

# Why DevOps

Sunday, 29 May 2022 11:45 AM

"Over the years, I have gained a wide-ranging set of skills & experience in IT that, I believe, make me supportive, professional as to ensure the company achieve their commercial and financial objectives."

I take pride in my work, I take my professional development seriously.

I always focus on how I can add value to the organization by providing secure and innovative solutions based on the needs of the business.

In addition to possessing solid technical knowledge capabilities,

I have good communication, collaboration, and decision-making skills.

Adding value by

- Frequent feature deployments
- Reduce time between bug fixes
- Reduce failure rate
- Quicker recovery time in case of release failures.

# Different phases in DevOps?

Wednesday, 15 February 2023 12:25 PM

## 4. What are the different phases in DevOps?

A: The various phases of the DevOps lifecycle are as follows:

Plan - Initially, there should be a plan for the type of application that needs to be developed. Getting a rough picture of the development process is always a good idea.

Code - The application is coded as per the end-user requirements.

Build - Build the application by integrating various codes formed in the previous steps.

Test - This is the most crucial step of the application development. Test the application and rebuild, if necessary.

Integrate - Multiple codes from different programmers are integrated into one.

Deploy - Code is deployed into a cloud environment for further usage. It is ensured that any new changes do not affect the functioning of a high traffic website.

Operate - Operations are performed on the code if required.

Monitor - Application performance is monitored. Changes are made to meet the end-user requirements.

### Technical benefits

- Continuous software delivery
- Less complex problems to manage
- Early detection and faster correction of defects

### Business benefits

- Faster delivery of features
- Stable operating environments
- Improved communication and collaboration between the teams

## 6. How will you approach a project that needs to implement DevOps?

The following standard approaches can be used to implement DevOps in a specific project:

### **Stage 1**

An assessment of the existing process and implementation for about two to three weeks to identify areas of improvement so that the team can create a road map for the implementation.

### **Stage 2**

Create a proof of concept (PoC). Once it is accepted and approved, the team can start on the actual implementation and roll-out of the project plan.

### **Stage 3**

The project is now ready for implementing DevOps by using version control/integration/testing/deployment/delivery and monitoring followed step by step.

By following the proper steps for version control, integration, testing, deployment, delivery, and monitoring, the project is now ready for DevOps implementation.

## 7. What is the difference between continuous delivery and continuous deployment?

### **Continuous Delivery**

- Ensures code can be safely deployed on to production
- Ensures business applications and services function as expected
- Delivers every change to a production-like environment through rigorous automated testing

### **Continuous Deployment**

- Every change that passes the automated tests is deployed to production automatically
- Makes software development and the release process faster and more robust
- There is no explicit approval from a developer and requires a developed culture of monitoring

# Agile Methodology

Saturday, June 27, 2020 6:08 PM

## What is Agile?

Business + Development

Business - What to build and what are the requirements.

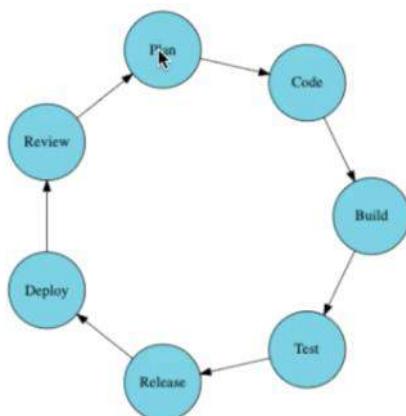
Development - responsible to build a product (Design, Coding, Testing, Deploying)

Product Owner:

- Always available for agile team.
- Agile team understand the business objectives clearly

Agile - One Team - Build software in small iteration called Sprint.

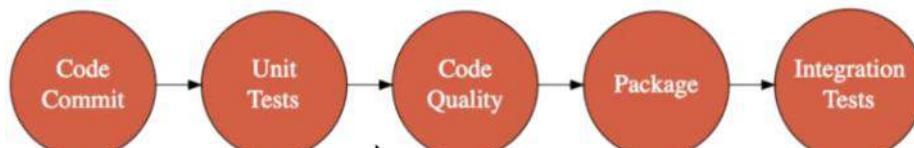
Each sprint contain



## *Agile - Each Iteration*

- Agile is Set of values & Principles
- It is used to develop general specific software or tool
  - Microsoft Office
  - Android
  - Tally

## Agile Automation: Continues integration





## *IAAC - Automate Operations*

Scrum Model - Master

Sprint > Duration - 2 Weeks - 10 WDs

Team Availability - 5 Developers

- 2 DevOps

100% TV (Team Velocity) All members available

8 Features

### **How Agile Works?**

- Build Short
- Build Often
- **This is the Methodology of Agile**

Releasing new software

Then taking reviews from customers that what implementation is required.

Then upgrading the software by modification and implementation launch new release.

Again

Taking reviews by customer whether customer is satisfied.

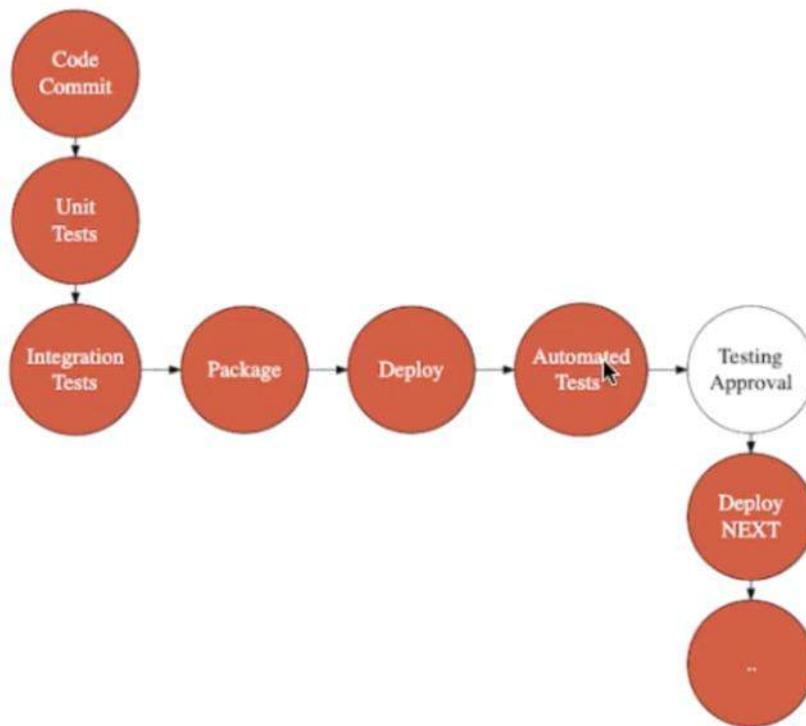
Upgrade and launch & upgrade again and launch

### **Principle of Agile.**

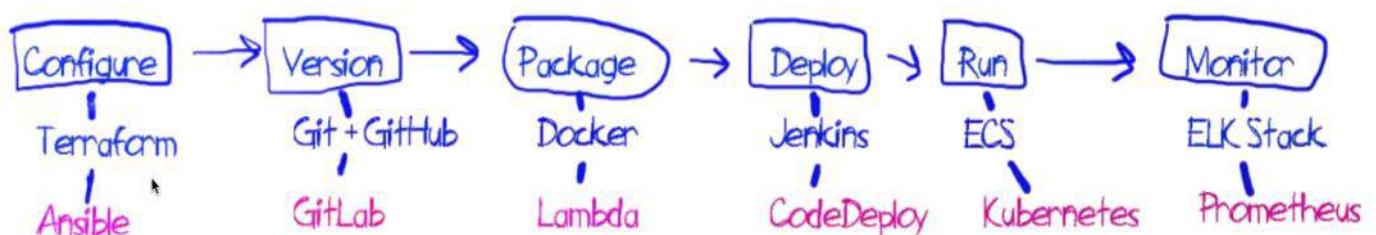
Gives your ability to take good decision in specific situation

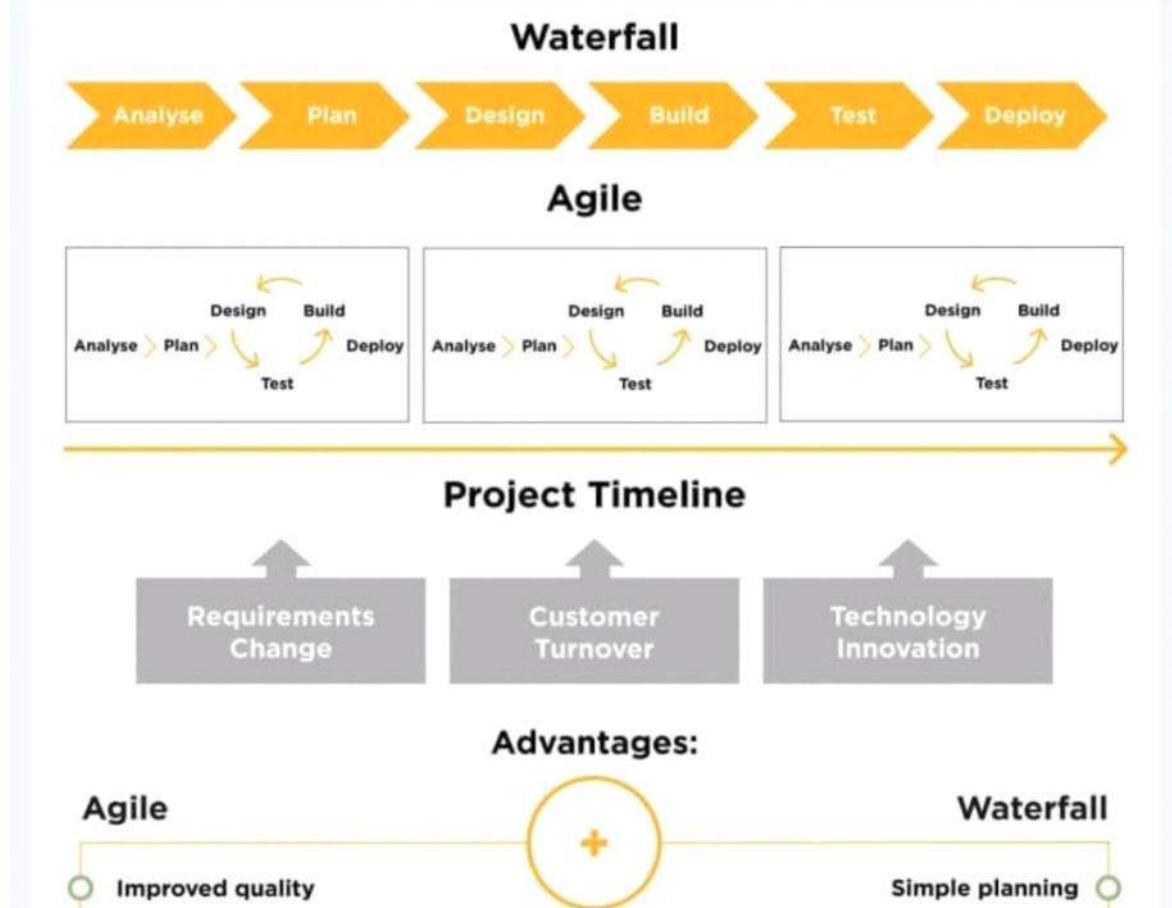
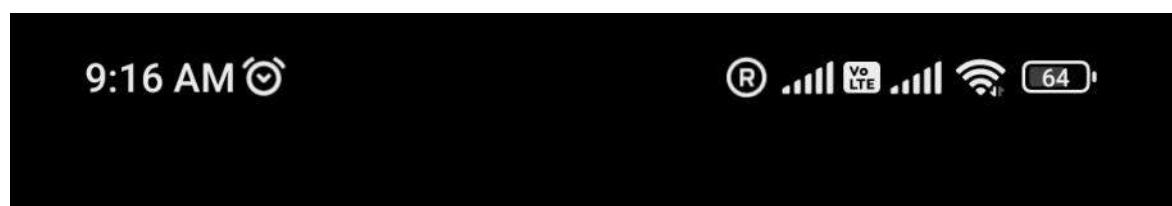
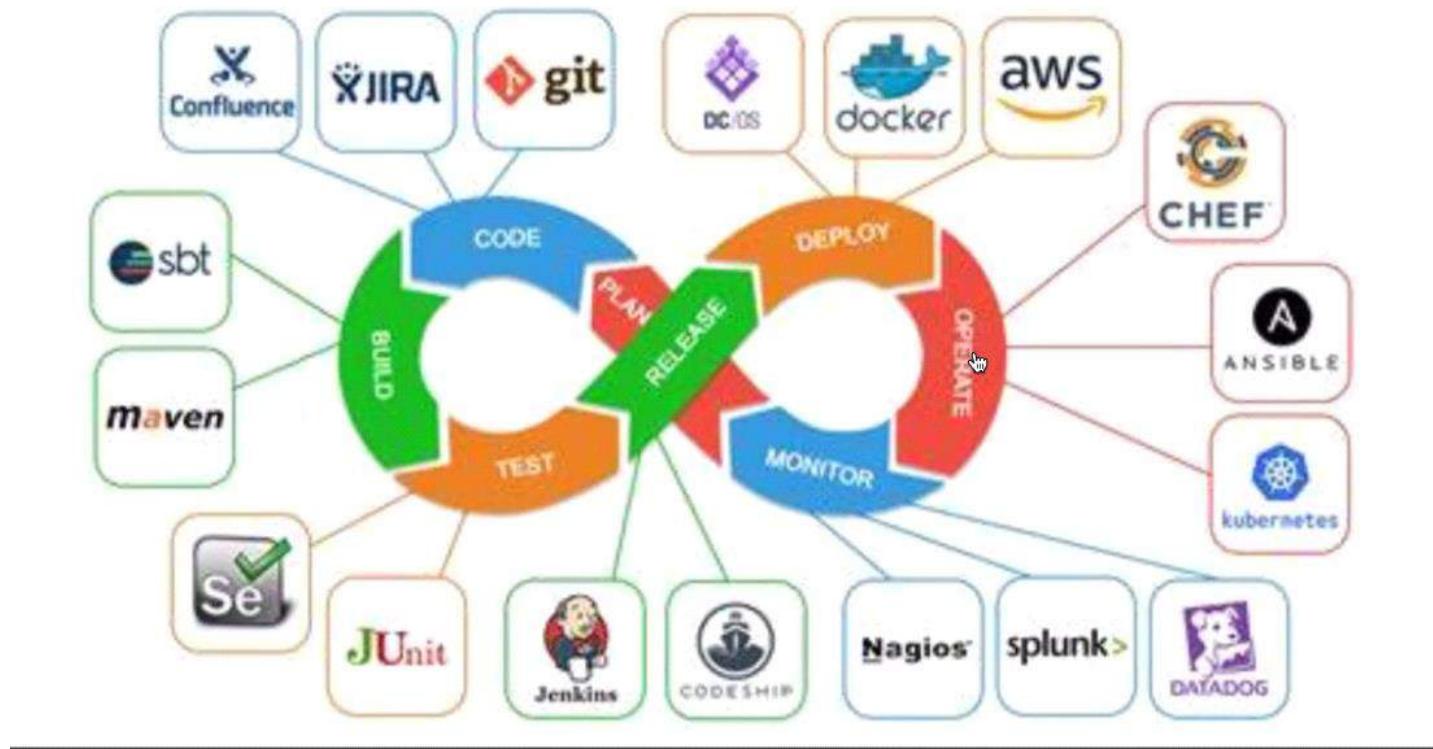
- Customer satisfaction is first priority through early and continuous delivery or software

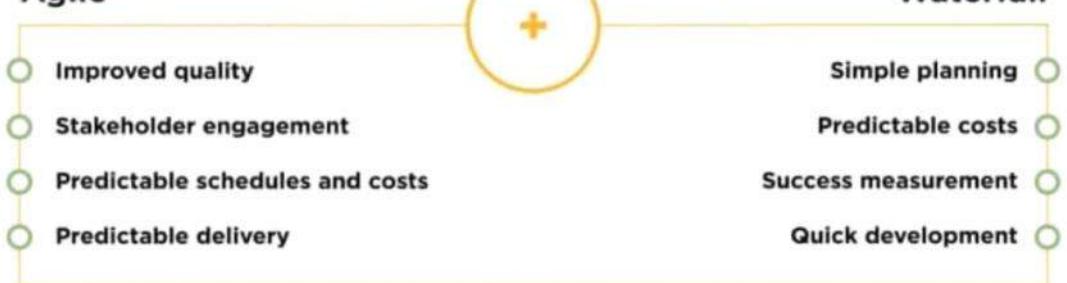
- Say no to requirement, Welcome changing even late development.
- Deliver work frequently
- Business people & developers must work together with project
- Software created by any specific team in Agile should fully support
- Until project finished Business people & developers must work together



## Continuous Delivery







### Disadvantages:



### When to choose?



# 6 phases

Sunday, 29 May 2022 12:01 PM

## Planning

- Understanding of the project to ultimately develop the best product
- It requires various inputs for the development and operation phases.
- It helps to gain clarity regarding the project development and management process.

## Development

- Where the project is built by developing system infra, developing features by writing codes, test cases and automation process.
- Developers store their codes in remote repo

## Continuous Integration (CI)

- Automation of code validation, build and testing to ensure changes are made properly.
- Using Jenkins tools.

## Deployment

- Using many tools and scripts to automate the process for feature activation release.
- Used Cloud services for infra

# Blue Green Deployment

Monday, June 29, 2020 10:22 AM

## Zero Downtime Deployments.

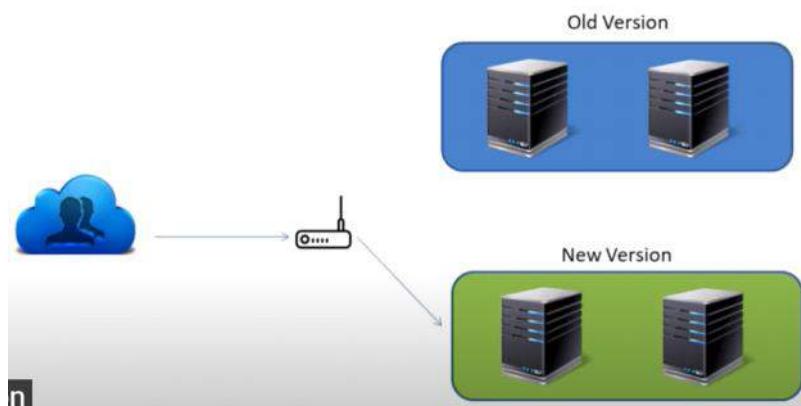
It's technique where we can reduce the downtime of the application.

So let's say you deploy software and that currently running on active production traffic will be called [Blue Environment](#) (Old Version).

Then you want to release the new software version and started deploying on non-Prod environment Call [Green Env](#) (New Version).

Once you are satisfied with new version you can switch production traffic from Blue Old env to Green new env Via Router.

So all the traffic will directed to new version and it will become a Production active version.



## Advantages

- Continuous Deployment and Fast Rolling Back
- You can test the update on the new server before it is made live.
- No down time when switching to a new version.
- You can easily and quickly roll back to the previous version if something doesn't work with the new version.

## Disadvantages

- High Cost
- More Overheads

# DEV / UAT / PROD

Saturday, September 11, 2021 4:22 PM

## Development

- Developers and programmers use the development server to test code directly.
- Most developers and programmers will have development environments set up for the work
- Where they can build and verify the work they are doing.
- This server is usually set up with the needed hardware, software and other necessary parts for debugging and deploying.

## Test

- From Dev Env they will deploy that finished work to a test server.
- This is used for work verification.
- The internal team completes the testing phase, usually with the use of a QA Tester.
- The tester will run used cases to ensure that the application is functioning as it should.
- If there is any bugs or other issues, they will create tasks for the developers or programmers to fix

## User Acceptance Testing (UAT)

- When work passes through the internal testing phase and is ready for approvals,
- The team deploys it to a UAT server where customer is also involved.
- Once in the server the work will get final client approvals before flipping the switch.
- The fundamental difference between a UAT and Test server is that UAT is configured to run as a production build.
- But the database is separate where it usually doesn't include caching and other configurations to handle scale.
- Customer will be using this UAT. It will function similar to how it would if it were in production.
- Customer will check if it is working accordingly or required any changes.
- This can speed up approvals and limit problems that may occur after going live.

## Staging

- Staging is for pre-deployment.
- A staging server's set up is like production with all production configurations and the team uses it to perform smoke testing.
- This ensures the code and everything works in a production configuration and architecture.
- Once the UI developer, backend developer, DevOps, and database administration check everything and it's an all-go from each, they push the code to production.

## Demo

- There may be a request from a client to set up a demo server.
- A demo server is essentially a frozen version of a production server that is usually a few deployments behind the production.
- When you complete final work and the client approves it, it may be deployed to a demo server.
- A sales and marketing team will usually use this server to promote the product and allow prospects and leads to interact with the product.

## Production

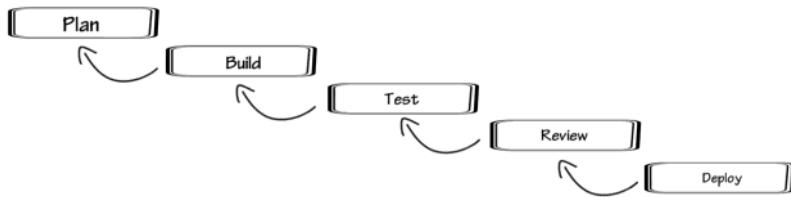
- Production servers are the final location for all finished and approved work.
- When you deploy code to a production server, this means everyone has approved it to go live ready for use
- Working code should only be deployed to a production server after it has been tested and approved for going live.
- Work should never be done on a production server without the use of some type of version control as this will be a high risk for things breaking while the product is in use.
- In certain situations, when a product goes offline or a production server goes down, it can cost a company a lot of money and this is definitely not something anyone wants to occur.

# Scrum & Waterfall

Saturday, June 27, 2020 5:49 PM

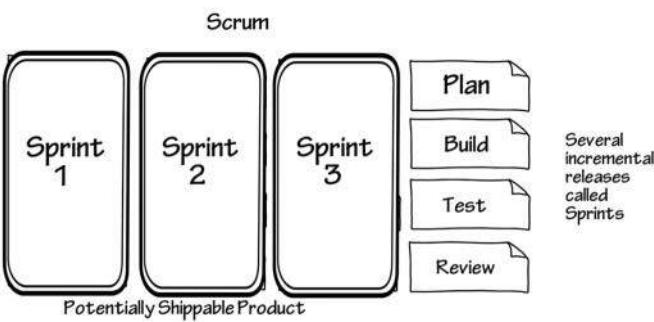
## Waterfall Cycle

- Plan
- Build
- Test
- Review
- Deploy



## Scrum

The process is broken into smaller pieces first, that's call sprint



## 3 Roles

- Three key role for framework to work well first.
- 1. Product Owner - Person responsible for defining the features that are needed in the product. Have bright ideas that turn into the product
- 2. Scrum Master - Servant leader to the team and responsible for protecting the team and process running the meetings and keeping things going. The team of Developers testers writers.
- 3. Team Members - The team works to get product done.

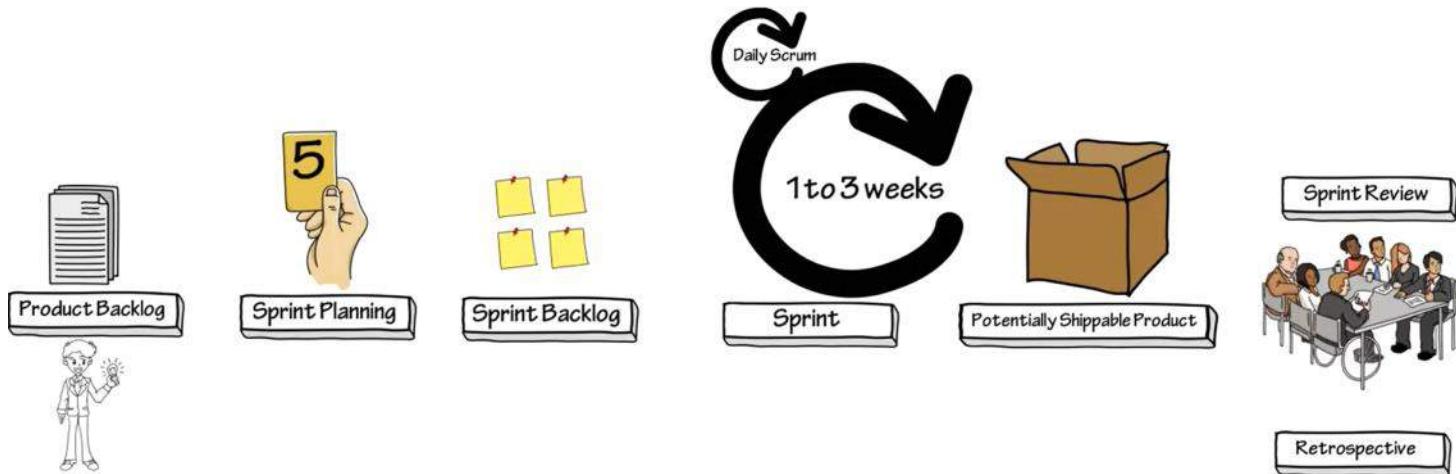
Product Backlog

Sprint Backlog

Burndown Chart

3 Ceremonies

- Sprint Planning
- Daily Scrum
- Sprint Review



Repeat this workflow for each sprint

# SDLC

Thursday, June 25, 2020 6:20 PM

## Software Development Life Cycle



### 1. Identify the Current Problems

“What are the current problems?” means getting input from all stakeholders, including customers, salespeople, industry experts, and programmers. Learn the strengths and weaknesses of the current system with improvement as the goal.

