | | list all containers which was loaded |
| docker images | | List images |
| docker rm | | delete container |
| docker rmi | | delete image |
| docker rmi -f | | Delete all Images |
| docker rm –f <id\_container | | Delete all Containers |
| docker pull <image> | | Download images |
| docker search <image> | | Search repo into Dockerhub |
| **Kubernetes** | Infrastructure for managing multiple containers | | | |
| Automated scheduling and management of application containers | | | |
| Ecosystem for managing a cluster of Docker containers | | | |
| **Command** | kubectl | | This is command which allows to perform every possible Kubernetes operation. |
| kubectl get nodes | |  |
| kubectl get pods --all-namespaces | |  |
| **Ansible** | Configuration management, and application-deployment tool | | | |
| An **Ansible ad hoc command**  command-line tool to automate a single task | | | |
| Playbook -  playbooks are regularly used to automate [IT infrastructure](https://www.redhat.com/en/topics/cloud-computing/what-is-it-infrastructure) | | | |
| **Command** | ansible all -shell -a "uptime" | | Check date/time into servers |
| ansible all -a "df -h" | | Check storage space |
| Ansible playbook <playbo> | | Playbook executing |
| **Jenkins** | CI process to build an application and package it into isolating container environment | | | |
| GIT | Command |  | |  |
| git init | | Create new repository |
| git clone /path/to/repository | | Create a working copy of a local repository: |
| git add <filename> | | Add one or more files to staging (index): |
| git commit -m "Commit message" | | Commit changes to head (but not yet to the remote repository |
| git push origin master | | Send changes to the master branch of your remote repository |
| git status | | List the files you've changed and those you still need to add or commit |
| git branch | | List all the branches in your repo, and also tell you what branch you're currently in: |
| **Network** | DNS | **DNS (Domain Name System)** is a network protocol used to translate hostnames into IP addresses (nslookup) | | |
| SSH | Cryptographic network protocol for operating network services | | |
| Firewall | A **Firewall** is a network security software that monitors and filters incoming and outgoing network traffic  (Iptables, UFW) | | |
| TCP | The Transmission Control Protocol a connection between client and server is established before data can be sent | | |
| UDP | User Datagram Protocol | | |
| IPv4 | Internet Protocol version 4 (**IPv4**) is the fourth version of the Internet Protocol (IP) (32 bits , 8 bits for each octets) | | |
| **Command** | hostname | finding host/domain name and IP address | |
| ping | test network connection | |
| ifconfig | getting network configuration | |
| ifconfig | Network connections, routing tables, interface statistics | |
| nslookup | query DNS lookup name | |
| telnet | communicate with another hostname | |
| traceroute | outing steps that packets take to get to network host | |
| finger | view user information | |
| telnet | checking status of destination host | |