### 2. Plan

“What do we want?” In this stage of the SDLC, the team determines the cost and resources required for implementing the analyzed requirements. how they can implement the project successfully with the lowest risk in mind.

### 3. Design

“How will we get what we want?” This phase of the SDLC starts by turning the software **specifications into a design plan called the Design Specification**. All stakeholders then review this plan and offer feedback and suggestions. It’s crucial to have a plan for collecting and incorporating stakeholder input into this document. Failure at this stage will almost certainly result in cost overruns at best and the total collapse of the project at worst.

### 4. Build

“Let’s create what we want.”

The actual development starts. It’s important that every developer sticks to the agreed blueprint. Also, make sure you have proper guidelines in place about the code style and practices.

### 5. Code Test

“Did we get what we want?” In this stage, we test for defects and deficiencies. We fix those issues until the product meets the original specifications.

In short, we want to verify if the code meets the defined requirements.

### 6. Software Deployment

“Let’s start using what we got.”

At this stage, the goal is to deploy the software to the production environment so users can start using the product. However, many organizations choose to move the product through different deployment environments such as a testing or staging environment.

This allows any stakeholders to safely play with the product before releasing it to the market. Besides, this allows any final mistakes to be caught before releasing the product.

# Maven

Wednesday, October 20, 2021 4:11 PM

## Apache maven technology

Project management tool - based on POM (Project object model), For project Build

Simplifies the build process

- It makes project is to build
- It provides uniform build process
- Provides project information (log document, cross referenced sources, mailing list, dependency list, unit test reports)
- Easy to migrate for new features of Maven

## Build Tool - building a process

- Generates source code (if auto-generated code is used)
- Generates documentation from source code
- Compile the source code
- Packages compiled code into JAR or ZIP file
- Installs the packaged code in local repository, server repository, or central repository

# Failed deployment

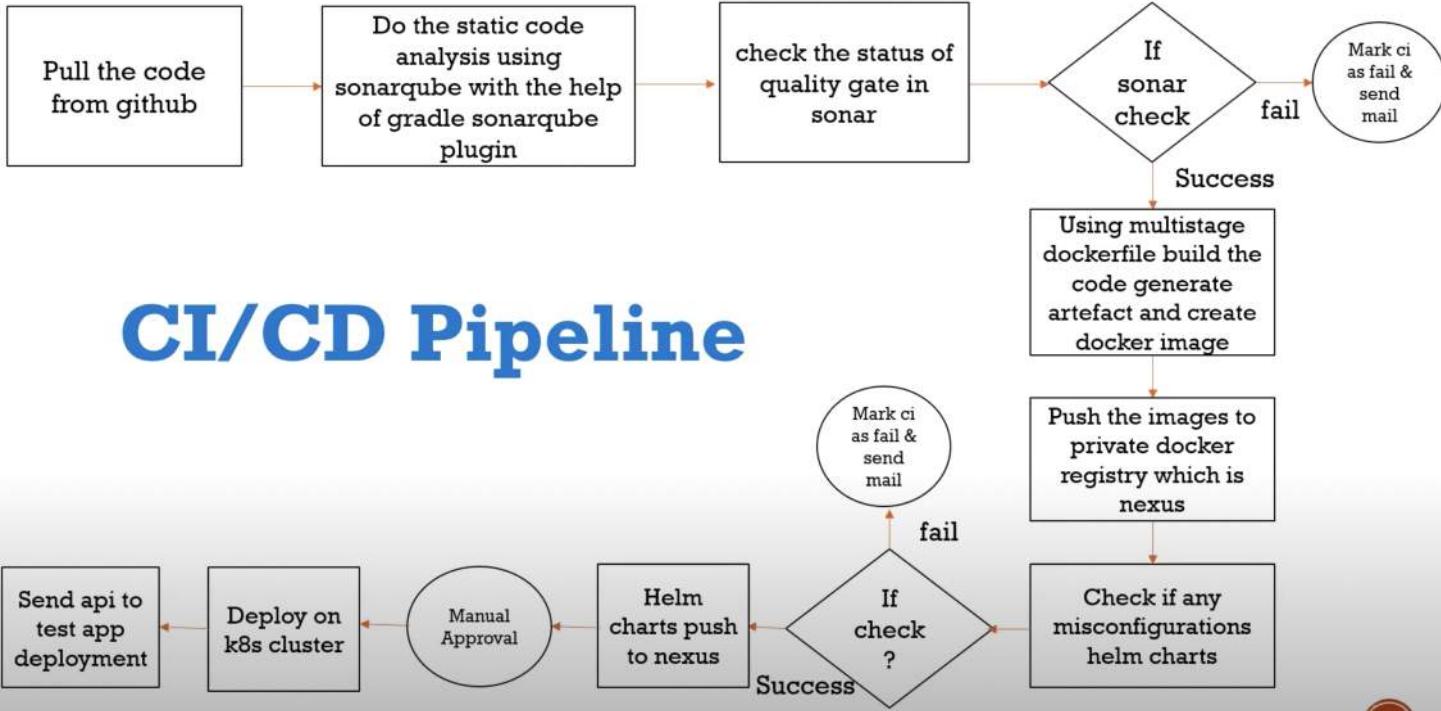
Tuesday, 25 October 2022 3:19 PM

I treat failures (such as outages) as opportunities to learn

- Automate Code testing
- Use docker for same environment
- Use Microservices
- Overcome risks to avoid failure

# CI/CD Pipeline

Monday, 20 February 2023 12:43 PM



**CI/CD**

- CI means **Continuous Integration** and CD means **Continuous Delivery** and **Continuous Deployment**.
- This is how SDLC works for dev-ops
- A **CI/CD pipeline** helps you automate steps in your software delivery process, such as **initiating code builds**, running **automated tests**, and **deploying to a staging or production environment**

**Continuous Integration**

*Continuous Integration is a development practice that requires developers to integrate code into a shared repository several times a day.*

**Continuous Delivery**

*Continuous delivery is a software development practice where software can be released to production at any time.*

**Continuous Delivery**

&amp;

**Continuous Deployment**

*Continuous Delivery is a software development practice where software can be released to production at any time.*

*Continuous Deployment is a software development practice where software is automatically released to production continuously.*

**Cardinal Principles of Continuous Integration**

A single central repository where the code lives.

Developers check-in/commit their code frequently.

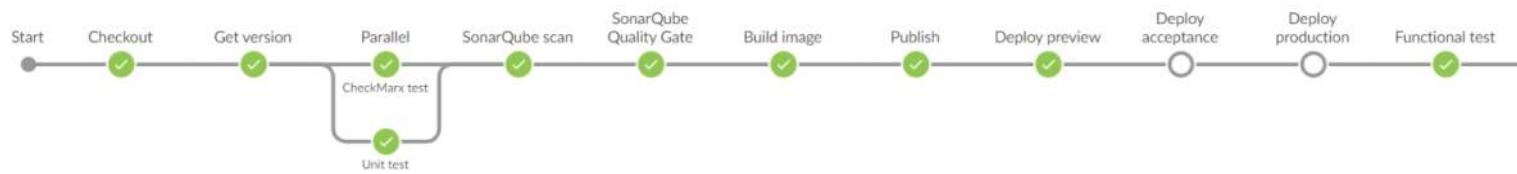
Build should be triggered every time a developer checks in code.

Build should be automated and fast.

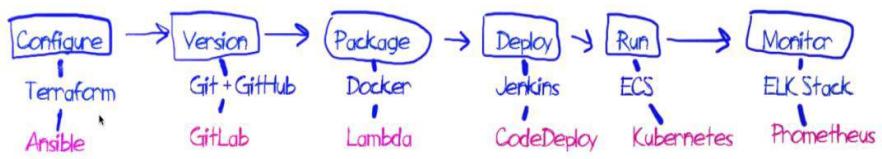
Build should compile the code as well as run automated.

Fixing a failed build should be top priority for the developers.

Build results should always be communicated to all developers.



Daily standup: 30 mins  
 every week: refinement session  
 sprint planning: every 2 weeks  
 sprint duration: 2 weeks  
 Retro: with sprint end



## Continuous Delivery

*Continuous delivery is a software development practice where software can be released to production at any time.*

### Old School Operations vs Continuous Delivery (CD)

Pain Point #1: Correctness of Instructions.

CD: Correctness of automated scripts can be verified at creation time.

Pain Point #2: Difference in instructions across environments.

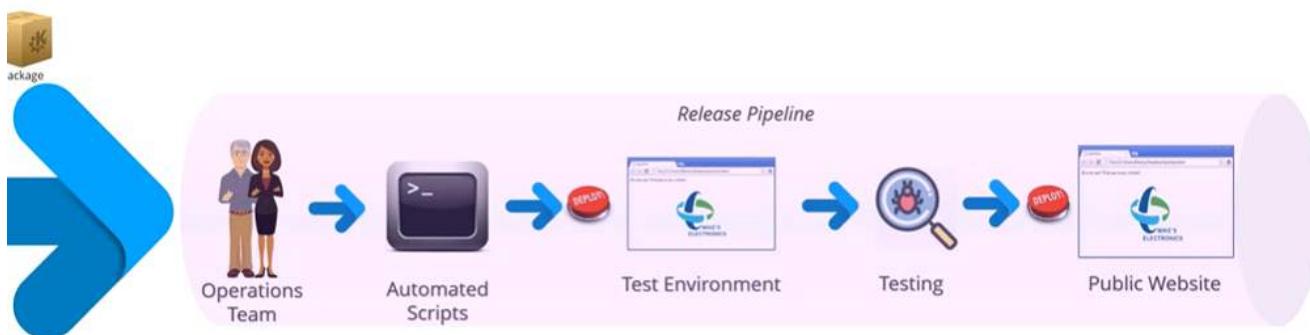
CD: Automated scripts can easily pick the tasks for each environment.

Pain Point #3: Error prone nature of manual tasks.

CD: Automation prevents the occurrence of human errors.

Pain Point #4: Deployments are sophisticated, high-impact with downtime.

CD: Automated deployments, easily repeatable, lesser time-to-market.



## Continuous Delivery

&

## Continuous Deployment

*Continuous Delivery is a software development practice where software can be released to production at any time.*

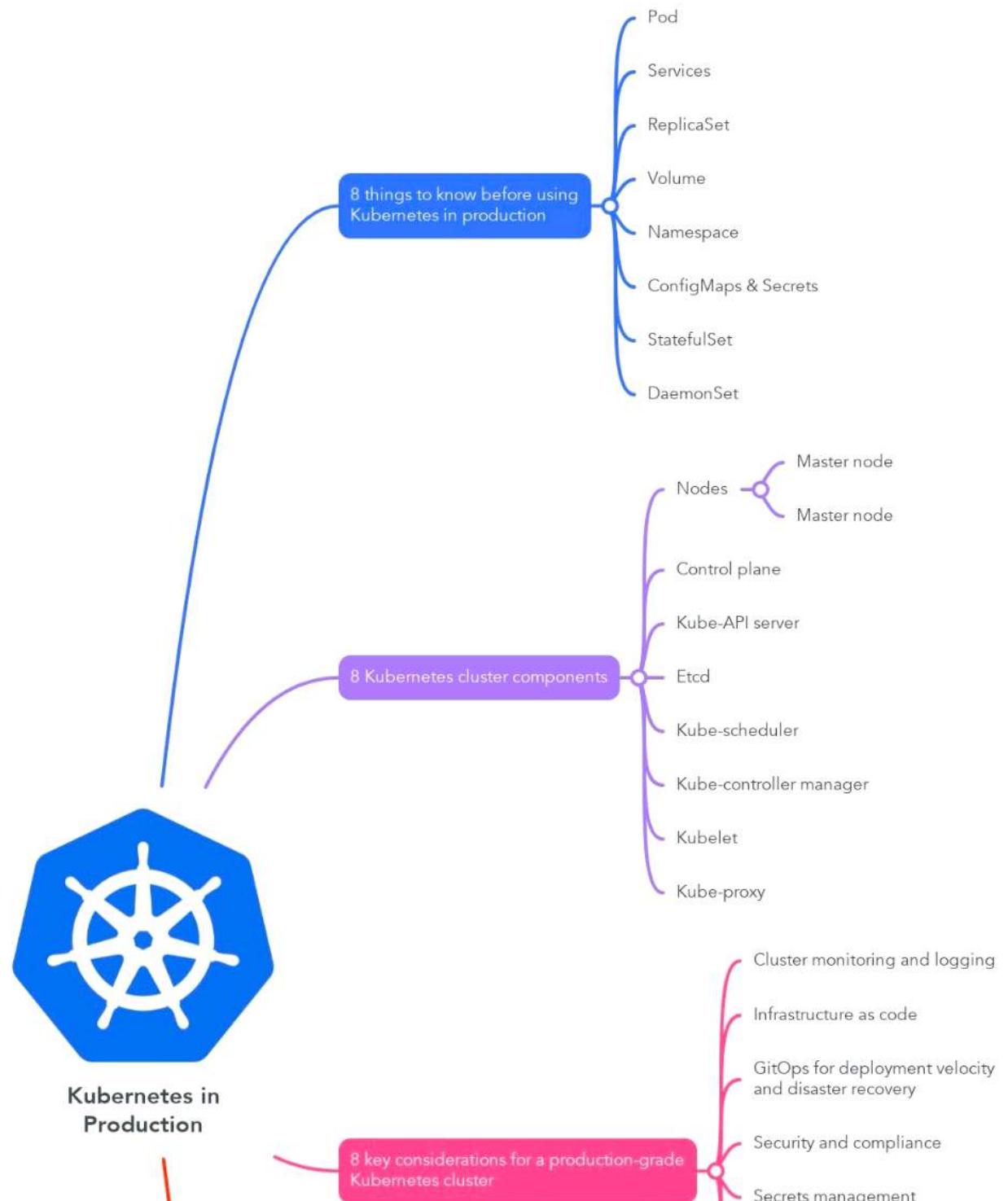
*Continuous Deployment is a software development practice where software is automatically released to production continuously.*

# Kubernetes

Monday, 8 May 2023 12:20 AM

11:48 AM ☀️ 🌃

Bluetooth R 31%



### 8 key considerations for a production-grade Kubernetes cluster

- Security and compliance
- Secrets management
- Self-hosted vs. managed Kubernetes
- Centralized Ingress
- Portability and scalability

### 8 best practices for managing a Kubernetes cluster in production

- Use the latest Kubernetes version
- Enforce resource isolation using namespaces
- Implement role-based access control
- Enforce version control for resource manifests
- Use labels and selectors to simplify resource management
- Set resource requests and limits for pods
- Use minimal images
- Perform regular log audits



# Kubernetes questions

Tuesday, 16 May 2023 12:09 AM

## Kubernetes Architecture:

Kubernetes follows a master-worker architecture where the **master node manages the cluster and worker nodes** run the application containers.

### Master Components:

- **API Server:** Acts as the central control point and exposes the Kubernetes API.
- **etcd:** A distributed key-value store that stores the cluster's configuration data.
- **Scheduler:** Assigns containers to nodes based on resource availability and scheduling policies.
- **Controller Manager:** Manages various controllers responsible for maintaining desired cluster state.

### Node Components:

- **Kubelet:** Runs on each node and manages containers, ensuring they are running and healthy.
- **Container Runtime:** Executes and manages containers (e.g., Docker, containerd).
- **kube-proxy:** Enables network communication between services and pods.

### Additional Components:

- **Ingress Controller**  
Handles incoming traffic to services from outside the cluster.
- **DNS Server**  
Provides DNS-based service discovery within the cluster.
- **Metrics Server:**  
Collects resource utilization metrics for nodes and pods.

## 1. What is Kubernetes and what are its main features?

Possible answer: Kubernetes is an open-source platform for automating deployment, scaling, and management of containerized applications. Its main features include container orchestration, automatic scaling, self-healing, service discovery and load balancing, rolling updates, and health checks.

## 2. What is a **Kubernetes Pod** and how is it **different from a container**?

A Kubernetes Pod is the smallest **deployable unit in Kubernetes** and represents a **single instance of a running process in a cluster**.

A Pod can contain one or more containers that share the same network namespace, IP address, and storage volumes.

While a **container is an isolated and lightweight runtime environment for an application**,

A Pod provides a higher-level abstraction that allows multiple containers to work together as a cohesive unit.

## 3. What is a **Kubernetes Service** and how does it provide network connectivity to Pods?

A Kubernetes Service is an abstraction that defines a logical set of Pods and a policy for accessing them.

It provides a stable IP address and DNS name for a set of Pods, and allows other components in the cluster to discover and communicate with them.

A Service uses a label selector to match the Pods it targets and can perform load-balancing and proxying of network traffic to them.

## 4. What is a **Kubernetes Deployment** and how does it manage the lifecycle of Pods?

- A Kubernetes Deployment is a higher-level abstraction that manages the lifecycle of a set of Pods.
- It ensures that a specified number of replicas are running at any given time,
- Can perform rolling updates and rollbacks of the Pods.
- A Deployment uses a ReplicaSet to create and scale the Pods
- Provides declarative configuration and automatic self-healing capabilities.

## 5. How do you **monitor and troubleshoot** a Kubernetes cluster?

There are several tools and approaches for monitoring and troubleshooting a Kubernetes cluster, including:

- API objects such as Pods, Services, and Deployments to gather metrics and logs
- Dashboard, a web-based UI for visualizing and managing the cluster
- Kubernetes health checks and probes to detect and handle failures
- Kubernetes Operators, custom controllers that automate common operational tasks
- External tools such as Prometheus, Grafana, and Fluentd to collect and analyze metrics and logs



# Best Practice

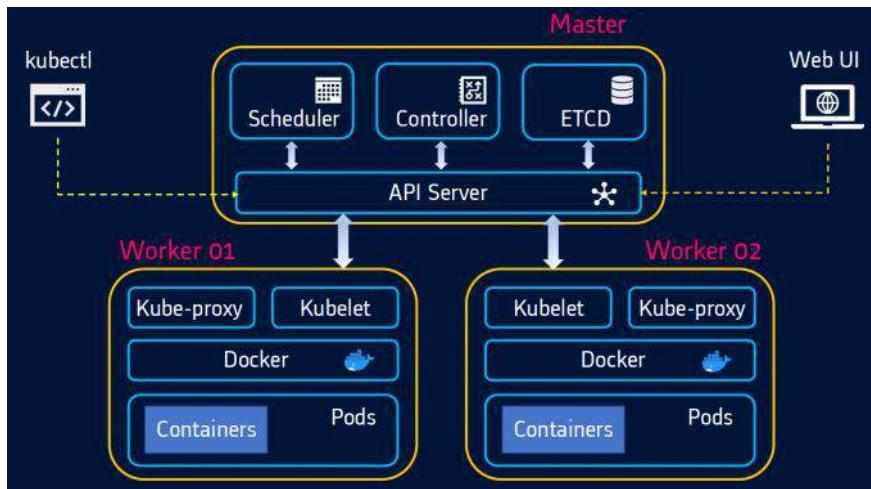
Wednesday, 17 May 2023 1:52 PM

## Kubernetes Best Practices:

- Define and use namespaces to logically segregate resources.
- Use labels and selectors to organize and group resources.
- Leverage deployments for managing application lifecycle and rolling updates.
- Configure resource requests and limits to ensure optimal resource allocation.
- Utilize readiness and liveness probes to ensure application health.
- Implement horizontal pod autoscaling (HPA) to scale based on resource usage.
- Store application configuration in ConfigMaps or Secrets, not in container images.
- Use persistent volumes (PV) and persistent volume claims (PVC) for data storage.
- Implement health checks and readiness probes for application reliability.
- Regularly monitor and analyze cluster and application metrics.

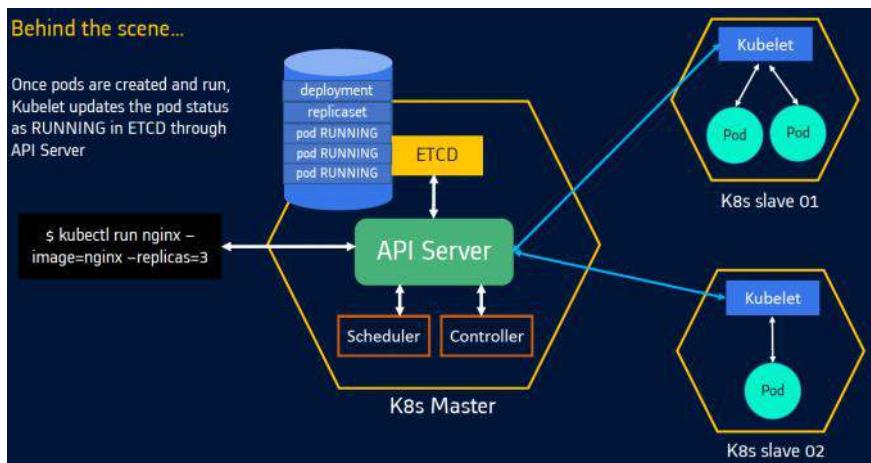
# K8s Architecture

Friday, 14 April 2023 6:27 PM



## Behind the scene

- When a command is given through `kubectl`
- API Server updates the deployment details in ETCD
- Controller manager through API server identifies its workload and creates a ReplicaSet.
- ReplicaSet creates required number of pods and updates the ETCD
- Scheduler identifies it's workload through API-Server and decides the nodes onto which the pod are to be scheduled.
- Pods are assigned to the node
- Kubelet identifies it's workload through API-Server and understands that it needs to deploy some pods on its node
- Kubelet instruct the docker daemon to create the pods and update the ETCD through API server.



# Master Components

Friday, May 6, 2022 12:30 AM

## Master

- Master responsible for managing the complete cluster.
- Responsible for the orchestration of containers on worker nodes
- Can be more than 1 master
- Manage the scheduling and deployment of container.

### - Main components

- ETCD
- Scheduler
- Controller
- API Server

## ETCD

- Etcd backend data store for Kubernetes cluster
  - All data relating to the state of the cluster
  - Port 2379, Etcdctrl client
  - Etcd stores all info on all the nodes in cluster
- Etcd Roles in K8s
- Every information from kubectl cmd is from Etcd.
  - Nodes, PODs, Configs, Secrets, Accounts, Roles, Bindings, Others, ReplicaSet

## Scheduler

- Handle the scheduling & distributing work or containers across multiple nodes.
- Looks for newly created containers and assigns them nodes.
- The process of selecting an available node in the cluster on which to run container.

## API Server manager

- Master communicate with rest of the cluster through the kube-apiservers.
- Validate and executes user's REST commands.
- Also validate the etcd configuration

## Controller Manager

- Brain behind Orchestration.
- Responsible for noticing when nodes, containers or endpoints goes down.
- It makes decisions to bring up new containers.
- Runs control loops that manage the state of cluster by checking nodes are running in cluster.

## Kubectl

- Command line utility
- Used to deploy and manage application on Kubernetes.

### **Cloud-controller-manager**

- Provides an interface between Kubernetes and various cloud platforms.
- Only used when using cloud-based resources alongside Kubernetes.

# Docker vs Kubernetes

Wednesday, 12 April 2023 11:41 AM

## Docker:

- Containers, isolated environment for application.
- Automated building and deploying application - Process of application.
- It's container platform for configuring, building and distributing containers.
- Docker is used in local development process.
- CI - Build docker images
- Pushed to private repository

## Kubernetes:

- Infrastructure for managing multiple containers in multi nodes.
- After container deployed - Automated scheduling and management of application containers
- Ecosystem for managing a cluster of docker containers.
- Cluster > K8s engine > K8s Cluster
- K8s engine spine on multi virtual or physical servers
- Engine creates one unified cluster
- Where docker Containers are running
- K8s service which enable docker to run in the cluster is kubelet
- Each Node has 1 kubelet installed

## Docker Swarm:

- Docker swarm is an alternative to kubernetes.
- Docker Swarm is also container orchestration tool
- Docker daemons run on each node instead of kubernetes engine

Kubernetes	Docker Swarm
Complex installation	Easier installation
More complex with a high learning curve, but more powerful	More lightweight and easier to use, but limited functionality
Supports auto-scaling	Manual scaling
Built in monitoring	Needs third party tools for monitoring
Manual setup of load balancer	Auto load balancing
Need for a separate CLI tool	Integrates docker CLI

# Worker Kubelet

Thursday, 26 May 2022 10:30 PM

## Kubelet

- Responsible for interacting with master to provide health information of the worker node.

## Kube Proxy

- Network traffic is routed properly to internal and external services as required.
- Rules defined by network policies in kube-controller manager and other custom controllers.

- Register node
- Create Pods
- Monitor Node & PODs

Kubeadm does not deploy kubelets

Manually install kubelet on worker nodes

```
Ps -aux | grep kubelet
```

## Kublet

- The process that actually schedules those pods and container underneath is kubelet
- Kublet interacts with both the container and the Node.
- Its responsible for taking that configuration.
- Kublet starts the pod with container inside then assigning resources from that node to container like cpu ram and storage resources.

# Basic Kubectl Command

Saturday, May 7, 2022 2:08 PM

## **Get nodes**

```
$ kubectl get nodes
```

## **Cluster info**

```
$ kubectl cluster-info
```

```
$ kubectl cluster-info dump --output-directory=/path/cluster
```

## **Get Pods**

```
$ kubectl get pods
```

## **Get Services**

```
$ kubectl get service
```

## **Get help**

```
$ kubectl create -h
```

## **Create nginx deployment**

```
$ kubectl create deployment nginx-depl --image=nginx
```

## **Get deployment**

```
$ kubectl get deployment
```

## **Get replicaset**

```
$ kubectl get replicaset
```

## **Edit deployment**

```
$ kubectl edit deployment nginx-depl
```

## **Check pod**

```
$ kubectl get pod
```

## **Check replicaset**

```
$ kubectl get replicasetet
```

## **Check logs**

```
$ kubectl logs nginx-depl-5c9d9cc4c
```

## **Run mongo**

```
$ kubectl describe pod mongo-depl-85ddc6d66-x5q4d
```

## **Create status**

```
$ kubectl create deployment nginx-depl --image=nginx
```

## **Describe service**

```
$ kubectl describe service nginx-service
```

## **Edit Deployment**

```
$ kubectl edit deployment nginx-depl
```

## Delete Deployment

```
$ kubectl delete deployment nginx-depl
```

## Deployment Yaml for nginx

```
[minikube@docker ~]$ kubectl get pod
No resources found in default namespace.
[minikube@docker ~]$ cat nginx-depl.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-depl
  labels:
    app: nginx
spec:
  replicas: 2
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.20
          ports:
            - containerPort: 80
```

## Apply yaml deployment

```
$ kubectl apply -f nginx-depl.yaml
```

## Check status

```
$ kubectl get deployment
$ kubectl get pod
```

## More information on Pod

```
$ kubectl get pod -o wide
```

## Status Auto output (Debug)

```
$ kubectl get deployment nginx-depl -o yaml > nginx-depl-output
```

## Delete Deployment

```
$ kubectl delete -f nginx-depl2.yaml
```

## Describe Service

```
$ kubectl describe service mongodb-service
```

## Namespaces

```
# kubectl get namespaces
# kubectl get pods --namespace kube-system
# kubectl get pods --all-namespaces
```

**Create Namespace**

```
# kubectl create namespace dev
```

**Draining Node (Maintenance)**

```
# kubectl drain <node name> --ignore-daemonsets --force
```

**Uncordon Node**

```
# kubectl uncordon <node name>
```

**Delete Deployment**

```
# kubectl delete deployment my-deployment
```

**Run nginx pod**

```
# kubectl run nginx --image nginx
```

**Create manifest file**

```
# kubectl run redis --image=redis123 --dry-run=client -o yaml > redis-definition.yaml
```

**Delete Node**

```
kubectl get nodes
```

```
kubectl drain <node-name>
```

```
kubectl drain <node-name> --ignore-daemonsets --delete-local-data
```

```
kops edit ig nodes
```

```
kubectl delete node <node-name>
```

```
kubectl get nodes -o wide
```

# Commands

Monday, 13 February 2023 11:05 PM

1. docker run -p 8080:8080 in28min/hello-world-rest-api:0.0.1.RELEASE
2. kubectl delete pod hello-world-rest-api-58ff5dd898-62l9d

## DEPLOYMENT

3. kubectl create deployment hello-world-rest-api --image=in28min/hello-world-rest-api:0.0.1.RELEASE
4. kubectl expose deployment hello-world-rest-api --type=LoadBalancer --port=8080
5. kubectl scale deployment hello-world-rest-api --replicas=3
6. kubectl autoscale deployment hello-world-rest-api --max=10 --cpu-percent=70
7. kubectl edit deployment hello-world-rest-api #minReadySeconds: 15
8. kubectl set image deployment hello-world-rest-api hello-world-rest-api=in28min/hello-world-rest-api:0.0.2.RELEASE
9. kubectl apply -f deployment.yaml
10. kubectl apply -f ./currency-conversion/deployment.yaml

## CLOUD

11. gcloud auth login
12. gcloud container clusters get-credentials in28minutes-cluster --zone us-central1-a --project solid-course-258105

## GET / SET

13. kubectl set image deployment hello-world-rest-api hello-world-rest-api=DUMMY\_IMAGE:TEST
14. kubectl set image deployment hello-world-rest-api hello-world-rest-api=in28min/hello-world-rest-api:0.0.2.RELEASE
15. kubectl get componentstatuses
16. kubectl get pods --all-namespaces
17. kubectl get events
18. kubectl get pods
19. kubectl get replicaset
20. kubectl get replicaset
21. kubectl get deployment
22. kubectl get service
23. kubectl get pods -o wide
24. kubectl get rs
25. kubectl get rs -o wide
26. kubectl get events --sort-by=.metadata.creationTimestamp
- 27.
28. kubectl set image deployment hello-world-rest-api hello-world-rest-api=in28min/hello-world-rest-api:0.0.2.RELEASE
- 29.
30. kubectl get deployment hello-world-rest-api -o yaml
31. kubectl get deployment hello-world-rest-api -o yaml > deployment.yaml
32. kubectl get service hello-world-rest-api -o yaml > service.yaml
33. kubectl apply -f deployment.yaml
34. kubectl get all -o wide
35. kubectl get svc --watch
36. kubectl get componentstatuses
37. kubectl get pods --all-namespaces

38. kubectl get pods --all-namespaces
39. kubectl get pods --all-namespaces -l app=hello-world-rest-api
40. kubectl get services --all-namespaces
41. kubectl get services --all-namespaces --sort-by=.spec.type
42. kubectl get services --all-namespaces --sort-by=.metadata.name

## ROLLOUT

43. kubectl rollout history deployment hello-world-rest-api
44. kubectl set image deployment hello-world-rest-api hello-world-rest-api=in28min/hello-world-rest-api:0.0.3.RELEASE --record=true
45. kubectl rollout undo deployment hello-world-rest-api --to-revision=1

## K8s Info

46. kubectl version
47. kubectl cluster-info
48. kubectl cluster-info dump
49. kubectl logs hello-world-rest-api-58ff5dd898-6ctr2
50. kubectl logs -f hello-world-rest-api-58ff5dd898-6ctr2
51. kubectl explain pods
52. kubectl describe pod hello-world-rest-api-58ff5dd898-9trh2
53. kubectl top node
54. kubectl top pod
55. kubectl diff -f deployment.yaml
56. kubectl delete replicaset.apps/hello-world-rest-api-797dd4b5dc
57. kubectl delete all -l app=hello-world-rest-api
58. kubectl delete deployment hello-world-rest-api
59. kubectl delete all -l app=hello-world-rest-api
60. kubectl delete pod hello-world-rest-api-67c79fd44f-8bhdt

## SHORTCUTS

61. kubectl get services
62. kubectl get svc
63. kubectl get ev
64. kubectl get rs
65. kubectl get ns
66. kubectl get nodes
67. kubectl get no
68. kubectl get pods
69. kubectl get po
70. kubectl get all

# Creating Pods -Dry run

Saturday, 15 April 2023 2:29 PM

```
# kubectl run nginx --image nginx --dry-run=client
pod/nginx created (dry run)

# kubectl run nginx --image nginx --dry-run=client -o yaml
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx
    name: nginx
spec:
  containers:
  - image: nginx
    name: nginx
    resources: {}
  dnsPolicy: ClusterFirst
  restartPolicy: Always
status: {}

# kubectl run test --image nginx --port 80

# kubectl get pods -o wide

# kubectl describe pod test

=====
```

Declarative way

```
# kubectl create -f pod-definition.yml
```

If manifest file is changed

```
# kubectl apply -f pod-definition.yml
```

```
# kubectl delete pod <pod-name>
```

# Pod

Saturday, May 7, 2022 4:31 PM

## POD

- A Pod is the smallest unit of Kubernetes
- it will run only one application on each Pod.
- Pods can hold multiple containers.
- each pod has its own IP address
- A new IP address will be added per creation of a pod after it dies
- this is usually something that we do not want and can be solved with Kubernetes Service.
- Every POD can reach other POD
- Any container in same pod will share same storage volumes and network resources and communicate using localhost.
- Pods are used as unit of replication in kubernetes
- If load too much, it can be configured to deploy new replicas of your pod to the cluster as necessary.

## Scaling Pods

- All containers within the pod get scaled together.
- The pod is the unit of scale in K8s.
- In K8s, **initcontainer** is sometimes used as second container inside pod.

# Service and Ingress

Thursday, May 5, 2022 9:35 PM

## Service (Communication)

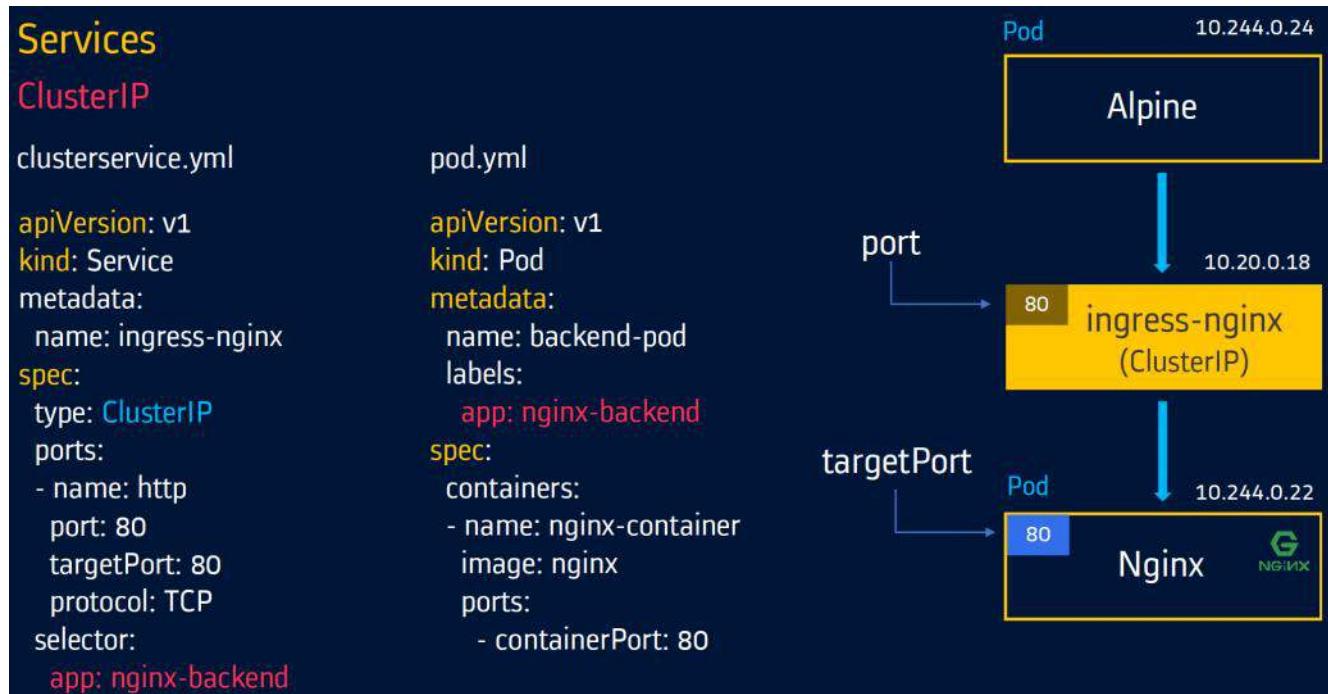
- Permanent IP address that can be attached to each Pod
- Lifecycle of Pod and service NOT connected
- External service opens the communication from external sources
- Internal service for internal
- A service allows communication between pods by adding a permanent IP to the pod
- This means if the pod dies it will conserve its IP.
- It also acts as a load balancer by locating the less loaded POD and sending those requests to it.

Three types:

- ClusterIP
- NodePort
- LoadBalancer

## ClusterIP

- Default kubernetes service.
- Gives service inside your cluster that other apps inside your cluster can access.
- It restricts access to the application within the cluster itself and no external access.
- Useful when a front-end app wants to communicate with back-end.
- Each clusterIP service gets a unique IP address inside the cluster.
- Similar to --links in docker.



# Kubernetes

## Services

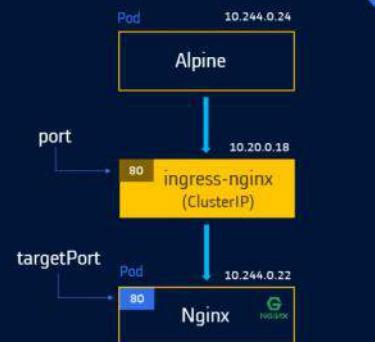
### ClusterIP

clusterip-service.yml

```
apiVersion: v1
kind: Service
metadata:
  name: ingress-nginx
spec:
  type: ClusterIP
  ports:
    - name: http
      port: 80
      targetPort: 80
      protocol: TCP
  selector:
    app: nginx-backend
```

pod.yml

```
apiVersion: v1
kind: Pod
metadata:
  name: backend-pod
  labels:
    app: nginx-backend
spec:
  containers:
    - name: nginx-container
      image: nginx
      ports:
        - containerPort: 80
```



kubectl create -f clusterservice.yml  
kubectl create -f pod.yml

root@alpine: # curl ingress-nginx

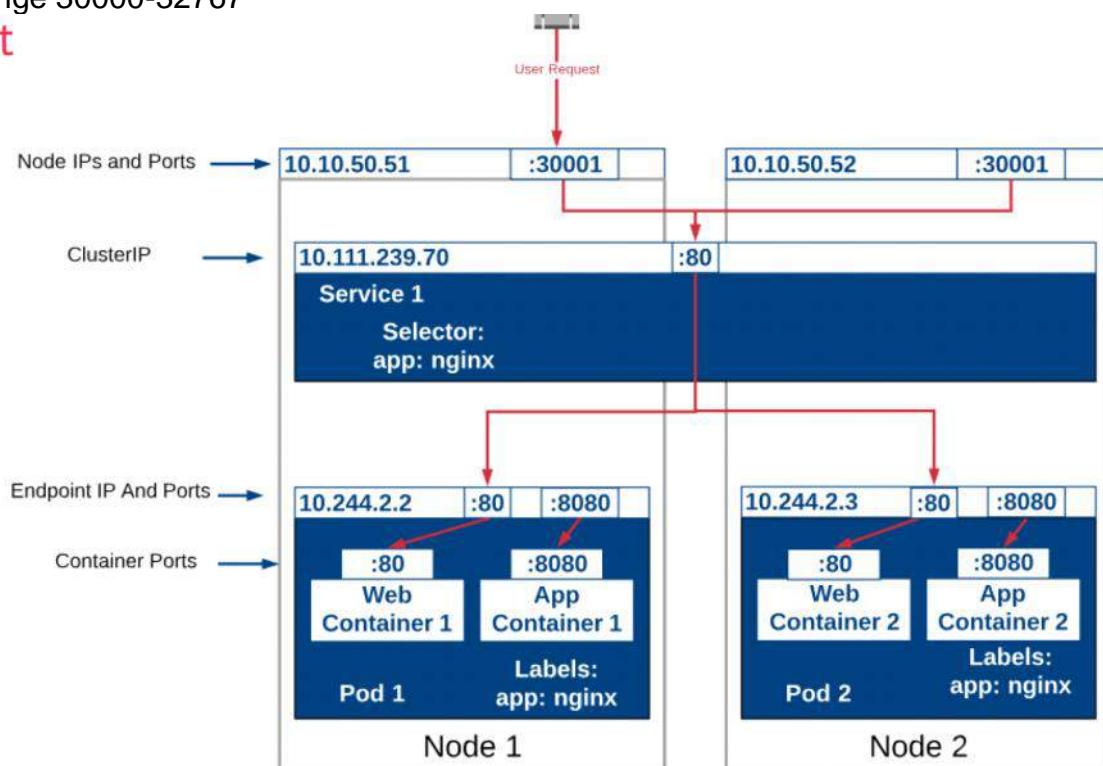
check the endpoints: kubectl describe svc/<svc-name>

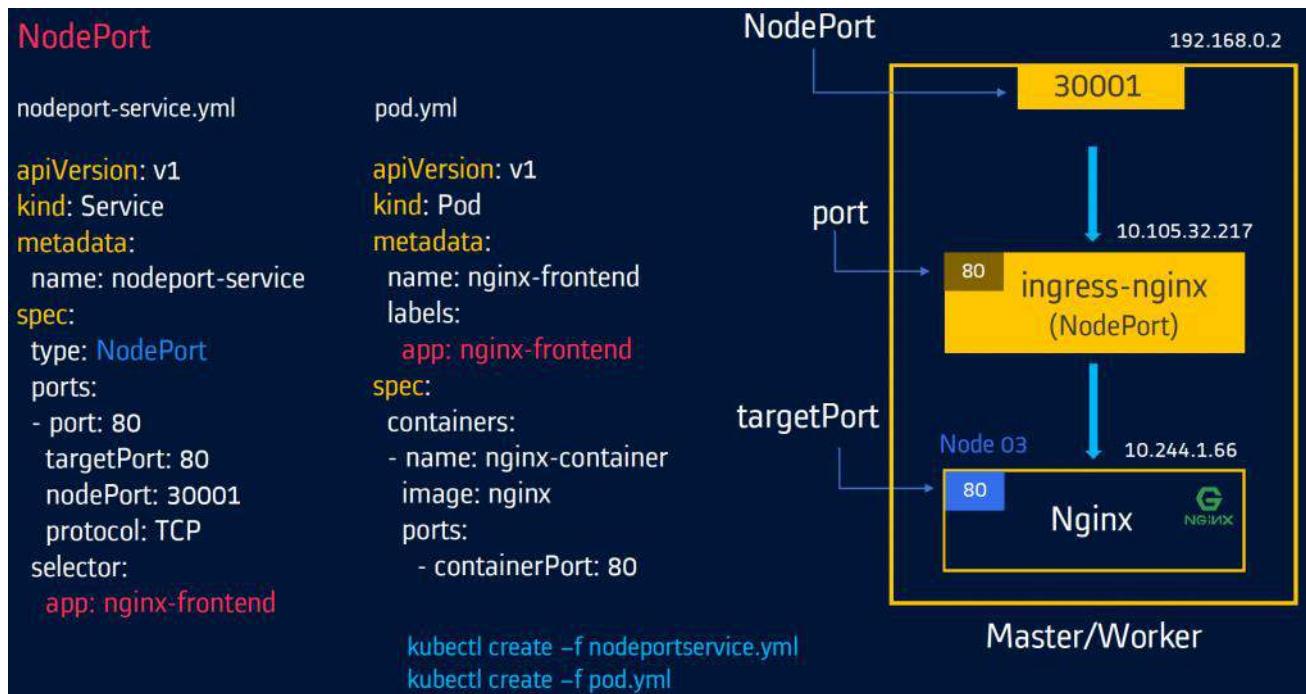
#####

### NodePort

- It opens a specific port on all the nodes in cluster and forward traffic to internal services.
- Useful when front end pods are to be exposed outside for user access.
- App can be reachable from any available nodes in the cluster
- Build on top of clusterIP
- Port range 30000-32767

### NodePort





## Kubernetes

Demo: **NodePort**

```
kubectl create -f nodeport-service.yml
kubectl create -f pod.yml
```

```
root@k-master:/home/osboxes# kubectl create -f node.yml
service/nodeport-nginx-service created
pod/nginx-frontend created
root@k-master:/home/osboxes#
```

**kubectl get services**

```
root@k-master:/home/osboxes# kubectl get services
NAME           TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes     ClusterIP  10.96.0.1   <none>       443/TCP   13m
nodeport-nginx-service  NodePort   10.105.32.217  <none>     80:30001/TCP  2m3s
root@k-master:/home/osboxes# kubectl get pods
NAME            READY   STATUS    RESTARTS   AGE
nginx-frontend  1/1     Running   0          2m10s
root@k-master:/home/osboxes#
```

```
# kubectl describe service nodeport-nginx-service
# kubectl get nodes -o wide
```

```
#####
#####
```

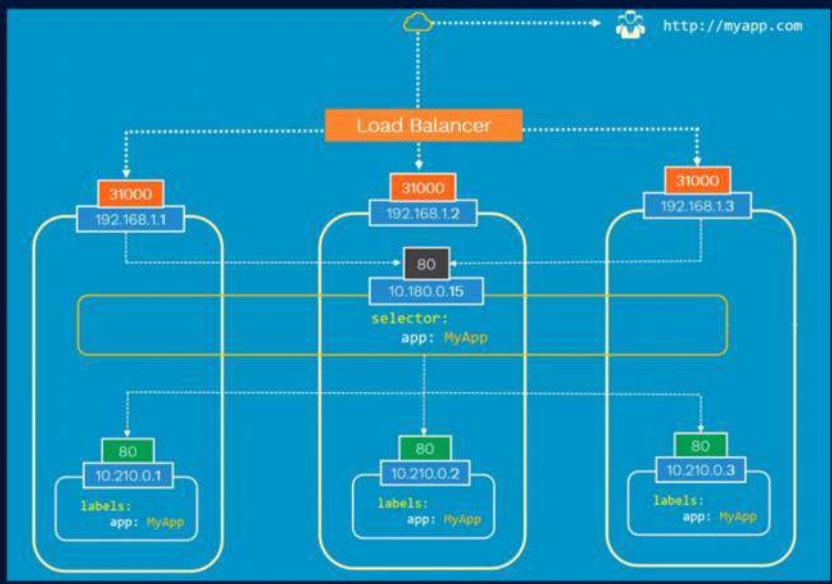
## Load Balancer

- Standard way to expose a service to the internet
- LB gives you single IP that will forward all external traffic to your service.
- No filtering - Can send any kind of traffic, HTTP, TCP, UDP or WebSocket's
  
- Every service exposed will get's it's own IP.
- Expensive to have external IP for each of the service.

## Services

### Load Balancer

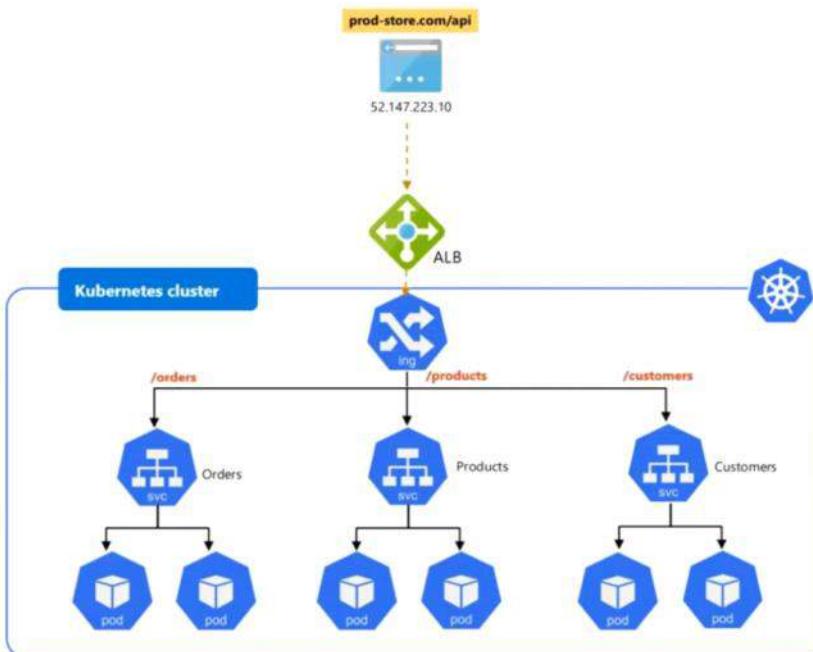
```
apiVersion: v1
kind: Service
metadata:
  name: lb-service
  labels:
    app: hello
spec:
  type: LoadBalancer
  selector:
    app: hello
  ports:
  - port: 80
    targetPort: 80
    protocol: TCP
```



#####
#####

### Ingress (route the traffic into cluster)

- When you would want your url to look like in prod (secure protocol and domain name)
- For Service the request goes to first ingress and it does forwarding then to service
- Helps users access the application using a single eternally accessible URL.

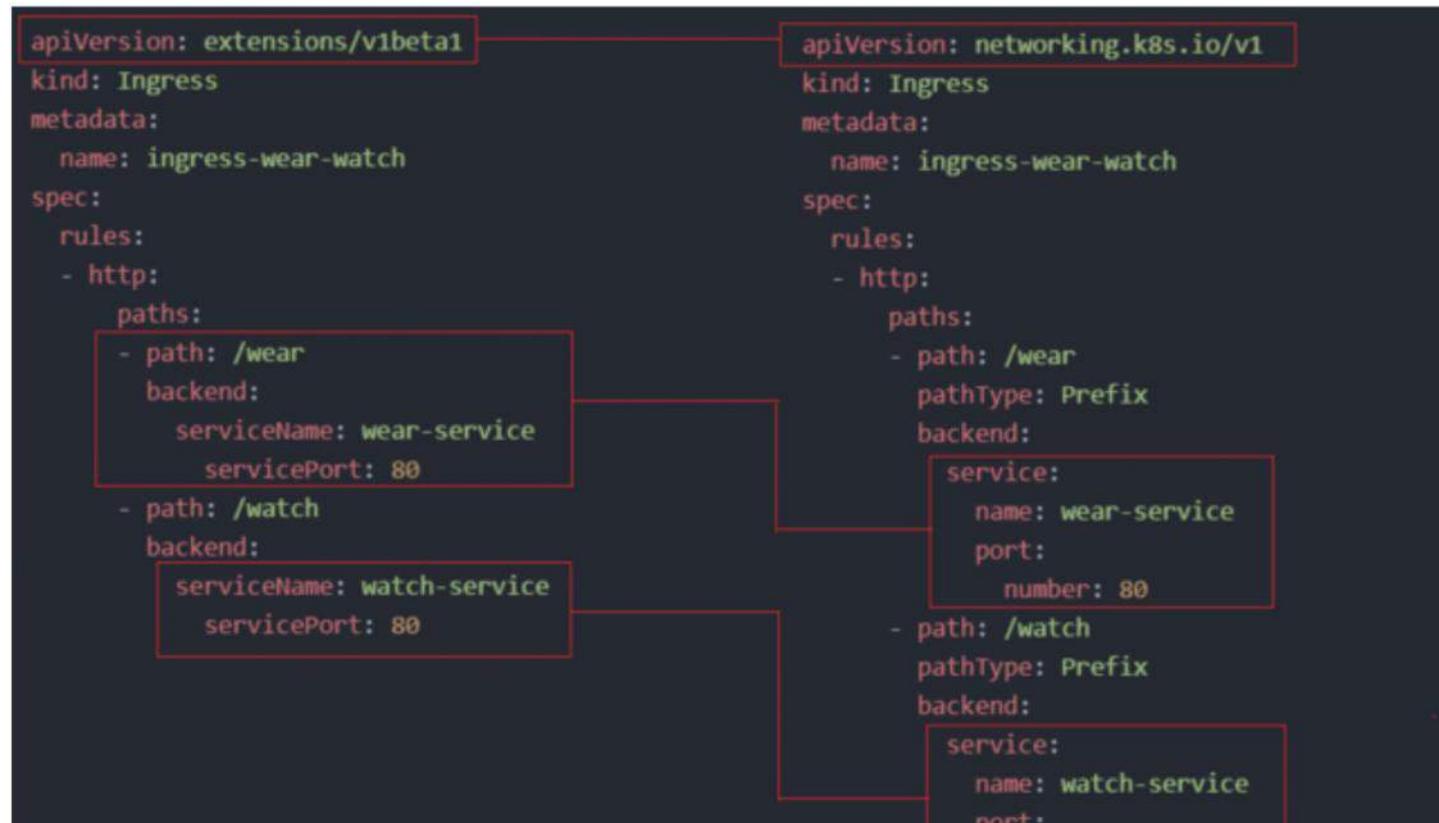
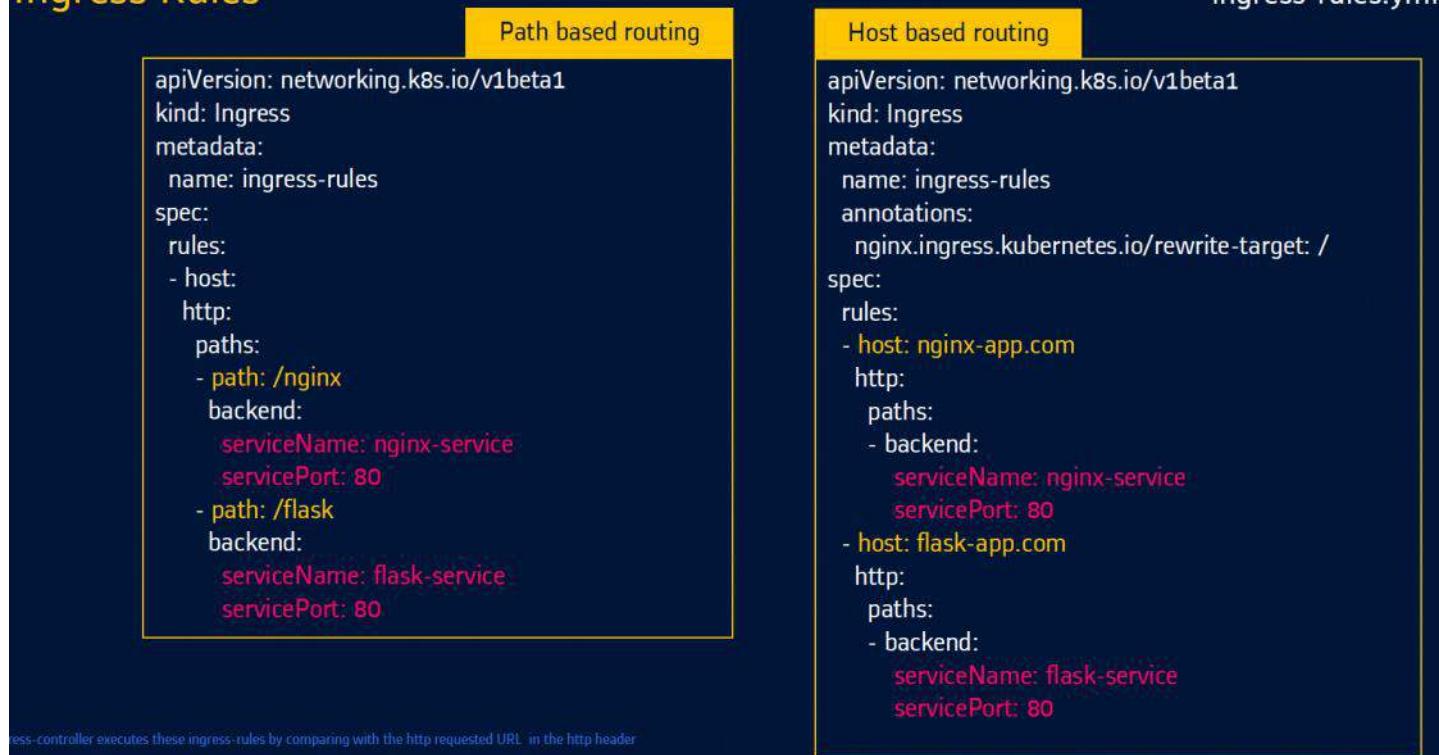


- Ingress acts as internal LoadBalancer
- Routes traffic based on URL path
- All applications will need only one public IP

## Ingress Controller

- It Implement rules defined by ingress resources.
- Useful when you want to expose multiple services under the same IP address.
- It can perform LB, Auth, SSL, URL/Path based routing by being inside the cluster living as deployment.

## Ingress Rules



# Imperatively | Declaratively

Saturday, 15 April 2023 2:03 PM

Deploy objects in two way

- Imperatively
  - o Verb-based commands like kubectl run, create, expose, delete, scale, edit
  - o For testing and experimentation
- Declaratively
  - o Objects are written in YAML and deploy using kubectl create or apply
  - o Best suited for prod env

# Manifest / Spec file

Saturday, 15 April 2023 2:09 PM

A file can include one or more API object descriptions (manifests).

```
# manifest file template
apiVersion - version of the Kubernetes API
              used to create the object
kind - kind of object being created
metadata - Data that helps uniquely identify
              the object, including a name and
              optional namespace
spec - configuration that defines the desired for
              the object
```

```
apiVersion: v1
kind: Pod
metadata:
  name: ...
spec:
  containers:
    - name: ...
...
apiVersion: v1
kind: Pod
metadata:
  name: ...
spec:
  containers:
    - name: ...
```

Multiple resource definitions

## Manifest files Man Pages

List all K8s API supported Objects and Versions

```
kubectl api-resources
kubectl api-versions
```

### Man pages for objects

```
kubectl explain <object>.<option>
kubectl explain pod
kubectl explain pod.apiVersion
kubectl explain pod.spec
```

```
apiVersion: v1
kind: Pod
metadata:
  name: ...
spec:
  containers:
    - name: ...
...
apiVersion: v1
kind: Pod
metadata:
  name: ...
spec:
  containers:
    - name: ...
```

Multiple resource definitions

# Replication

Thursday, May 19, 2022 9:30 PM

## Replication

- Single pod may not be sufficient to handle the user traffic.
  - If pod goes down, k8s will not bring this pod up again.
- 
- K8s support different controllers
    - o Replication Controller
    - o ReplicaSet

## Replication Controller

It will ensure that the desired number of containers are running all the times in your application group.

High availability by replacing the unhealthy/dead pods with new ones

We need replication controller is to create multiple pods to share the load across them.

```
# kubectl run nginx -image=nginx -replicas=3
```

## ReplicaSet

Replica controller is deprecated and replaced by Replicaset

- Higher level API that gives the ability to easily run multiple instances of given pod
- Replace any failed pods with new ones.
- Replica count is controlled by the replicas field in the resource definition file.
- Replicaset uses set-based selectors
- ReplicaController uses equality based selectors.

# Kubernetes

## Pod vs ReplicaSet

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
  labels:
    app: webapp
spec:
  containers:
    - name: nginx-container
      image: nginx
      ports:
        - containerPort: 80
```

pod.yml

*Actually definition of POD itself*

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx-replicaset
labels:
  app: webapp
  type: front-end
spec:
  replicas: 3
  selector:
    matchLabels:
      app: webapp
  template:
    metadata:
      name: nginx-pod
      labels:
        app: webapp
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

# Kubernetes

## ReplicaSet Manifest file

Number of pods (replicas)

Which pods to watch?

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: hello
  labels:
    app: hello
spec:
  replicas: 5
  selector:
    matchLabels:
      app: hello
  template:
    metadata:
      labels:
        app: hello
    spec:
      containers:
        - name: hello-container
          image: busybox
          command: [ ... ]
```

Pod template to use

# Kubernetes

## ReplicaSet

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx-replicaset
  labels:
    app: webapp
    type: front-end
spec:
  replicas: 3
  selector:
    matchLabels:
      app: webapp
  template:
    metadata:
      name: nginx-pod
      labels:
        app: webapp
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

## Kubernetes

### ReplicaSet

```
kubectl create -f replica-set.yml
```

```
root@k-master:/home/osboxes# kubectl create -f replica-set.yml
replicaset.apps/nginx-replicaset created
root@k-master:/home/osboxes#
```

```
kubectl get rs -o wide
```

```
root@k-master:/home/osboxes# kubectl get rs -o wide
NAME      DESIRED  CURRENT  READY   AGE    CONTAINERS   IMAGES   SELECTOR
nginx    3         3        3       76s   nginx-container   nginx   app=webapp
root@k-master:/home/osboxes#
```

```
kubectl get pods -o wide
```

```
root@k-master:/home/osboxes# kubectl get pods -o wide
NAME          READY  STATUS    RESTARTS   AGE     IP           NODE
nginx replicaset-67nb9  1/1   Running   0          93s   10.244.1.9  k-slave01
nginx replicaset-g85dz  1/1   Running   0          93s   10.244.2.4  k-slave02
nginx replicaset-l9cpz  1/1   Running   0          93s   10.244.1.8  k-slave01
```

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: nginx-replicaset
  labels:
    app: webapp
    type: front-end
spec:
  replicas: 3
  selector:
    matchLabels:
      app: webapp
  template:
    metadata:
      name: nginx-pod
      labels:
        app: webapp
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

## Kubernetes

### ReplicaSet

- `kubectl edit replicaset <replicaset-name>` - edit a replicaset; like image, replicas
- `kubectl delete replicaset <replicaset-name>` - delete a replicaset; like image, replicas
- `kubectl delete -f replica-set.yml`
- `kubectl get all` - get pods, replicasets, deployments, services all in one shot
- `kubectl replace -f replicaset-definition.yml` -replaces the pods with updated definition file
- `kubectl scale --replicas=6 -f replicaset-definition.yml` – scale using definition file
- `kubectl scale --replicas=6 replicaset <replicaset-name>` - using name of replicaset

# ConfigMap and Secret

Thursday, May 5, 2022 11:23 PM

## ConfigMap (External Configuration)

- For small change like database url kubernetes has a component called config map.
- It's basically external configuration of your application.
- It contain configuration data like urls of database or some other services that you use and in kubernetes you just connect to Pod so that Pod actually gets the data that configMap contains.
- If you change the name of the service the end point of the service you just adjust the configMap and that's it you don't have to build new image and have to go through this whole cycle.

## Secret

- Part of the external configuration can also be database username and password which may also change in the application deployment process.
- Another component called secret same like configMap
- Difference is it used to store secret data like credentials & certificates things in base63 encoded
- Connect it directly to Pod so it can see the data
- Can use the data from configMap or secret inside the application Pod using environment variables.

# Deployment

Friday, May 6, 2022 12:07 AM

## Deployment (Pod blueprint with replicating mechanism)

- If Pod dies service will forward the request to another replica
- The replica is connected to the same service
- Service has 2 functionalities. Permanent IP & Load balancer
- Blueprint for my application pod and specify how many replicas of the pod you would like to run.
- This blueprint component is called deployment.
- You can scale up or scale down the number of replicas.
- That part is layer of abstraction on top of containers and deployment is another abstraction on top of Pod
- It Provides decelerative updates for Pods and ReplicaSets
- Similar to ReplicaSets but with advanced function.
  - o Easily deploy a RS
  - o Rolling updates pods
  - o Rollback to previous deployment versions.
  - o Scale deployment
  - o Pause and resume deployment
- K8s triggers a Rollout when we create new deployment.
- It's process of gradually deploying or upgrading your application containers.
- For every rollout upgrade, a version history will be created which help rolling back.

Different ways to release updates.

- Recreate: terminate old version and release new one, downtime required.
- Rolling update: Release new version on rolling update fashion.
- Blue/Green: Release a new version alongside the old version then switch traffic.

```
spec:  
  replicas: 10  
strategy:  
  type: Recreate
```

```
spec:  
  replicas: 10  
strategy:  
  type: RollingUpdate  
  rollingUpdate:  
    maxSurge: 2  
    maxUnavailable: 0
```

## Deployments

- `kubectl create deployment nginx --image nginx --dry-run -o yaml`
- `kubectl create -f deployment.yml --record` (`--record` is optional, it just records the events in the deployment)

```
root@k-master:/home/osboxes# kubectl create -f deployment.yml
deployment.apps/nginx-deployment created
root@k-master:/home/osboxes#
```

- `kubectl get deployments`

```
root@k-master:/home/osboxes# kubectl get deployments -o wide
NAME           READY   UP-TO-DATE   AVAILABLE   AGE   CONTAINERS
nginx-deployment  5/5     5          5          117s  nginx-container
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 10
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

## Deployments

- `kubectl describe deployment <deployment-name>`

```
root@k-master:/home/osboxes# kubectl describe deployment nginx-deployment
Name:           nginx-deployment
Namespace:      default
CreationTimestamp:  Tue, 19 May 2020 03:40:19 -0400
Labels:          app=nginx
Annotations:    deployment.kubernetes.io/revision: 1
                 kubernetes.io/change-cause: kubectl create --filename=deployment.yml
Selector:        app=nginx
Replicas:       5 desired | 5 updated | 5 total | 5 available | 0 unavailable
StrategyType:   RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=nginx
  Containers:
    nginx-container:
      Image:  nginx
      Port:   80/TCP
      Host Port:  80/TCP
      Environment:  <none>
      Mounts:  <none>
      Volumes: <none>
  Conditions:
    Type     Status  Reason
    ...
    Available  True    MinimumReplicasAvailable
    Progressing  True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   nginx-deployment-96577bc6d (5/5 replicas created)
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 10
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

## Deployments

- `kubectl get pods -o wide`

```
root@k-master:/home/osboxes# kubectl get pods -o wide
NAME           READY   STATUS    RESTARTS   AGE   IP          NODE
TEST
nginx-deployment-96577bc6d-2hkpk  1/1    Running   0          3m34s  10.244.2.20  k-slave02
nginx-deployment-96577bc6d-h5gdv  1/1    Running   0          3m34s  10.244.2.19  k-slave02
nginx-deployment-96577bc6d-nqtn6  1/1    Running   0          3m34s  10.244.2.22  k-slave02
nginx-deployment-96577bc6d-pd4cg  1/1    Running   0          3m34s  10.244.2.21  k-slave02
nginx-deployment-96577bc6d-tphhh  1/1    Running   0          3m34s  10.244.2.23  k-slave02
root@k-master:/home/osboxes#
```

- `kubectl edit deployment <deployment -name>` - perform live edit of deployment
- `kubectl scale deployment <deployment -name> --replicas2`
- `kubectl apply -f deployment.yml` – redeploy a modified yaml file; Ex: replicas changed to 5, image to nginx:1.18

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 10
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
```

## Deployments

- `kubectl rollout status deployment <deployment -name>`

- `kubectl rollout history deployment <deployment -name>`

```
root@k-master:/home/osboxes# k rollout history deployment.apps/nginx-deployment
deployment.apps/nginx-deployment
REVISION  CHANGE-CAUSE
1         kubectl create --filename=deployment.yml --record=true
2         kubectl create --filename=deployment.yml --record=true
```

## Deployments

- `kubectl rollout undo deployment <deployment -name>`

```
root@k-master:/home/osboxes# kubectl rollout undo deployment.apps/nginx-deployment
deployment.apps/nginx-deployment rolled back
root@k-master:/home/osboxes# k rollout history deployment.apps/nginx-deployment
deployment.apps/nginx-deployment
REVISION  CHANGE-CAUSE
2          kubectl create --filename=deployment.yml --record=true
3 ←        kubectl create --filename=deployment.yml --record=true
```

- `kubectl rollout undo deployment <deployment-name> --to-revision=1`
  - `kubectl rollout pause deployment <deployment-name>`
  - `kubectl rollout resume deployment <deployment-name>`
  - `kubectl delete -f <deployment-yaml-file>` - deletes deployment and related dependencies
  - `kubectl delete all --all` – deletes pods, replicaset, deployments and services in current namespace

# Stateful Set

Sunday, 23 April 2023 3:08 PM

## Stateful Set (for DB application)

- DB can't be replicated via deployment because DB has a state which is data
- If replicas is there then need to use shared data storage
- It should be created using stateful sets and not deployments
- Stateful set just like deployment that would take care of replicating the pods and scaling them up or down but make sure database reads and writes are in sync
- Best practice to host DB outside of K8s cluster

This is how we deploy new pods, it is meant for stateless applications (applications that do not store data).

# K8s Ports

Sunday, May 8, 2022 12:27 PM

6443 - Allow communication for your endpoint

10250 - Kubelete to communicate with your master node.

Add firewall

```
# firewall-cmd --permanent --add-port=6443/tcp  
# firewall-cmd --permanent --add-port=10250/tcp
```

Allow bridge networking

```
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf  
net.bridge.bridge-nf-call-ip6tables = 1  
net.bridge.bridge-nf-call-iptables = 1  
EOF
```

Reload system

```
sysctl --system
```

Install package

```
# yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes; systemctl enable --now kubelet
```

Initiate kubeadm

```
# kubeadm init --pod-network-cidr 10.244.0.0/16 --apiserver-advertise-address=192.168.27.128
```

If CRI error then Solution is:

```
rm /etc/containerd/config.toml  
systemctl restart containerd  
kubeadm init
```

Protocol	Direction	Port Range	Purpose	Used By
TCP	Inbound	6443*	Kubernetes API server	All
TCP	Inbound	2379-2380	etcd server client API	kube-apiserver, etcd
TCP	Inbound	10250	Kubelet API	Self, Control plane
TCP	Inbound	10251	kube-scheduler	Self
TCP	Inbound	10252	kube-controller-manager	Self

# PV & PVC

Wednesday, 29 June 2022 3:28 PM

A persistent volume claim (**PVC**) is a request for storage by a user from a **PV**. Claims can request specific size and access modes

- `ReadWriteOnce` – the volume can be mounted as read-write by a single node
- `ReadOnlyMany` – the volume can be mounted read-only by many nodes
- `ReadWriteMany` – the volume can be mounted as read-write by many nodes

```
#PV
apiVersion: v1
kind: PersistentVolume
metadata:
  name: task-pv-volume
  labels:
    type: local
spec:
  storageClassName: manual
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data"
```

```
#PVC
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: task-pv-claim
spec:
  storageClassName: manual
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 3Gi
```

## Commands:

```
# kubectl create -f pv.yaml
# kubectl get pv pv-volume

# kubectl create -f pvc.yaml
# kubectl get pvc pvc-claim

# kubectl delete pvc pvc-claim
# kubectl delete pv pv-volume
```

## POD

```
apiVersion: v1
kind: Pod
metadata:
  name: task-pv-pod
spec:
  volumes:
    - name: task-pv-storage
      persistentVolumeClaim:
        claimName: task-pv-claim
  containers:
    - name: task-pv-container
      image: nginx
      ports:
        - containerPort: 80
          name: "http-server"
  volumeMounts:
    - mountPath: "/usr/share/nginx/html"
      name: task-pv-storage
```

# Labels and Selectors

Saturday, 15 April 2023 3:27 PM

## Labels and Selectors

### Labels

- Labels are key/value pairs that are attached to objects, such as pods
- Labels allows to logically group certain objects by giving various names to them
- You can label pods, services, deployments and even nodes

```
kubectl get pods -l environment=production  
kubectl get pods -l environment=production,  
tier=frontend
```

```
"metadata": {  
  "labels": {  
    "key1" : "value1",  
    "key2" : "value2"  
  }  
}
```

```
apiVersion: v1  
kind: Pod  
metadata:  
  name: label-demo  
  labels:  
    environment: production  
    app: nginx  
spec:  
  containers:  
  - name: nginx  
    image: nginx:1.14.2  
    ports:  
    - containerPort: 80
```

## Labels and Selectors

- If labels are not mentioned while deploying k8s objects using imperative commands, the label is auto set as `app: <object-name>`

```
kubectl run --image nginx nginx  
kubectl get pods --show-labels
```

```
root@k8s-master:/home/osboxes# kubectl get pods --show-labels  
NAME      READY   STATUS    RESTARTS   AGE     LABELS  
nginx    1/1     Running   0          90s    run=nginx
```

### Adding Labels

```
kubectl label pod nginx environment=dev
```

```
root@k8s-master:/home/osboxes# kubectl label pod nginx environment=dev  
pod/nginx labeled  
root@k8s-master:/home/osboxes# kubectl get pods --show-labels  
NAME      READY   STATUS    RESTARTS   AGE     LABELS  
nginx    1/1     Running   0          6m45s   environment=dev,run=nginx  
root@k8s-master:/home/osboxes#
```

## Selectors

- Selectors allows to filter the objects based on labels
- The API currently supports two types of selectors: **equality-based** and **set-based**
- A label selector can be made of multiple requirements which are comma-separated

### Equality-based Selector

- Equality- or inequality-based requirements allow filtering by label keys and values.
- Three kinds of operators are admitted `=,==,!=`

```
environment = production  
tier != frontend
```

```
apiVersion: v1  
kind: Pod  
metadata:  
  name: cuda-test  
spec:  
  containers:  
    - name: cuda-test  
      image: "k8s.gcr.io/cuda-vector-add:v0.1"  
      resources:  
        limits:  
          nvidia.com/gpu: 1  
      nodeSelector:  
        accelerator: nvidia-tesla-p100
```

Used by Replication Controllers and Services

### Set-based Selector

- Set-based label requirements allow filtering keys according to a set of values.
- Three kinds of **operators** are supported: `in,notin,exists` (only the key identifier).

```
kubectl get pods -l 'environment in (production, qa)'
```

```
environment in (production, qa)  
tier notin (frontend, backend)  
partition  
!partition
```

```
selector:  
matchLabels:  
  component: redis  
matchExpressions:  
  - {key: tier, operator: In, values: [cache]}  
  - {key: environment, operator: NotIn, values: [dev]}
```

Used by ReplicaSets, Deployments, DaemonSets

# Namespace

Saturday, 17 December 2022 3:56 PM

Namespace are Kubernetes objects which partition a single Kubernetes cluster into multiple virtual clusters. Each of namespaces have their own set of rules that defines who does what. Have their own set of resources that they can consume.

Companies often choose to deploy projects created by separate teams to shared cluster. Multiple deployment in single cluster is risky, high chances of deleting deployment belong to different projects.

Namespaces allow you to group objects together so you can control them as a unit / group.

K8s cluster is created with the following three namespaces:

**Default** - Create automatically when cluster is setup. And whatever you create such as Pods, deployment, services etc are created inside default namespace.

**Kube-public** - This is where resources that should be made available to all users are created.

**Kube-system** - Set of pods and services for its internal purpose such as those required by network solution, dns services etc. To isolate this from the users so you don't accidentally delete or modify them.

## Namespaces

### kubectl get namespaces

```
root@k8s-master:/home/osboxes# kubectl get ns
NAME      STATUS   AGE
default   Active   68m
kube-node-lease   Active   68m
kube-public   Active   68m
kube-system   Active   68m
```

### kubectl get all -n kube-system (lists available objects under a specific namespace)

```
root@k8s-master:/home/osboxes# kubectl get all -n kube-system
NAME          READY   STATUS    RESTARTS   AGE
pod/coredns-66bff467f8-bm6kr   1/1     Running   0          68m
pod/coredns-66bff467f8-hj9ll   1/1     Running   0          68m
pod/etcfd-k8s-master           1/1     Running   0          68m
pod/kube-apiserver-k8s-master  1/1     Running   0          68m
pod/kube-controller-manager-k8s-master  1/1     Running   0          68m
pod/kube-flannel-ds-amd64-bc5vg  1/1     Running   0          67m
pod/kube-flannel-ds-amd64-cg48x  1/1     Running   0          68m
pod/kube-flannel-ds-amd64-xz8qn  1/1     Running   0          67m
pod/kube-proxy-rq77v            1/1     Running   0          68m
pod/kube-proxy-tl99m            1/1     Running   0          67m
pod/kube-proxy-w7bqz            1/1     Running   0          67m
pod/kube-scheduler-k8s-master   1/1     Running   0          68m

NAME        TYPE    CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kube-dns  ClusterIP  10.96.0.10  <none>       53/UDP,53/TCP,9153/TCP  68m
```

### kubectl get all --all-namespaces (lists available objects under all available namespaces)

## Namespaces

### Create a namespace

```
kubectl create ns dev # Namespace for Developer team  
kubectl create ns qa # Namespace for QA team  
kubectl create ns production # Namespace for Production team
```

### Deploy objects in a namespace

```
kubectl run nginx --image=nginx -n dev  
kubectl get pod/nginx -n dev  
kubectl apply --namespace=qa -f pod.yaml
```

```
root@k8s-master:/home/osboxes# kubectl run --image=nginx nginx -n dev  
pod/nginx created  
root@k8s-master:/home/osboxes# k get pods -n dev  
NAME      READY   STATUS    RESTARTS   AGE  
nginx     1/1     Running   0          7m37s  
root@k8s-master:/home/osboxes#
```

### Delete a namespace

```
kubectl delete ns production
```

Mysql.connect("db-service.dev.svc.cluster.local")

Cluster.local - domain

Svc - service

Dev - Namespace

Db-service - Service name

# Disable swap for Kubernetes?

Monday, 13 February 2023 5:44 PM

The idea of kubernetes is to tightly pack instances to as close to 100% utilized as possible. All deployments should be pinned with CPU/memory limits. So if the scheduler sends a pod to a machine it should never use swap at all. You don't want to swap since it'll slow things down.

Its mainly for performance.

systemctl restart containerd

# Manifest

Monday, 24 April 2023 9:27 PM

# Configuration file

Saturday, May 7, 2022 4:11 PM

## Three parts of yaml configuration file

- Declaring (apiVersion, Kind [Deployment | Service | Ingress | Pod])
  - Metadata (name, label)
  - Specification (replicas, Selector, template, ports)
  - Status (Automatically generated and added by kubernetes) [Desired state | Actual State]
- 
- Attributes of spec are specific to the kind
  - K8s updates state continually
  - Where does K8s get this status data? - etcd
  - Etcd holds the current status of any K8s component
  - Format of configuration file is YAML
  - Store configuration file with your code or own git repo

```
K8s > ! nginx-deployment.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: nginx-depl
5    labels:
6    spec:
7      replicas: 2
8      selector:
9      template:
```

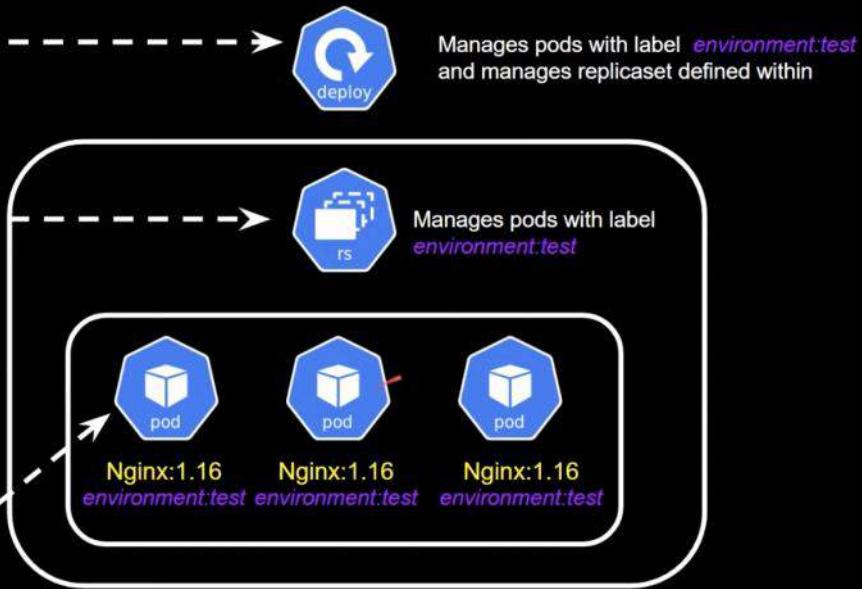
```
K8s > ! nginx-service.yaml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: nginx-service
5  spec:
6    selector:
7    ports:
```

# Deployment

Monday, 24 April 2023 9:27 PM

## Label

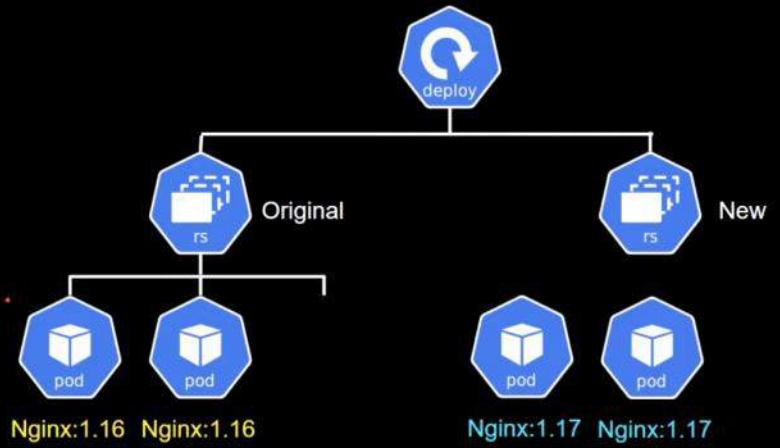
```
! nginx-deployment-withrolling.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    labels:
5      environment: test
6    name: testdeploy
7  spec:
8    replicas: 3
9    selector:
10   matchLabels:
11     environment: test
12   minReadySeconds: 10
13   strategy:
14     rollingUpdate:
15       maxSurge: 1
16       maxUnavailable: 0
17     type: RollingUpdate
18   template:
19     metadata:
20       labels:
21         environment: test
22     spec:
23       containers:
24         - image: nginx:1.16
25           name: nginx
```



## Rolling Update

- It will create new replica and create new container with new version
- `minReadySeconds: 10` - It will wait for 10 sec with new Pod and then it will delete 1 old pod

```
ginx-deployment-withrolling.yaml
  apiVersion: apps/v1
  kind: Deployment
  metadata:
    labels:
      environment: test
      name: testdeploy
  spec:
    replicas: 3
    selector:
      matchLabels:
        environment: test
    minReadySeconds: 10
    strategy:
      rollingUpdate:
        maxSurge: 1
        maxUnavailable: 0
      type: RollingUpdate
    template:
      metadata:
        labels:
          environment: test
      spec:
        containers:
          - image: nginx:1.17
            name: nginx
```



# ConfigMAP

Monday, 24 April 2023 11:37 PM

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: mongo-config
data:
  mongo-url: mongo-service
```

# Secret

Monday, 24 April 2023 11:37 PM

```
apiVersion: v1
kind: Secret
metadata:
  name: mongo-secret
  type: Opaque
data:
  mongo-user: mongo-admin
  mongo-password: mongo-p@ssW0rd
```

# Service

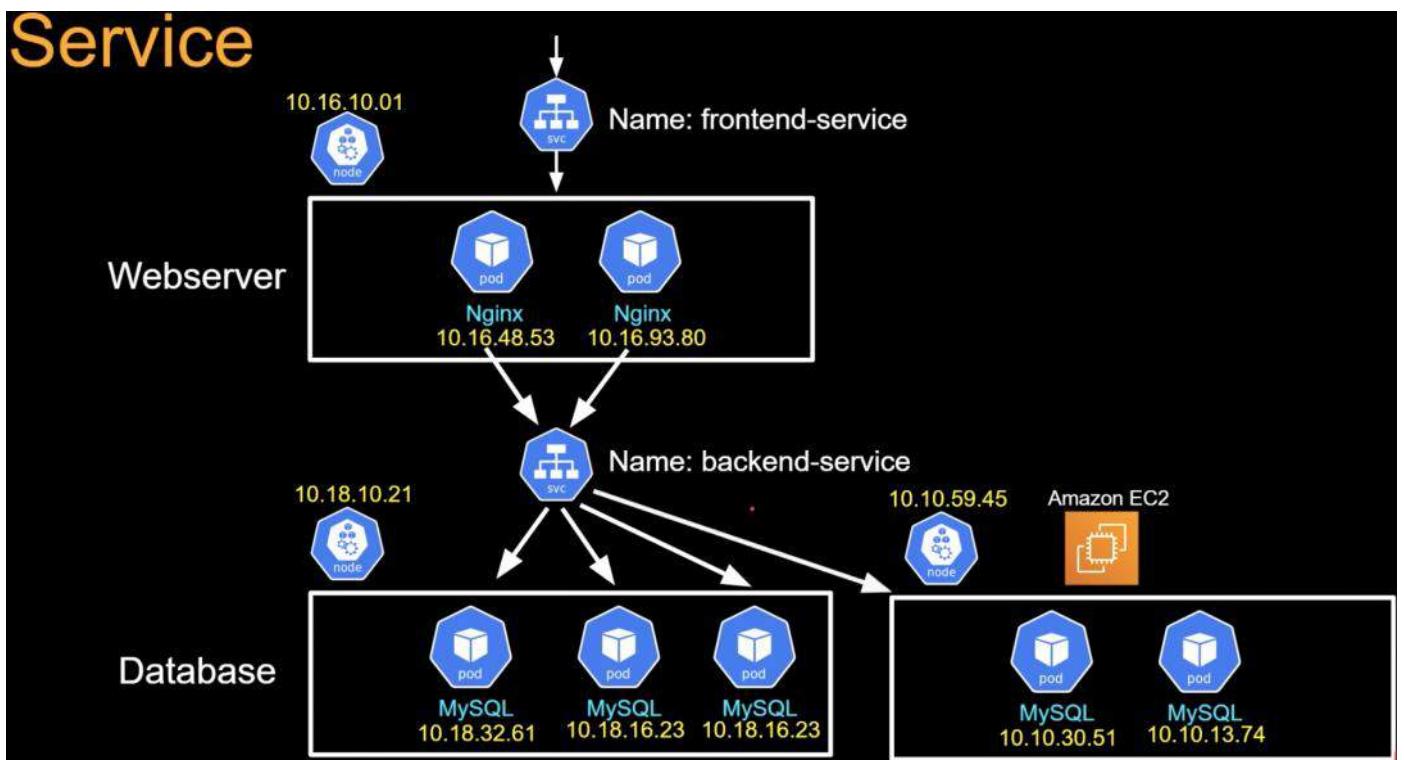
Monday, 24 April 2023 9:28 PM

## Frontend-service

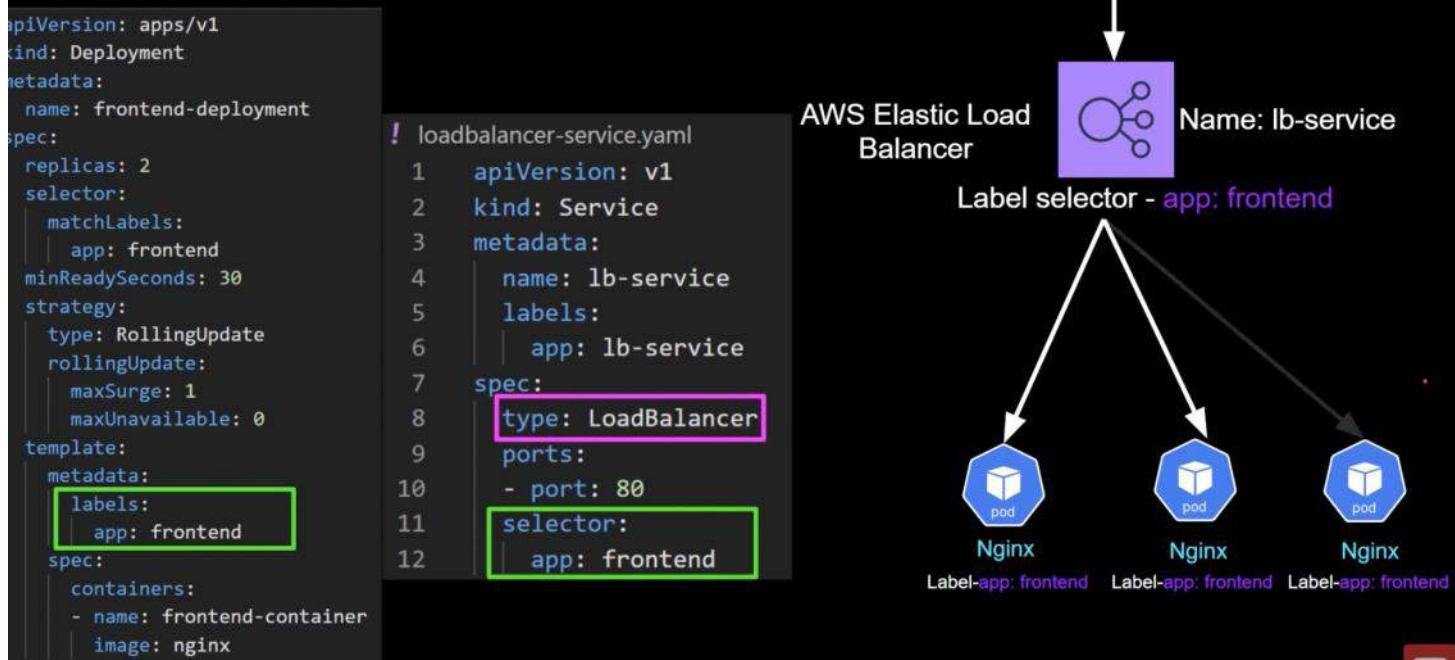
```
apiVersion: v1
kind: Service
metadata:
  name: frontend-service
spec:
  type: NodePort
  selector:
    app: webserver
  ports:
    - protocol: TCP
      port: 3000
      targetPort: 3000
      nodePort: 30100
```

## Backend-service

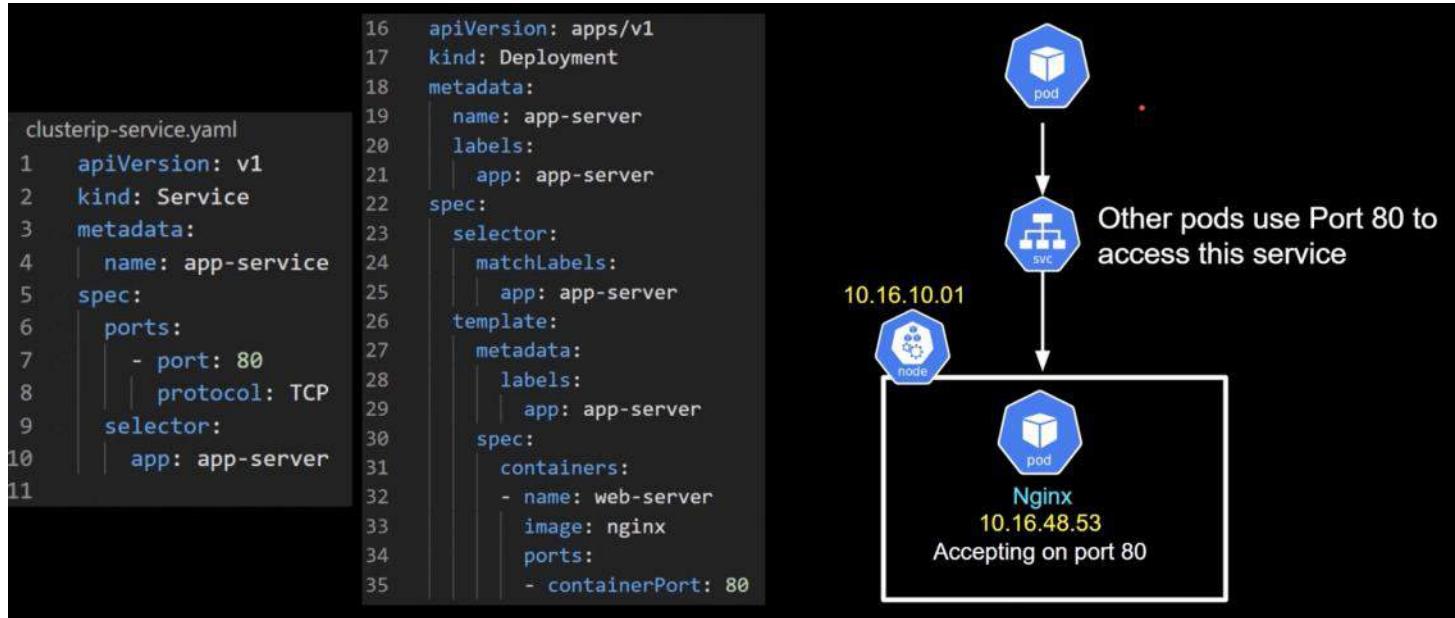
```
apiVersion: v1
kind: Service
metadata:
  name: backend-service
spec:
  selector:
    app: database
  ports:
    - protocol: TCP
      port: 27017
      targetPort: 27017
```



## LoadBalancer

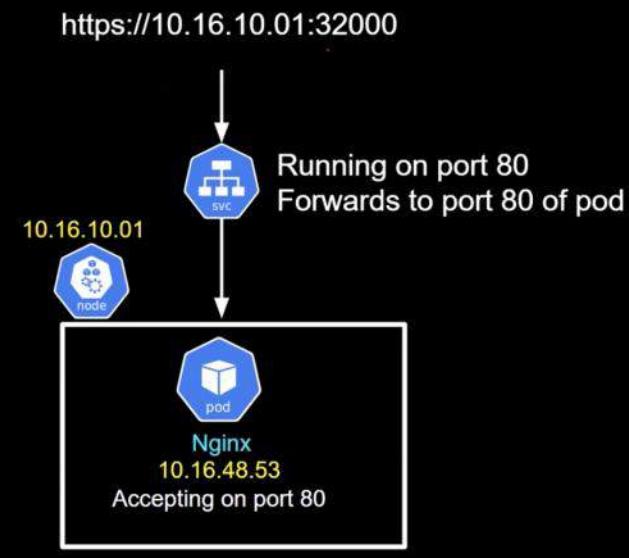


## ClusterIP



## Nodeport

```
nodeport-service.yaml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: app-service
5  spec:
6    type: NodePort
7    ports:
8      - nodePort: 32000
9        port: 80
10       targetPort: 80
11     selector:
12       app: app-server
13
14
15
16   apiVersion: apps/v1
17   kind: Deployment
18   metadata:
19     name: app-server
20     labels:
21       app: app-server
22   spec:
23     selector:
24       matchLabels:
25         app: app-server
26     template:
27       metadata:
28         labels:
29           app: app-server
30       spec:
31         containers:
32           - name: web-server
33             image: nginx
34             ports:
35               - containerPort: 80
```



# Ingress

Monday, 24 April 2023 9:28 PM

## Ingress - What and Why?

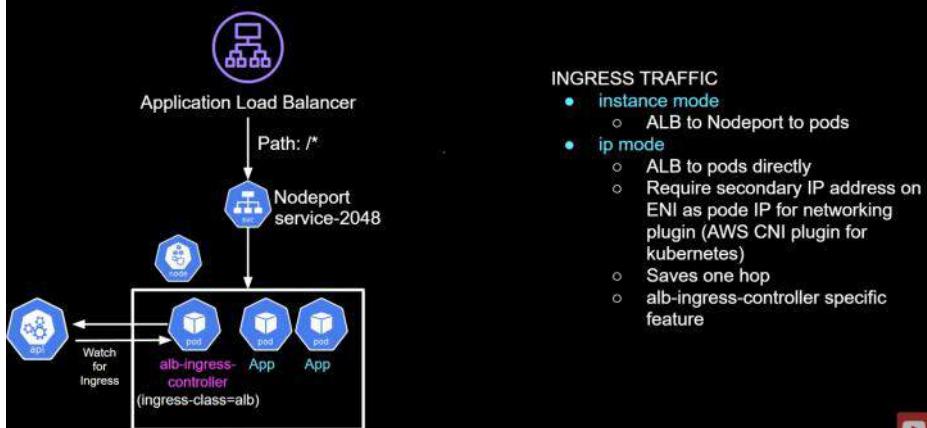
### INGRESS CONTROLLER

- Monitors Ingress resources
- Creates necessary AWS resources for Ingress
  - Such as ALB for ALB Ingress Controller
- One Cluster can have more than one Ingress Controller!
  - Ingress Resource defines which Ingress Controller to use

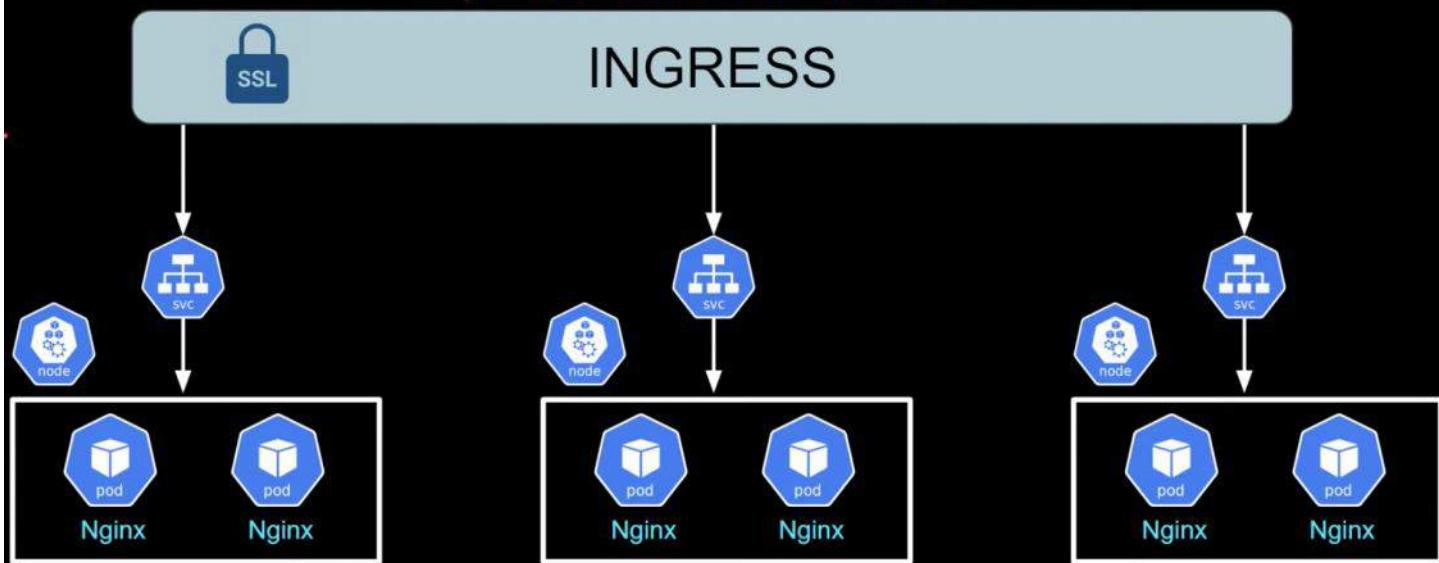
### INGRESS RESOURCE

- Selects which Ingress Controller to use
- Defines the URL Path and corresponding backend Service

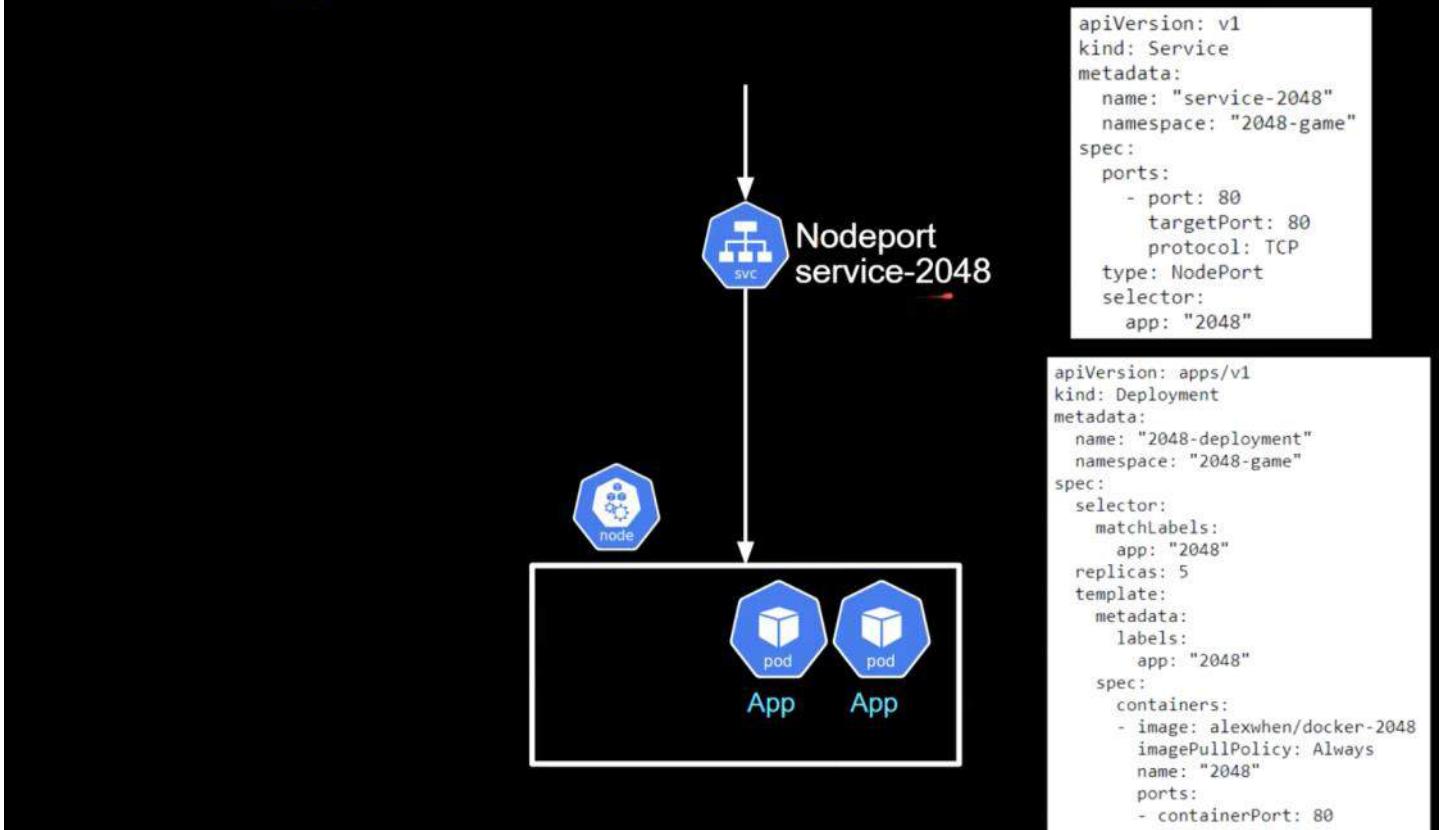
## ALB Ingress



<https://batcave.com/track-joker>  
<https://batcave.com/monitor-batcave>  
<https://batcave.com/order-new-batsuit>



## ALB Ingress



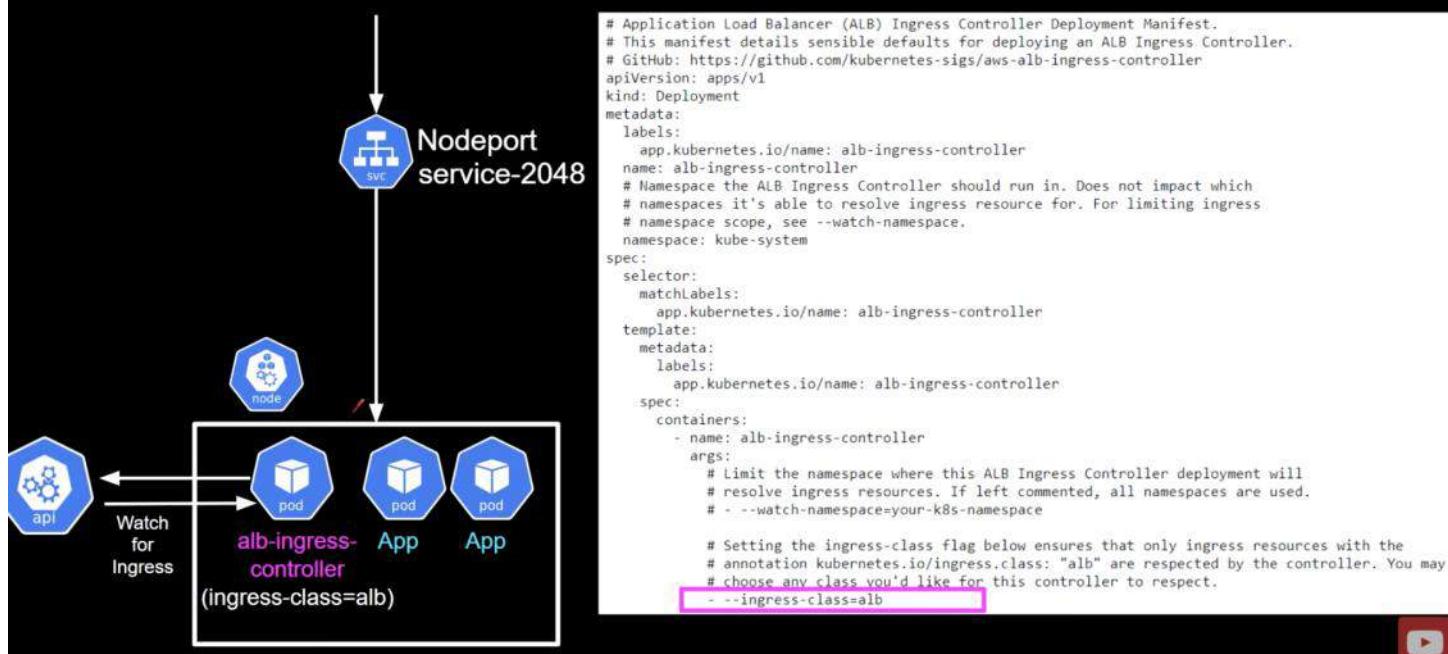
## ALB Ingress Controller

```

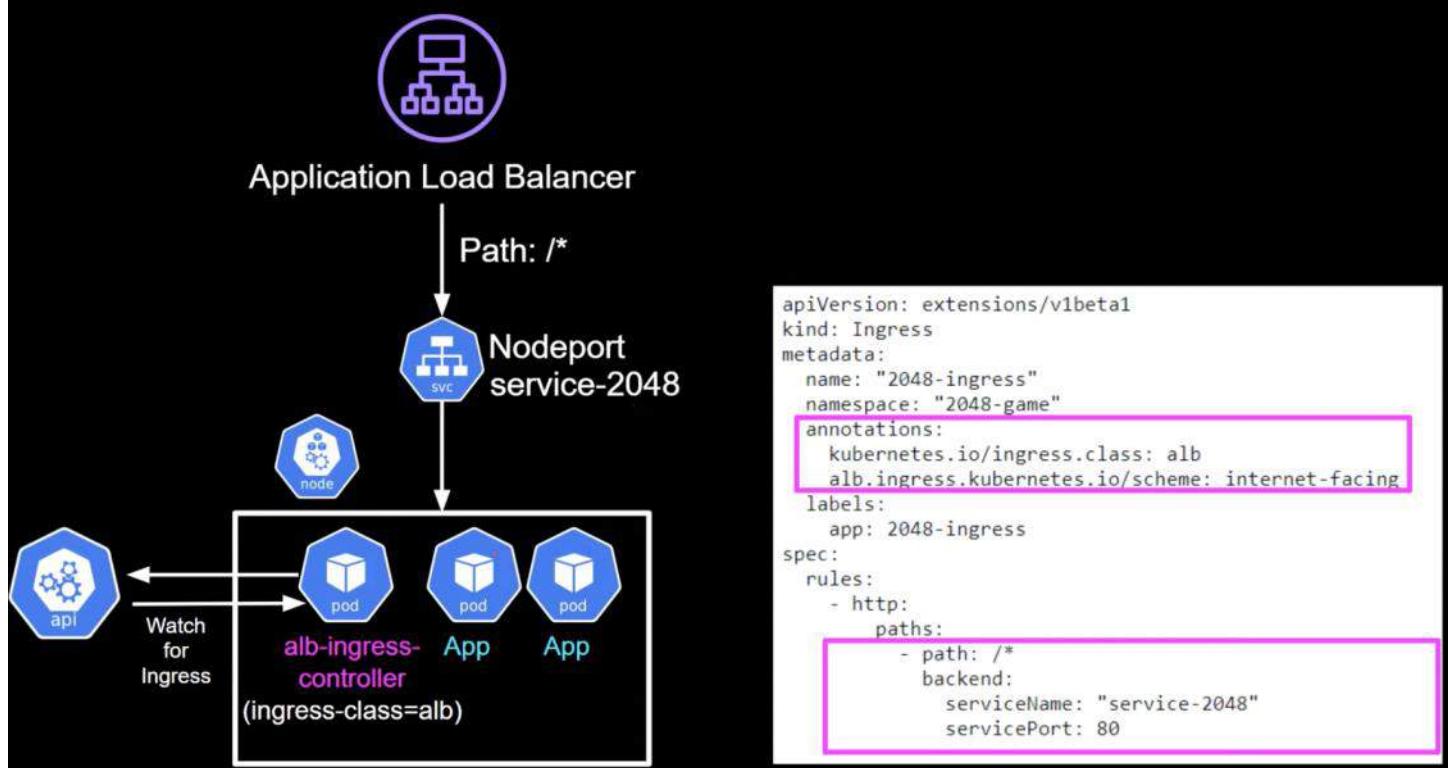
# Application Load Balancer (ALB) Ingress Controller Deployment Manifest.
# This manifest details sensible defaults for deploying an ALB Ingress Controller.
# Github: https://github.com/kubernetes-sigs/aws-alb-ingress-controller

```

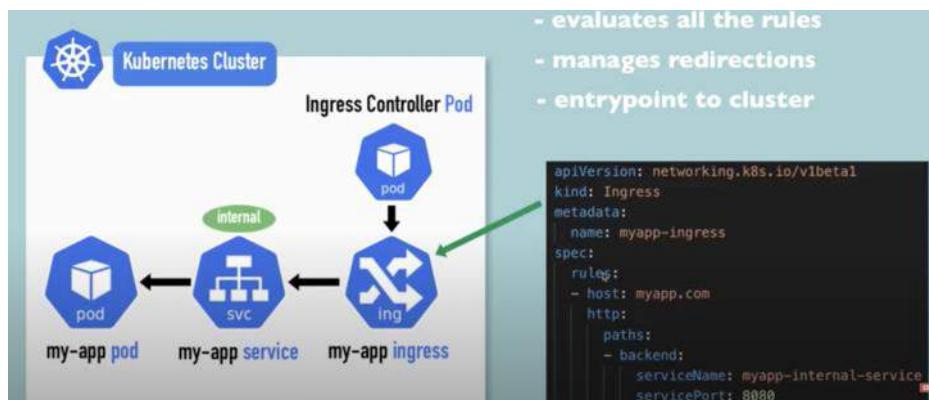
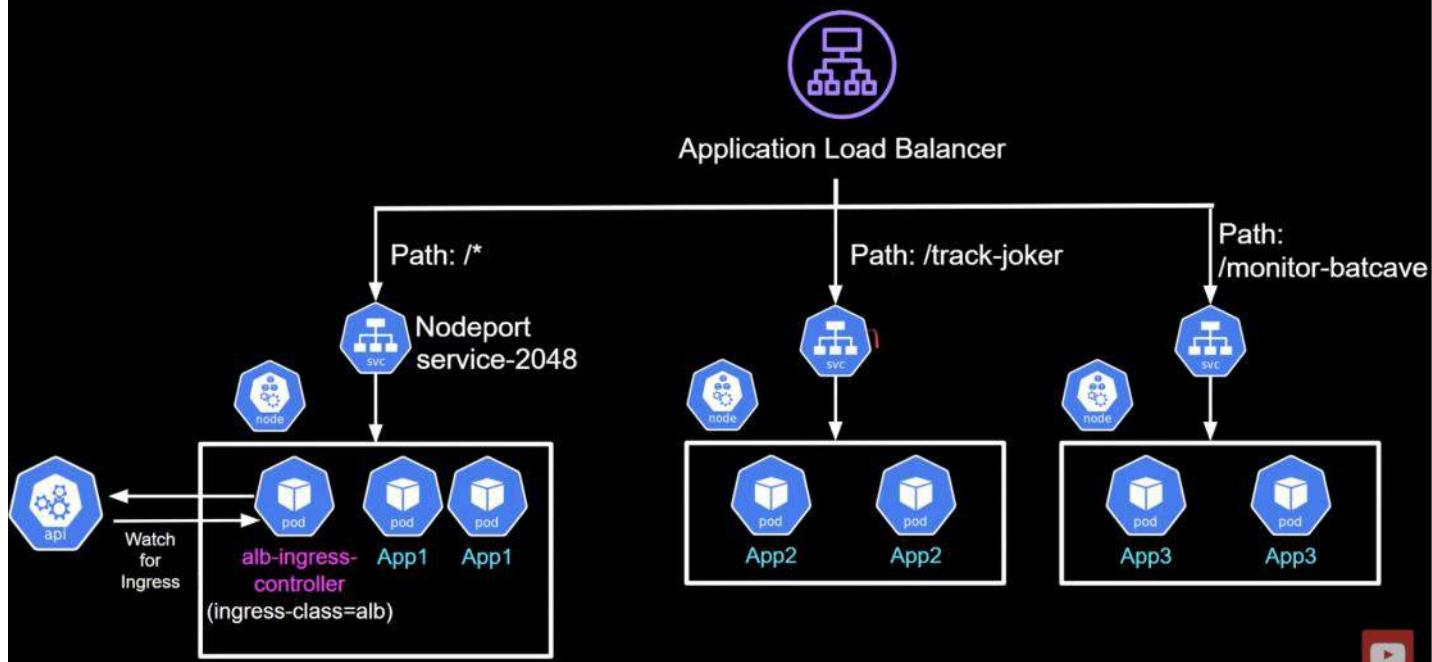
# ALB Ingress Controller



## Ingress resource



# ALB Ingress



```
1 apiVersion: networking.k8s.io/v1
2 kind: Ingress
3 metadata:
4   name: dashboard-ingress
5   namespace: kubernetes-dashboard
6 annotations:
7   kubernetes.io/ingress.class: "nginx"
8 spec:
9   rules:
10    - host: dashboard.com
11      http:
12        paths:
13          - path: /
14            pathType: Exact
15        backend:
16          service:
17            name: kubernetes-dashboard
18            port:
19              number: 80
```

## Multiple sub-domains or domains

```
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: name-virtual-host-ingress
spec:
  rules:
  - host: analytics.myapp.com
    http:
      paths:
        backend:
          serviceName: analytics-service
          servicePort: 3000
  - host: shopping.myapp.com
    http:
      paths:
        backend:
          serviceName: shopping-service
          servicePort: 8080
```

http://analytics.myapp.com



analytics service



analytics pod

# TLS

Monday, 24 April 2023 11:33 PM

## Configuring TLS Certificate - https://

The screenshot shows two code snippets side-by-side. On the left is an Ingress configuration:

```
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: tls-example-ingress
spec:
  tls:
    - hosts:
      - myapp.com
      secretName: myapp-secret-tls
  rules:
    - host: myapp.com
      http:
        paths:
          - path: /
            backend:
              serviceName: myapp-internal-service
              servicePort: 8080
```

On the right is a Secret configuration:

```
apiVersion: v1
kind: Secret
metadata:
  name: myapp-secret-tls
  namespace: default
data:
  tls.crt: base64 encoded cert
  tls.key: base64 encoded key
type: kubernetes.io/tls
```

A white arrow points from the 'secretName' field in the Ingress configuration to the 'name' field in the Secret configuration.

## Configuring TLS Certificate - https://

The screenshot shows a Secret configuration:

```
apiVersion: v1
kind: Secret
metadata:
  name: myapp-secret-tls
  namespace: default
data:
  tls.crt: base64 encoded cert
  tls.key: base64 encoded key
type: kubernetes.io/tls
```

Three numbered annotations are overlaid on the right side:

- 1) Data keys need to be "tls.crt" and "tls.key"
- 2) Values are file contents  
NOT file paths/locations
- 3) Secret component must be in the same namespace as the Ingress component

# Configuring TLS Certificate - https://

```
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: tls-example-ingress
spec:
  tls:
    - hosts:
      - myapp.com
      secretName: myapp-secret-tls
  rules:
    - host: myapp.com
      http:
        paths:
          - path: /
            backend:
              serviceName: myapp-internal-service
              servicePort: 8080
```

# Pod

Monday, 24 April 2023 9:28 PM

# PVC

Monday, 24 April 2023 9:28 PM

# YAML K8S

Tuesday, 5 July 2022 5:48 PM

## Get all info

```
# kubectl get all
```

## Services

```
# kubectl get service hellworldexample-helloworld -n default -o yaml > service.yaml
```

## All Service resources

```
# kubectl get service --all-namespaces -o yaml > all-service.yaml
```

## Deployment

```
# kubectl get deployment myreleasename-helloworld -n default -o yaml > deployment.yaml
```

## All Deployment

```
# kubectl get deploy --all-namespaces -o yaml > all-deployment.yaml
```

## Bash script for all

```
for n in $(kubectl get -o=name pvc,configmap,serviceaccount,secret,ingress,service,deployment,statefulset,hpa,job,cronjob)
do
  mkdir -p $(dirname $n)
  kubectl get -o=yaml $n > $n.yaml
done
```

# Security

Monday, 24 April 2023 11:40 PM

## **RBAC (Role-Based Access Control):**

Kubernetes provides RBAC mechanisms to control access to cluster resources. Define roles, role bindings, and service accounts to enforce proper access controls.

## **Network Policies:**

Kubernetes network policies allow you to define and enforce communication rules between pods and namespaces. Be familiar with how to create network policies to restrict traffic and secure network communications within your cluster.

## **Pod Security Policies (PSP):**

PSPs define a set of security requirements that pods must adhere to. Know how to configure and apply PSPs to restrict the usage of privileged containers, host namespaces, and other potentially insecure features.

## **Container Image Security:**

Ensure that you use only trusted and verified container images from reputable sources.

Regularly update your container images to address any known vulnerabilities.

Use vulnerability scanning tools to detect and mitigate potential security issues.

## **Secrets Management:**

Understand how to securely manage sensitive information like passwords, API keys, and TLS certificates in Kubernetes.

Avoid storing secrets directly in manifests and instead utilize Kubernetes Secret objects or external secret management tools.

## **Monitoring and Logging:**

Implement robust monitoring and logging solutions to detect and investigate any security incidents within your Kubernetes cluster.

Utilize tools like Prometheus and Grafana to monitor cluster health, and centralize logs for analysis using tools such as Elasticsearch and Kibana.

## **Secure Image Pulling:**

Ensure that only authorized entities can pull images from your container registry.

Use authentication mechanisms, such as image pull secrets or integrated registry authentication, to control access to images.

## **Regular Updates and Patching:**

Stay updated with the latest Kubernetes releases and security patches.

Regularly apply updates to your cluster to address any security vulnerabilities and ensure you have a secure and stable environment.

## **Resource Limits and Quotas:**

Implement resource limits and quotas for pods and namespaces to prevent resource exhaustion and potential denial-of-service attacks.

## **Backup and Disaster Recovery:**

Establish regular backup and disaster recovery mechanisms to protect your Kubernetes cluster and its data.

Regularly test the backup and restore procedures to ensure their effectiveness.

# Troubleshooting

Friday, 15 July 2022 11:14 PM

<https://itecnote.com/tecnote/1-nodes-had-taints-that-the-pod-didnt-tolerate-in-kubernetes-cluster/>

<https://github.com/calebhailey/homelab/issues/3>

[https://www.mirantis.com/cloud-native-concepts/getting-started-with-kubernetes/what-is-kubernetes-management/?utm\\_source=google&utm\\_medium=paidsearch&utm\\_campaign=16880236533&utm\\_adgroup=138553948514&utm\\_content=598619916634&utm\\_term=kubernetes%20software&utm\\_region=apac&utm\\_focus=non-brand&gclid=CjwKCAjwoMSWBhAdEiwAVJ2ndmr8hUECJxPqXkg3om49tM2Dqb2oTdewuiu AD4KOAP3seOzwmB4IRoCZaEQAvD\\_BwE](https://www.mirantis.com/cloud-native-concepts/getting-started-with-kubernetes/what-is-kubernetes-management/?utm_source=google&utm_medium=paidsearch&utm_campaign=16880236533&utm_adgroup=138553948514&utm_content=598619916634&utm_term=kubernetes%20software&utm_region=apac&utm_focus=non-brand&gclid=CjwKCAjwoMSWBhAdEiwAVJ2ndmr8hUECJxPqXkg3om49tM2Dqb2oTdewuiu AD4KOAP3seOzwmB4IRoCZaEQAvD_BwE)

## PV-PVC

<https://medium.com/avmconsulting-blog/persistent-volumes-pv-and-claims-pvc-in-kubernetes-bd76923a61f6>

<https://kubernetes.io/docs/concepts/storage/persistent-volumes/#claims-as-volumes>

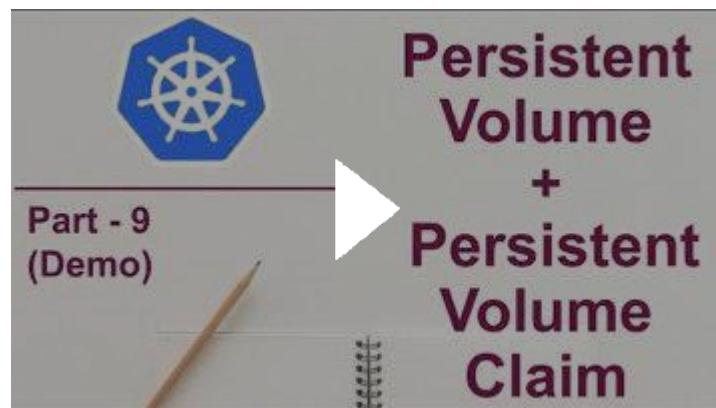
<https://cloud.netapp.com/blog/kubernetes-persistent-storage-why-where-and-how>

## Wordpress with Mysql

<https://kubernetes.io/docs/tutorials/stateful-application/mysql-wordpress-persistent-volume/>

## Spring boot App with MySQL

[How to use Persistent Volume and Persistent Claims | Kubernetes - Part 9](#)



# Upgrading K8s with kubeadm

Tuesday, September 21, 2021 9:38 PM

## Upgrading K8s with kubeadm

### Relevant Documentation

- [Upgrading kubeadm Clusters](#)

### Lesson Reference

First, upgrade the control plane node.

Drain the control plane node.

```
kubectl drain <control plane node name> --ignore-daemonsets  
Upgrade kubeadm.
```

```
sudo apt-get update && \  
sudo apt-get install -y --allow-change-held-packages kubeadm=1.22.2-00  
kubeadm version
```

Plan the upgrade.

```
sudo kubeadm upgrade plan v1.22.2  
Upgrade the control plane components.
```

```
sudo kubeadm upgrade apply v1.22.2  
Upgrade kubelet and kubectl on the control plane node.
```

```
sudo apt-get update && \  
sudo apt-get install -y --allow-change-held-packages kubelet=1.22.2-00 kubectl=1.22.2-00  
Restart kubelet.
```

```
sudo systemctl daemon-reload  
sudo systemctl restart kubelet
```

Uncordon the control plane node.

```
kubectl uncordon <control plane node name>  
Verify that the control plane is working.
```

```
kubectl get nodes  
Upgrade the worker nodes.
```

Note: In a real-world scenario, you should not perform upgrades on all worker nodes at the same time. Make sure enough nodes are available at any given time to provide uninterrupted service.

Run the following on the control plane node to drain worker node 1:

```
kubectl drain <worker 1 node name> --ignore-daemonsets --force
```

Log in to the first worker node, then Upgrade kubeadm.

```
sudo apt-get update && \  
sudo apt-get install -y --allow-change-held-packages kubeadm=1.22.2-00  
kubeadm version
```

Upgrade the kubelet configuration on the worker node.

```
sudo kubeadm upgrade node
```

```
Upgrade kubelet and kubectl on the worker node 1
```

```
sudo apt-get update && \  
-----
```

```
sudo apt-get install -y --allow-change-held-packages kubelet=1.22.2-00 kubectl=1.22.2-00
Restart kubelet.
```

```
sudo systemctl daemon-reload
sudo systemctl restart kubelet
```

From the control plane node, uncordon worker node 1.

```
kubectl uncordon <worker 1 node name>
```

Repeat the upgrade process for worker node 2.

From the control plane node, drain worker node 2.

```
kubectl drain <worker 2 node name> --ignore-daemonsets --force
On the second worker node, upgrade kubeadm.
```

```
sudo apt-get update && \
sudo apt-get install -y --allow-change-held-packages kubeadm=1.22.2-00
kubeadm version
```

Perform the upgrade on worker node 2.

```
sudo kubeadm upgrade node
sudo apt-get update && \
sudo apt-get install -y --allow-change-held-packages kubelet=1.22.2-00 kubectl=1.22.2-00
sudo systemctl daemon-reload
sudo systemctl restart kubelet
```

From the control plane node, uncordon worker node 2.

```
kubectl uncordon <worker 2 node name>
```

Verify that the cluster is upgraded and working.

```
kubectl get nodes
```

From <<https://linuxacademy.com/cp/courses/lesson/course/7948/lesson/5/module/748>>

# Helm Charts

Tuesday, August 31, 2021 10:39 PM

Helm is a Kubernetes package and operations manager.

Using a packaging manager, Charts, Helm allows us to package Kubernetes releases into a convenient zip (.tgz) file. A Helm chart can contain any number of Kubernetes objects, all of which are deployed as part of the chart. A Helm chart will usually contain at least a Deployment and a Service, but it can also contain an Ingress, Persistent Volume Claims, or any other Kubernetes object.

Install and Uninstall Apps		Add, Remove, and Update Repos		Plugin Management		
<code>helm install [name] [chart]</code>	Install an app	<code>helm repo add [name] [url]</code>	Add a repository from the internet	<code>helm plugin install [path/url1] [path/url2] ...</code>	Install plugins	
<code>helm install [name] [chart] --namespace [namespace]</code>	Install an app in a specific namespace	<code>helm repo remove [name]</code>	Remove a repository from your system	<code>helm plugin list</code>	View a list of all the installed plugins	
<code>helm install [name] [chart] --values [yaml-file/url]</code>	Override the default values with those specified in a file of your choice	<code>helm repo update</code>	Update repositories	<code>helm plugin update [plugin1] [plugin2] ...</code>	Update plugins	
<code>helm install [name] --dry-run --debug</code>	Run a test install to validate and verify the chart	<th data-cs="2" data-kind="parent">List and Search Repos</th> <th data-kind="ghost"></th> <td><code>helm plugin uninstall [plugin]</code></td> <td>Uninstall a plugin</td>	List and Search Repos		<code>helm plugin uninstall [plugin]</code>	Uninstall a plugin
<code>helm uninstall [release] name</code>	Uninstall a release	<code>helm repo list</code>	List chart repositories	Chart Management		
Perform App Upgrade and Rollback		<code>helm repo index</code>	Generate an index file containing charts found in the current directory	<code>helm create [name]</code>	Create a directory containing the common chart files and directories (Chart.yaml, values.yaml, charts/ and templates/)	
<code>helm upgrade [release] [chart]</code>	Upgrade an app	<code>helm search [keyword]</code>	Search charts for a keyword	<code>helm package [chart-path]</code>	Package a chart into a chart archive	
<code>helm upgrade [release] [chart] --atomic</code>	Tell Helm to roll back changes if the upgrade fails	<code>helm search repo [keyword]</code>	Search repositories for a keyword	<code>helm lint [chart]</code>	Run tests to examine a chart and identify possible issues	
<code>helm upgrade [release] [chart] --install</code>	Upgrade a release. If it does not exist on the system, install it	<code>helm search hub [keyword]</code>	Search Helm Hub	<code>helm show all [chart]</code>	Inspect a chart and list its contents	
<code>helm upgrade [release] [chart] --version [version-number]</code>	Upgrade to a version other than the latest one	Release Monitoring		<code>helm show chart [chart]</code>	Display the chart's definition	
<code>helm rollback [release] [revision]</code>	Roll back a release	<code>helm list</code>	List all the available releases in the current namespace	<code>helm show values [chart]</code>	Display the chart's values	
Download Release Information		<code>helm list --all-namespaces</code>	List all the available releases across all namespaces	<code>helm pull [chart]</code>	Download a chart	
<code>helm get all [release]</code>	Download all the release information	<code>helm list --namespace [namespace]</code>	List all the releases in a specific namespace	<code>helm pull [chart] --untardir [directory]</code>	Download a chart and extract the archive's contents into a directory	
<code>helm get hooks [release]</code>	Download all hooks	<code>helm list --output [format]</code>	List all the releases in a specific output format	<code>helm dependency list [chart]</code>	Display a list of a chart's dependencies	
<code>helm get manifest [release]</code>	Download the manifest	<code>helm list --filter [expression]</code>	Apply a filter to the list of releases using regular (Pearl compatible) expressions	Get Help and Version Info		
<code>helm get notes [release]</code>	Download the notes	<code>helm status [release]</code>	See the status of a release	<code>helm --help</code>	See the general help for Helm	
<code>helm get values [release]</code>	Download the values file	<code>helm history [release]</code>	See the release history	<code>helm [command] --help</code>	See help for a particular command	
<code>helm history [release]</code>	Fetch release history	<code>helm env</code>	See information about the Helm client environment	<code>helm version</code>	See the installed version of Helm	



# Container runtime

Thursday, 13 April 2023 1:16 PM

## “[ERROR CRI]: container runtime is not running: output:” Code Answer

[ERROR CRI]: container runtime is not running: output:

shell by [devops unicorn](#) on May 14 2022 [Comment](#)

0

1 rm -rf /etc/containerd/config.toml

2 systemctl restart containerd

3 kubeadm init

# On CentOS

Wednesday, 27 July 2022 12:42 PM

```
# Disable firewall & selinux
sed -i --follow-symlinks 's/SELINUX=enforcing/SELINUX=disabled/g' /etc/sysconfig/selinux
systemctl stop firewalld
systemctl disable firewalld

# Disable swap
swapoff -a
nano /etc/fstab

# Docker Installation
curl -fsSL https://get.docker.com -o get-docker.sh; sh get-docker.sh
systemctl start docker; systemctl enable docker
curl -L "https://github.com/docker/compose/releases/download/1.26.2/docker-compose-\$\(uname -s\)-\$\(uname -m\)" -o
/usr/local/bin/docker-compose; chmod +x /usr/local/bin/docker-compose; ln -s /usr/local/bin/docker-compose /usr/bin/docker-
compose; docker-compose --version

# Add repo
cat <<EOF > /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86_64
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
EOF

# Install K8s packages
yum install kubeadm kubelet kubectl -y
systemctl start kubelet; systemctl enable kubelet

rm -rf /etc/containerd/config.toml
systemctl restart containerd
systemctl enable containerd

kubeadm init

mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config

kubectl get nodes

export kubever=$(kubectl version | base64 | tr -d '\n')
kubectl apply -f "https://cloud.weave.works/k8s/net?k8s-version=\$kubever"
kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml

kubectl apply -f https://github.com/weaveworks/weave/releases/download/v2.8.1/weave-daemonset-k8s.yaml

kubectl get nodes
```

# Patching

Friday, 3 February 2023 11:46 AM

# Security Updates - Linux

Friday, June 19, 2020 3:51 PM

## Is it possible to limit yum so that it lists or installs only security updates?

### Environment

- ① Red Hat Enterprise Linux 8.x
- ① Red Hat Enterprise Linux 7.x
- ① Red Hat Enterprise Linux 6.x
- ① Red Hat Enterprise Linux 5.1 and later
- ① Red Hat Network Hosted
- ① Red Hat Satellite

### Issue

- ① Is it possible to limit yum so that it lists or installs only security updates?
- ① How to update a system using yum and only apply security errata?
- ① Wanted to update security patches without modifying OS version. How can we do that?
- ① How to patch the system only with security erratas ?

### Resolution

- ① Install the yum-security plugin

It is now possible to limit yum to install only security updates (as opposed to bug fixes or enhancements) using Red Hat Enterprise Linux 5,6, and 7. To do so, simply install the yum-security plugin:

**For Red Hat Enterprise Linux 7 and 8** The plugin is already a part of yum itself, no need to install anything.

**For Red Hat Enterprise Linux 6**

Raw  
# yum install yum-plugin-security

**For Red Hat Enterprise Linux 5**

Raw  
# yum install yum-security  
Alternatively, download the yum-security package from the Red Hat Network (RHN) and manually install it on the system.

**For Red Hat Enterprise Linux 6, 7 & 8**

- ① Using yum-security plugin
- ① To list all available erratas without installing them, run:

Raw  
# yum updateinfo list available  
① To list all available security updates without installing them, run:

Raw  
# yum updateinfo list security all  
# yum updateinfo list sec  
① To get a list of the currently installed security updates this command can be used:

Raw  
# yum updateinfo list security installed

## For Red Hat Enterprise Linux 5

① Using yum-security plugin

② To list all available erratas without installing them, run:

Raw

```
# yum list-sec
```

③ To list all available security updates without installing them, run:

Raw

```
# yum list-security --security
```

## For Red Hat Enterprise Linux 5, 6, 7 and 8

① To list all available security updates with verbose descriptions of the issues they apply to:

Raw

```
# yum info-sec
```

② Run the following command to download and apply all available security updates from Red Hat Network hosted or Red Hat Network Satellite:

Raw

```
# yum -y update --security
```

**NOTE:** It will install the last version available of any package with at least one security errata thus can install non-security erratas if they provide a more updated version of the package.

③ To only install the packages that have a security errata use

Raw

```
# yum update-minimal --security -y
```

④ yum-security also allows installing security updates based on the CVE reference of the issue. To install a security update using a CVE reference run:

Raw

```
# yum update --cve <CVE>
```

e.g.

Raw

```
# yum update --cve CVE-2008-0947
```

Viewing available advisories by severities:

Raw

```
# yum updateinfo list
```

This system is receiving updates from RHN Classic or RHN Satellite.

```
RHSA-2014:0159 Important/Sec. kernel-headers-2.6.32-431.5.1.el6.x86_64
```

```
RHSA-2014:0164 Moderate/Sec. mysql-5.1.73-3.el6_5.x86_64
```

```
RHSA-2014:0164 Moderate/Sec. mysql-devel-5.1.73-3.el6_5.x86_64
```

```
RHSA-2014:0164 Moderate/Sec. mysql-libs-5.1.73-3.el6_5.x86_64
```

```
RHSA-2014:0164 Moderate/Sec. mysql-server-5.1.73-3.el6_5.x86_64
```

```
RHBA-2014:0158 bugfix nss-sysinit-3.15.3-6.el6_5.x86_64
```

```
RHBA-2014:0158 bugfix nss-tools-3.15.3-6.el6_5.x86_64
```

If you want to apply only one specific advisory:

Raw

```
# yum update --advisory=RHSA-2014:0159
```

However, if you would like to know more information about this advisory before to apply it:

Raw

```
# yum updateinfo RHSA-2014:0159
```

Similarly, you can view CVEs which affect the system with:

Raw

```
# yum updateinfo list cves
```

Loaded plugins: auto-update-debuginfo, product-id, search-disabled-repos, subscription-manager

```
CVE-2017-1000380 Moderate/Sec. kernel-3.10.0-693.11.1.el7.x86_64
```

```
CVE-2017-1000380 Moderate/Sec. kernel-devel-3.10.0-693.11.1.el7.x86_64
```

```
CVE-2017-1000380 Moderate/Sec. kernel-headers-3.10.0-693.11.1.el7.x86_64
```

```
CVE-2017-1000380 Moderate/Sec. kernel-tools-3.10.0-693.11.1.el7.x86_64
CVE-2017-1000380 Moderate/Sec. kernel-tools-libs-3.10.0-693.11.1.el7.x86_64
CVE-2017-1000380 Moderate/Sec. perf-3.10.0-693.11.1.el7.x86_64
CVE-2017-1000380 Moderate/Sec. python-perf-3.10.0-693.11.1.el7.x86_64
CVE-2016-10002 Moderate/Sec. squid-7.3.5.20-2.el7_3.2.x86_64
updateinfo list done
```

For more commands consult the manual pages of yum-security with

```
Raw  
# man yum-security
```

If you face any missing dependency issue while applying security patches on system then refer to yum update --security fails with missing dependency errors.

# Patching on Linux cluster system

Thursday, 2 February 2023 5:48 PM

Put the system in downtime in ground work

Check remote console working

Run supportscripts

Run sosreport

Run /opt/lenovo/saphana/bin/

```
watch -n 5 '/usr/sap/hostctrl/exe/sapcontrol -nr 00 -function GetSystemInstanceList'
```

```
/usr/sap/hostctrl/exe/sapcontrol -nr 00 -function GetSystemInstanceList;/usr/sap/hostctrl/exe/sapcontrol -nr 00 -function GetProcessList
```

```
[root@gsrhdrp1 Utilities]# cd /opt/lenovo/saphana/bin/  
[root@gsrhdrp1 Utilities]# ./saphana-support-lenovo.sh
```

```
[root@gswp2 bin]# pcs status
```

Cluster name: rhelhanapcs

Stack: corosync

Current DC: gswp1 (version 1.1.16-12.el7\_4.2-94ff4df) - partition with quorum

Last updated: Tue Apr 13 04:13:11 2021

Last change: Tue Apr 13 04:12:35 2021 by root via crm\_attribute on gswp1

2 nodes configured

7 resources configured

Online: [ gswp1 gswp2 ]

Full list of resources:

```
rsc_ip_SAPHana_HBP_HDB00 (ocf::heartbeat:IPaddr2): Started gswp1  
Clone Set: rsc_SAPHanaTopology_HBP_HDB00-clone [rsc_SAPHanaTopology_HBP_HDB00]  
    Started: [ gswp1 gswp2 ]  
Master/Slave Set: rsc_SAPHana_HBP_HDB00-master [rsc_SAPHana_HBP_HDB00]  
    Masters: [ gswp1 ]  
    Slaves: [ gswp2 ]  
st_ipmi_gswp1 (stonith:fence_ipmilan): Started gswp1  
st_ipmi_gswp2 (stonith:fence_ipmilan): Started gswp1
```

Daemon Status:

corosync: active/disabled

pacemaker: active/disabled

pcsd: active/enabled

```
[root@gswp2 bin]#
```

Greetings,

Additionally for gsrhq for staging, it appears that there is a configuration issue when trying to reach the RHEL repositories. Here is the output that I receive when running the command "yum subscription status" and then "yum check-update". Please investigate this and resolve before the change early next week. Thanks.

```
[root@gsrhqa ~]# subscription-manager status  
+-----+  
| System Status Details |  
+-----+  
Overall Status: Invalid  
Red Hat Enterprise Linux Server:  
- Not supported by a valid subscription.
```

---

```
[root@gsrhqa ~]# yum check-update  
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager, versionlock  
Network error code: 400  
Repository 'RHEL' is missing name in configuration, using id  
file:///RHEL/repo/repodata/repomd.xml: [Errno 14] curl#37 - "Couldn't open file /RHEL/repo/repodata/repomd.xml"  
Trying other mirror.
```

One of the configured repositories failed (Red Hat Enterprise Linux 7.2), and yum doesn't have enough cached data to continue. At this point the only safe thing yum can do is fail. There are a few ways to work "fix" this:

1. Contact the upstream for the repository and get them to fix the problem.
2. Reconfigure the baseurl/etc. for the repository, to point to a working upstream. This is most often useful if you are using a newer distribution release than is supported by the repository (and the packages for the previous distribution release still work).
3. Run the command with the repository temporarily disabled  
    yum --disablerrepo=InstallMedia ...
4. Disable the repository permanently, so yum won't use it by default. Yum will then just ignore the repository until you permanently enable it again or use --enablerrepo for temporary usage:  
    yum-config-manager --disable InstallMedia  
or

```

subscription-manager repos --disable=InstallMedia
5. Configure the failing repository to be skipped, if it is unavailable.
Note that yum will try to contact the repo. when it runs most commands,
so will have to try and fail each time (and thus, yum will be much
slower). If it is a very temporary problem though, this is often a nice
compromise:
    yum-config-manager --save --setopt=InstallMedia.skip_if_unavailable=true
failure: repodata/repo.xml from InstallMedia: [Errno 256] No more mirrors to try.
file:///RHEL/repo/repodata/repomd.xml: [Errno 14] curl#37 - "Couldn't open file /RHEL/repo/repodata/repomd.xml"

```

```

Cd LMS/Updates/Utilities
Ls -l
Rpm -Uvh lenovo-sapahana-support <version>.rpm
Rpm -qa | grep support
Rpm -Uvh storcli
Rpm -qa | grep storcli

```

Kindly copy the validated output at the bottom of the CR form

```

[root@gswp2 megaraid]# yum repolist
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is registered with an entitlement server, but is not receiving updates. You can use subscription-manager to assign subscriptions.

```

```

* WARNING *
The subscription for following product(s) has expired:
- Oracle Java (for RHEL Server)
- Red Hat Ansible Engine
- Red Hat Beta
- Red Hat CloudForms
- Red Hat CloudForms Beta
- Red Hat CodeReady Linux Builder for x86_64
- Red Hat CodeReady Linux Builder for x86_64 - Extended Update Support
- Red Hat Enterprise Linux Atomic Host
- Red Hat Enterprise Linux Fast Datapath
- Red Hat Enterprise Linux Fast Datapath Beta for x86_64
- Red Hat Enterprise Linux Server
- Red Hat Enterprise Linux for x86_64
- Red Hat Enterprise Linux for x86_64 - Extended Update Support
- Red Hat OpenShift Container Platform
- Red Hat OpenShift Enterprise Client Tools
- Red Hat OpenShift Enterprise Infrastructure
- Red Hat OpenShift Service Mesh
- Red Hat Software Collections (for RHEL Server)
- Red Hat Software Collections Beta (for RHEL Server)
- dotNET on RHEL (for RHEL Server)
- dotNET on RHEL Beta (for RHEL Server)
You no longer have access to the repositories that provide these products. It is important that you apply an active subscription in order to resume access to security and other critical updates. If you don't have other active subscriptions, you can renew the expired subscription.
repolist: 0
[root@gswp2 megaraid]#

```

```

[root@gswp2 megaraid]# subscription-manager release --show
Traceback (most recent call last):
File "/sbin/subscription-manager", line 9, in <module>
    load_entry_point('subscription-manager==1.21.10', 'console_scripts', 'subscription-manager')()
File "/usr/lib64/python2.7/site-packages/subscription_manager/scripts/subscription_manager.py", line 85, in main
    return managercli.ManagerCLI().main()
File "/usr/lib64/python2.7/site-packages/subscription_manager/managercli.py", line 2667, in main
    ret = CLI.main(self)
File "/usr/lib64/python2.7/site-packages/subscription_manager/cli.py", line 183, in main
    return cmd.main()
File "/usr/lib64/python2.7/site-packages/subscription_manager/managercli.py", line 501, in main
    return_code = self._do_command()
File "/usr/lib64/python2.7/site-packages/subscription_manager/managercli.py", line 1456, in _do_command
    self.show_current_release()
File "/usr/lib64/python2.7/site-packages/subscription_manager/managercli.py", line 1393, in show_current_release
    release = self._get_consumer_release()
File "/usr/lib64/python2.7/site-packages/subscription_manager/managercli.py", line 1387, in _get_consumer_release
    consumer = self.cp.getConsumer(self.identity.uuid)
File "/usr/lib64/python2.7/site-packages/rhsm/connection.py", line 1105, in getConsumer
    return self.conn.request_get(method)
File "/usr/lib64/python2.7/site-packages/rhsm/connection.py", line 693, in request_get
    return self._request("GET", method, headers=headers)
File "/usr/lib64/python2.7/site-packages/rhsm/connection.py", line 719, in _request
    info=info, headers=headers)
File "/usr/lib64/python2.7/site-packages/rhsm/connection.py", line 597, in _request
    self.validateResponse(result, request_type, handler)
File "/usr/lib64/python2.7/site-packages/rhsm/connection.py", line 680, in validateResponse
    raise NetworkException(response['status'])
rhsm.connection.NetworkException: HTTP error (400 - Bad Request)

```

```

[root@gswp2 megaraid]# subscription-manager status
+-----+
System Status Details

```

```

+-----+
Overall Status: Insufficient

Red Hat OpenShift Container Platform Broker/Master Infrastructure:
- Only supports 2 of 88 cores.

[root@gswp2 megaraid]# subscription-manager list
+-----+
   Installed Product Status
+-----+
Product Name: Red Hat Enterprise Linux Server
Product ID: 69
Version: 7.6
Arch: x86_64
Status: Partially Subscribed
Status Details:
Starts: 09/14/2019
Ends: 09/14/2020

subscription-manager status
subscription-manager release --show
subscription-manager release --list
subscription-manager release --unset
subscription-manager remove --all
subscription-manager release --set=7.6
subscription-manager release --show
subscription-manager repos --disable=InstallMedia
subscription-manager attach --auto
subscription-manager repos --list-enabled

yum clean all
yum versionlock clear
yum check-update
yum upgrade --> if produces error, please follow next step
yum upgrade --skip-broken

[root@gswp2 ~]# cd LMS/Updates/Q1-2021/Drivers/megaraid/
[root@gswp2 megaraid]# pwd
/root/LMS/Updates/Q1-2021/Drivers/megaraid

[root@gswp2 megaraid]# tar zxvf Invgy_dd_raid_mr3-07.712.02.00-0_rhel7_x86-64.tgz

[root@gswp2 megaraid]# ls
disks  License_gpl.txt          Invgy_dd_raid_mr3-07.712.02.00-0_rhel7_x86-64.tgz  Invgy_dd_raid_mr3-07.712.02.00-0_rhel7_x86-64.xml  SRPMS
install.sh  Invgy_dd_raid_mr3-07.712.02.00-0_rhel7_x86-64.chg  Invgy_dd_raid_mr3-07.712.02.00-0_rhel7_x86-64.txt  RPMS

[root@gswp2 megaraid]# rpmbuild --rebuild SRPMS/kmod-megaraid_sas-07.712.02.00-1.src.rpm

[root@gswp2 megaraid]# cd ..//emulex/

[root@gswp2 emulex]# pwd
/root/LMS/Updates/Q1-2021/Drivers/emulex

[root@gswp2 emulex]# tar zxvf elx-Invgy_dd_fc_lp-12.6.221.21-2_rhel7_x86-64.tgz
./
./RPMS/
./RPMS/redhat-release-server-7.8/
./RPMS/redhat-release-server-7.8/elx-vector-map-1-1.rhel7u8.x86_64.rpm
./RPMS/redhat-release-server-7.8/elx-nvmefc-connect-12.6.61.0-1.rhel7u8.noarch.rpm
./RPMS/redhat-release-server-7.8/kmod-elx-lpfc-12.6.221.21-1.rhel7u8.x86_64.rpm
./RPMS/redhat-release-server-7.7/
./RPMS/redhat-release-server-7.7/elx-nvmefc-connect-12.6.61.0-1.rhel7u7.noarch.rpm
./RPMS/redhat-release-server-7.7/kmod-elx-lpfc-12.6.221.21-1.rhel7u7.x86_64.rpm
./RPMS/redhat-release-server-7.7/elx-vector-map-1-1.rhel7u7.x86_64.rpm
./RPMS/redhat-release-server-7.6/
./RPMS/redhat-release-server-7.6/elx-nvmefc-connect-12.6.61.0-1.rhel7u6.noarch.rpm
./RPMS/redhat-release-server-7.6/kmod-elx-lpfc-12.6.221.21-1.rhel7u6.x86_64.rpm
./RPMS/redhat-release-server-7.6/elx-vector-map-1-1.rhel7u6.x86_64.rpm
./install.sh
./disks/
./disks/elx-lpfc-12.6.221.21.rhel7u8.x86_64.iso
./disks/elx-lpfc-12.6.221.21.rhel7u6.x86_64.iso
./disks/elx-lpfc-12.6.221.21.rhel7u7.x86_64.iso
./SRPM/
./SRPM/redhat-release-server-7.8/
./SRPM/redhat-release-server-7.8/elx-lpfc-12.6.221.21-1.rhel7u8.src.rpm
./SRPM/redhat-release-server-7.7/
./SRPM/redhat-release-server-7.7/elx-lpfc-12.6.221.21-1.rhel7u7.src.rpm
./SRPM/redhat-release-server-7.6/
./SRPM/redhat-release-server-7.6/elx-lpfc-12.6.221.21-1.rhel7u6.src.rpm
[root@gswp2 emulex]#
```

Below rpmbuild command will fail, so do not worry about it.

```
[root@gswp2 emulex]# rpmbuild --rebuild SRPMS/redhat-release-server-7.6/elx-lpfc-12.6.221.21-1.rhel7u6.src.rpm
```

Install.sh script may also fail because of the dependency issues, do not worry

```
[root@gswp2 emulex]# ./install.sh
[root@gswp2 emulex]# cd RPMS/redhat-release-server-7.6/
[root@gswp2 redhat-release-server-7.6]# pwd
/root/LMS/Updates/Q1-2021/Drivers/emulex/RPMS/redhat-release-server-7.6
[root@gswp2 redhat-release-server-7.6]# ls
elx-nvmefc-connect-12.6.61.0-1.rhel7u6.noarch.rpm elx-vector-map-1-1.rhel7u6.x86_64.rpm kmod-elx-lpfc-12.6.221.21-1.rhel7u6.x86_64.rpm

Below command will take long time to update, keep your session active/alive
[root@gswp2 redhat-release-server-7.6]# rpm -Uvh kmod-elx-lpfc-12.6.221.21-1.rhel7u6.x86_64.rpm
[root@gsrhdp1 redhat-release-server-7.6]# rpm -qa kmod-elx-lpfc
kmod-elx-lpfc-12.6.221.21-1.rhel7u6.x86_64

[root@gswp2 redhat-release-server-7.6]# cd ..
[root@gswp2 RPMS]# cd ..
[root@gswp2 emulex]# cd ..

[root@gswp2 Drivers]# ls
emulex megaraid mellanox old_mellanox_for_uninstall_script

[root@gswp2 Drivers]# modinfo mlx4_en | grep version
version: 4.6-1.0.1
rhelversion: 7.6
srcversion: 7225D7F266BAD8B6A0161CC
vermagic: 3.10.0-957.el7.x86_64 SMP mod_unload modversions

[root@gswp2 Drivers]# cd old_mellanox_for_uninstall_script/
[root@gswp2 old_mellanox_for_uninstall_script]# ls
mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.chg mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.txt
mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.tgz mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.xml

[root@gswp2 old_mellanox_for_uninstall_script]# tar zxfv mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.tgz
[root@gswp2 old_mellanox_for_uninstall_script]# ll
total 183980
drwxr-xr-x 2 1078 101 4096 May 2 2019 disks
-rw-r--r-- 1 1078 101 13995 May 2 2019 install.sh
-rw-r--r-- 1 1078 101 956 Jun 15 2017 License_gpl.txt
-rw-r--r-- 1 root root 575 Mar 10 18:16 mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.chg
-rw-r--r-- 1 root root 188315821 Mar 10 18:16 mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.tgz
-rw-r--r-- 1 root root 6860 Mar 10 18:16 mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.txt
-rw-r--r-- 1 root root 22802 Mar 10 18:16 mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.xml
-rw-r--r-- 1 1078 101 2116 May 2 2019 post-install.sh
-rw-r--r-- 1 1078 101 1239 May 2 2019 pre-install.sh
drwxr-xr-x 6 1078 101 4096 May 2 2019 RPMS
drwxr-xr-x 3 1078 101 4096 May 2 2019 SRPM
[root@gswp2 old_mellanox_for_uninstall_script]#
[root@gswp2 old_mellanox_for_uninstall_script]# cd ../mellanox/
[root@gswp2 mellanox]# ls
mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.chg mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.txt
mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.tgz mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.xml

[root@gswp2 mellanox]# tar zxfv mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.tgz
disks/
disks/mlnx-en-5.1-0.6.6.0.rhel7.6.x86_64.iso
disks/mlnx-en-5.1-0.6.6.0.rhel7.7.x86_64.iso
disks/mlnx-en-5.1-0.6.6.0.rhel7.8.x86_64.iso
install.sh
License_gpl.txt
RPMS/
RPMS/redhat-release-server-7.6/
RPMS/redhat-release-server-7.6/mlnx-en-utils-5.1-0.6.6.0.gc72091b.rhel7u6.x86_64.rpm
RPMS/redhat-release-server-7.6/kmod-mlnx-en-5.1-0.6.6.0.gc72091b.rhel7u6.x86_64.rpm
RPMS/redhat-release-server-7.7/
RPMS/redhat-release-server-7.7/kmod-mlnx-en-5.1-0.6.6.0.gc72091b.rhel7u7.x86_64.rpm
RPMS/redhat-release-server-7.7/mlnx-en-utils-5.1-0.6.6.0.gc72091b.rhel7u7.x86_64.rpm
RPMS/redhat-release-server-7.8/
RPMS/redhat-release-server-7.8/kmod-mlnx-en-5.1-0.6.6.0.gc72091b.rhel7u8.x86_64.rpm
RPMS/redhat-release-server-7.8/mlnx-en-utils-5.1-0.6.6.0.gc72091b.rhel7u8.x86_64.rpm
SRPM/
SRPM/redhat-release-server-7/
SRPM/redhat-release-server-7/mlnx-en-5.1-0.6.6.0.gc72091b.src.rpm

[root@gswp2 mellanox]# ls
disks License_gpl.txt mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.tgz mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.xml SRPM
install.sh mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.chg mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.txt RPMS
```

```
[root@gswp2 mellanox]#  
Here, install.sh will throw error, noting to worry about, follow next step  
[root@gswp2 mellanox]# cd ./old_mellanox_for_uninstall_script/  
[root@gswp2 old_mellanox_for_uninstall_script]# ls  
disks  License_gpl.txt  
install.sh  mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.tgz  mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.xml  pre-install.sh  SRPM  
mnlx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.chg  mlnx-Invgy_dd_nic_ib-4.6-1.0.1.1.1_sles12_x86-64.txt  post-install.sh  RPMS  
[root@gswp2 old_mellanox_for_uninstall_script]# ./pre-install.sh  
This will take long time, keep your session active.....  
[root@gswp2 old_mellanox_for_uninstall_script]# cd ..//mellanox/  
[root@gswp2 mellanox]# ls  
disks  License_gpl.txt  
install.sh  mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.tgz  mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.xml  SRPM  
mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.chg  mlnx-Invgy_dd_nic_cx.eth-5.1-0.6.6.0-0_rhel7_x86-64.txt  RPMS  
[root@gswp2 mellanox]# ./install.sh  
Reboot here-----  
[root@gswp2 Drivers]# modinfo mlx5_core | grep version  
[root@gswp2 Drivers]# modinfo lpfc | grep version  
[root@gswp2 Drivers]# modinfo megaraid_sas | grep version  
[root@gswp2 Drivers]# modinfo mlx5_core | grep version  
  
[root@gswp2 ~]# cd ~/LMS/Updates/Q1-2021/Updates/Firmware/HDD/  
[root@gswp2 HDD]# ll  
total 352436  
-rwxr-xr-x 1 root root 360556914 Mar 10 18:17 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin  
-rw-r--r-- 1 root root 105125 Mar 10 18:16 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.chg  
-rw-r--r-- 1 root root 11665 Mar 10 18:16 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.txt  
-rw-r--r-- 1 root root 216550 Mar 10 18:16 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.xml  
[root@gswp2 HDD]#  
  
[root@gswp2 HDD]# ./Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin -s  
[root@gsrhdp1 HDD]# ./Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin -s  
Running in 64,6 mode  
  
(c)Copyright Lenovo 2020.  
FdrvWL -- FLASH v:20.1.5[Mon Sep 21 07:39:06 2020]  
-----  
1 A: 0 S: 8 PN:01GV756:01GV765:01GV762 SNo:47XB01EB FW:CH47 size:400GB Flash: Latest  
2 A: 9 PN:01GV756:01GV765:01GV762 SNo:47XB01A5 FW:CH47 size:400GB Flash: Latest  
3 A: 0 S: 10 PN:00YK163:01GR790-01GR787 SNo:C2XKB02D FW:CF47 size:3840GB Flash: Latest  
4 A: 0 S: 11 PN:00YK163:01GR790-01GR787 SNo:C2XKB0EP FW:CF47 size:3840GB Flash: Latest  
5 A: 0 S: 12 PN:00YK163:01GR790-01GR787 SNo:C2XKB0D6 FW:CF47 size:3840GB Flash: Latest  
6 A: 0 S: 13 PN:00YK163:01GR790-01GR787 SNo:C2XKB02E FW:CF47 size:3840GB Flash: Latest  
  
[root@gswp2 HDD]# ./Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin -s ---> This step is for validation  
[root@gswp2 Drivers]# modinfo mlx5_core | grep version  
[root@gswp2 Drivers]# modinfo lpfc | grep version  
[root@gswp2 Drivers]# modinfo megaraid_sas | grep version  
=====Validation completed-----  
[root@gsrhdp1 Utilities]# modinfo megaraid_sas | grep version  
version: 07.712.02.00  
rhelversion: 7.6  
srcversion: C8328C2E3BFC8F1D2E39CC6  
vermagic: 3.10.0-957.el7.x86_64 SMP mod_unload modversions  
[root@gsrhdp1 Utilities]# modinfo lpfc | grep version  
version: 0:12.6.221.21  
rhelversion: 7.6  
srcversion: C14CE8DD4FFE3B0C2F5D39E  
vermagic: 3.10.0-957.el7.x86_64 SMP mod_unload modversions  
[root@gsrhdp1 Utilities]# modinfo mlx5_core | grep version  
version: 5.1-0.6.6  
rhelversion: 7.6  
srcversion: 819AC6CE04DA250FD1137C0  
vermagic: 3.10.0-957.el7.x86_64 SMP mod_unload modversions
```

```
[root@gsrhdp1 Utilities]# 
[root@gsrhdp1 Utilities]# rpm -qa | grep -i storcli
storcli-007.1510.0000.0000-1.noarch
[root@gsrhdp1 Utilities]# rpm -qa | grep -i lenovo-saphana-support
lenovo-saphana-support-202102221431-dcf08d77.noarch
[root@gsrhdp1 Utilities]#
[root@gsrhdp1 Utilities]# cd .../Updates/Firmware/HDD/
[root@gsrhdp1 HDD]# ll
total 352440
-rwxr----- 1 root root 360556914 Apr  9 02:23 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin
-rw-r----- 1 root root 105125 Apr  9 02:22 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.chg
-rw-r----- 1 root root 11665 Apr  9 02:22 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.txt
-rw-r----- 1 root root 216550 Apr  9 02:22 Invgy_fw_drives_all-1.35.02-0_linux_x86-64.xml
[root@gsrhdp1 HDD]# ./Invgy_fw_drives_all-1.35.02-0_linux_x86-64.bin -s
Running in 64,6 mode
```

(c)Copyright Lenovo 2020.

```
FdrvWL -- FLASH          v:20.1.5[Mon Sep 21 07:39:06 2020]
-----
1 A: 0 S: 8 PN:01GV756:01GV765-01GV762 SNo:47XB01EB FW:CH47 size:400GB Flash: Latest
2 A: 0 S: 9 PN:01GV756:01GV765-01GV762 SNo:47XB01A5 FW:CH47 size:400GB Flash: Latest
3 A: 0 S: 10 PN:00YK163:01GR790-01GR787 SNo:C2X BK02D FW:CF47 size:3840GB Flash: Latest
4 A: 0 S: 11 PN:00YK163:01GR790-01GR787 SNo:C2X BK0E FP:CF47 size:3840GB Flash: Latest
5 A: 0 S: 12 PN:00YK163:01GR790-01GR787 SNo:C2X BK0D6 FW:CF47 size:3840GB Flash: Latest
6 A: 0 S: 13 PN:00YK163:01GR790-01GR787 SNo:C2X BK02E FW:CF47 size:3840GB Flash: Latest
```

```
Ada(Type)  DevFnd Flashed(SAS SATA NVME) Failed
0(MR )   6      0 0 0 0 0
```

-----  
6 0 0 0 0 0 Error(s)

Update completed successfully.

```
[root@gsrhdp1 HDD]#
```

# RPM

Friday, 3 February 2023 3:05 PM

Rpm -qPR - To check depend

# Kernel upgrade to 5.17

Tuesday, 26 April, 2022 5:45 PM

## Kernel Update

```
103 cat /etc/os-release
104 uname -r
105 rpm -qa kernel-*
106 rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
107 rpm -Uvh http://www.elrepo.org/elrepo-release-7.0-3.el7.elrepo.noarch.rpm
108 yum --disablerepo="*" --enablerepo="elrepo-kernel" list available
109 yum --enablerepo=elrepo-kernel install kernel-ml
110 vim /etc/grub2.cfg
111 vim /etc/default/grub
112 rpm -qa kernel*
113 uname -msr
114 vim /etc/default/grub
115 grub2-mkconfig -o /boot/grub2/grub.cfg
116 uname -msr
```

```
yum -y install https://www.elrepo.org/elrepo-release-7.el7.elrepo.noarch.rpm
sudo rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
yum --disablerepo="*" --enablerepo="elrepo-kernel" list available | grep kernel-ml
yum --disablerepo="*" --enablerepo="elrepo-kernel" list available | grep kernel-lt
```

```
yum --enablerepo=elrepo-kernel install kernel-ml
```

GRUB configuration file i.e /etc/default/grub.

Using the text editor of your choice and setting GRUB\_DEFAULT=0

```
# grub2-set-default 0
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

```
uname -sr
```

## New server procedures:

1. yum update -y
2. yum install bash-completion vim nano screen -y
3. install required package
4. Change timezone to UTC (because default is CST) - timedatectl set-timezone UTC
5. check logrotation.d
6. check security group for port policy

# Roll Back the Kernel Version

Friday, 3 February 2023 5:33 PM

```
[root@jenkins ~]# grub2-editenv list  
saved_entry=CentOS Linux (3.10.0-1160.66.1.el7.x86_64) 7 (Core)
```

1. **# grub2-set-default "CentOS Linux (3.10.0-327.el7.x86\_64) 7 (Core)"  
# grub2-editenv list**

2. After the verification is complete, restart the OS from the default kernel.

```
CentOS Linux (3.10.0-862.3.2.el7.x86_64) 7 (Core)  
CentOS Linux (3.10.0-327.el7.x86_64) 7 (Core)  
CentOS Linux (0-rescue-2b86009638bb45c9ad2f4e3d14ba820a) 7 (Core)
```

3. Run the **uname -a** command to check whether the kernel version is restored.

# Downgrade kernel

Friday, 3 February 2023 5:52 PM

downgrades within the same major version (such as from **RHEL/CentOS 7.6 to 7.5**) but not between major versions (such as from **RHEL/CentOS 7.0 to 6.9**).

## 1. Check Kernel Version

```
# yum list kernel-3.10.0-862*
```

```
# yum install kernel-3.10.0-862.el7
```

## 2. Reboot System

```
# reboot
```

## 3. Downgrade RHEL/CentOS

```
# yum downgrade redhat-release
```

## 4. Confirm Downgrade

```
# cat /etc/redhat-release
```

# Kernel in Rescue

Friday, 3 February 2023 5:58 PM

## Booting into rescue mode

Boot the system using installation DVD or ISO and enter into the rescue mode. Follow the steps below for a detailed instruction on how to boot into rescue mode.

### 1. Attach the ISO image

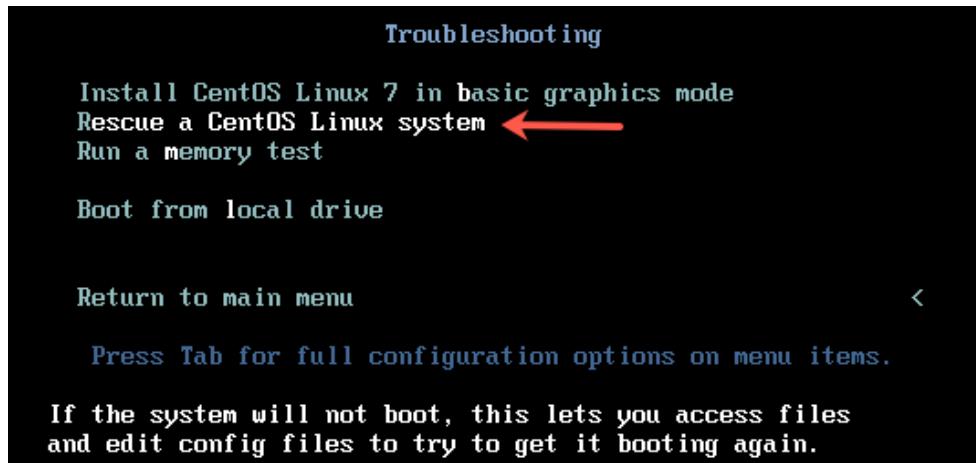
You can use an actual installation DVD instead of ISO image, but I find using ISO image easy and there is no need to go to the data center to physically insert the DVD into the system. Different virtualization platforms have similar features to attach/mount the ISO image to a VM guest. Make sure you change the boot order to boot from the ISO image.

### 2. Boot up the system

Boot up the CentOS 7 system from ISO image. At the boot screen, Select the **Troubleshooting option** at the end of the screen.



3. At the next screen, select the option **Rescue a CentOS Linux system**.



4. On the next screen, press enter to continue. When asked if you would like Rescue to find your installation, choose Continue.

```
Starting installer, one moment...
anaconda 21.48.22.93-1 for CentOS Linux 7 started.
* installation log files are stored in /tmp during the installation
* shell is available on TTY2
* if the graphical installation interface fails to start, try again with the
  inst.text bootoption to start text installation
* when reporting a bug add logs from /tmp as separate text/plain attachments
=====
Rescue
```

The rescue environment will now attempt to find your Linux installation and mount it under the directory : /mnt/sysimage. You can then make any changes required to your system. Choose '1' to proceed with this step. You can choose to mount your file systems read-only instead of read-write by choosing '2'. If for some reason this process does not work choose '3' to skip directly to a shell.

1) Continue ←

2) Read-only mount

3) Skip to shell

4) Quit (Reboot)

Please make a selection from the above: 1\_ ←

If you run into trouble detecting your install, retry using the Skip option and manually detect and mount your storage. You would get a message shown in the picture below if the rescue mode has detected the correct installation.

```
=====
Rescue Mount

Your system has been mounted under /mnt/sysimage.

If you would like to make your system the root environment, run the command:

    chroot /mnt/sysimage
Please press <return> to get a shell.
[anaconda] 1:main* 2:shell 3:log 4:storage-log 5:program-log      Switch tab: Alt+Tab | Help: F1
```

Now we skip the chroot step here as we do not want to enter the root environment.

## Installing the kernel

1. Next is to install the kernel appropriate to your installed system. It is important to have same installation media as that of the installed system version. Install the kernel using rpm command on the root environment /mnt/sysimage.

```
# cd /mnt/install/repo/Packages
# rpm -ivh --root=/mnt/sysimage kernel-3.10.0-514.el7.x86_64
```

```

sh-4.2# rpm -ivh --root=/mnt/sysimage kernel-3.10.0-514.el7.x86_64.rpm
Preparing... ################################################ [100%]
      package kernel-3.10.0-514.el7.x86_64 is already installed
sh-4.2#
[Anaconda] 1:main* 2:shell 3:log 4:storage-log 5:program-log      Switch tab: Alt+Tab | Help: F1

```

**2. Generate Grub2 configuration** – Next step is to change your root directory to /mnt/sysimage using the chroot command. This makes your system the root environment. Here you can generate the grub2 configuration for the newly installed kernel.

```

# chroot /mnt/sysimage
# grub2-mkconfig -o /boot/grub2/grub.cfg

```

**3. Verify** – Check for the file in /boot to have the new kernel. Also verify the kernel menuentry in the file **/boot/grub2/grub.cfg**.

```

ls -lrt /boot/vmlinuz-
-rwxr-xr-x. 1 root root 5392080 Nov 22 2016 /boot/vmlinuz-3.10.0-514.el7.x86_64
-rwxr-xr-x. 1 root root 5392080 Oct 1 12:44 /boot/vmlinuz-0-
rescue-4bd23218ddab41e587bdd39ae2fcf09a
# cat /boot/grub2/grub.cfg
.....
menuentry 'CentOS Linux (3.10.0-514.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu --
class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-514.el7.x86_64-
advanced-7efe94a2-10ec-40e4-8d89-a52faf13535e' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' f88bd588-6f4d-4050-bd3f-443cf2049ee7
    else
        search --no-floppy --fs-uuid --set=root f88bd588-6f4d-4050-bd3f-443cf2049ee7
    fi
    linux16 /vmlinuz-3.10.0-514.el7.x86_64 root=/dev/mapper/cl-root ro crashkernel=auto
    rd.lvm.lv=cl/root rd.lvm.lv=cl/swap rhgb quiet LANG=en_US.UTF-8
    initrd16 /initramfs-3.10.0-514.el7.x86_64.img
}
.....

```

#### 4. Filesystems relabeling

Create the file /.autorelabel to relabel the filesystems, during the next reboot, in case you are using SELinux:

```

# touch /.autorelabel
You can now exit the chroot environment and reboot the system.

```

# Recovery

Friday, 3 February 2023 1:02 PM

# InitRam - Initial ramDisk image build

Thursday, June 25, 2020 6:03 PM

Find out your kernel version:

```
# uname -r  
2.6.15.4
```

Make backup of existing ram disk:

```
# cp /boot/initrd.$(uname -r).img /root
```

To create initial ramdisk image type following command as the root user:

```
# mkinitrd -o /boot/initrd.$(uname -r).img $(uname -r)  
# ls -l /boot/initrd.$(uname -r).img
```

You may need to modify grub.conf to point out to correct ramdisk image, make sure following line existing in grub.conf file:

```
initrd /boot/initrd.img-2.6.15.4.img
```

When the system boots using an initrd image created by mkinitrd command, the linuxrc will wait for an amount of time which is configured through mkinitrd.conf, during which it may be interrupted by pressing ENTER. After that, the modules specified in will be loaded.

From <<https://www.cyberciti.biz/faq/rebuild-the-initial-ramdisk-image/>>

## Introduction to Grub Rescue

Grub Rescue: This guide is all about how to recover from a boot issue. One of the most facing issues could be the “grub rescue” prompt. Most of the questions in a few days came or how can I recover from grub rescue error. This may be due to human error or due to some patching activity as well. Let's go through a few steps on how to resolve the grub rescue boot issue.

## Grub configuration files

BIOS-based grub config

```
# /boot/grub2/grub.cfg
```

UEFI-based grub config

```
# /boot/efi/EFI/redhat/grub.cfg
```

If you need to configure any kernel parameter we should not touch the above configuration files, instead, edit the below file.

```
# vim /etc/default/grub
```

The available grub parameters are well documented in kernel official [admin-guide](#).

## Accidentally deleted files from /boot

In case, if you have accidentally deleted all files from /boot partition this guide may help to recover from it. For demonstration purpose, we have intentionally deleted all the files from /boot mount point.

```
[root@prodsvr1 ~]# df -hP /boot/
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1     1014M   33M  982M   4% /boot
[root@prodsvr1 ~]#
[root@prodsvr1 ~]# ls -lthr /boot/
total 0
[root@prodsvr1 ~]#
```

/boot mount point without files

## Type of Grub errors

While the reboot you may get the below grub prompts. Let's see why we getting different grub prompts.

The below one with only “**grub>**” because grub.cfg file missing or corrupted.

```
Minimal BASH-like line editing is supported. For the first word,
TAB lists possible command completions. Anywhere else TAB lists
possible device or file completions.
```

[www.linuxsysadmins.com](http://www.linuxsysadmins.com)

```
grub>
```

```
grub>
```

This could be due to all the files related to grub are deleted, /boot/grub2/ is empty or even this directory may be missing.

```
Booting from local disk...
.
error: file '/grub2/i386-pc/normal.mod' not found.
Entering rescue mode...          www.linuxsysadmins.com
grub rescue>
```

grub Rescue prompt

Any type of grub errors can be fixed easily by booting from the ISO/DVD rescue option.

## Starting the Recovery process

Boot from RHEL ISO or DVD. Select “Troubleshooting” option from the menu.



RHEL ISO boot screen

Choose Rescue mode by selecting “Rescue a Red Hat Enterprise Linux System” from the menu.



RHEL Rescue boot screen menu

This will load the anaconda installer with rescue mode. Once we get the options for rescue mode select 1 and press return key to enter into chroot environment.

```

Starting installer, one moment...
anaconda 21.48.22.93-1 for Red Hat Enterprise Linux 7.3 started.
* installation log files are stored in /tmp during the installation
* shell is available on TTY2
* if the graphical installation interface fails to start, try again with the
  inst.text bootoption to start text installation
* when reporting a bug add logs from /tmp as separate text/plain attachments
=====
=====
Rescue www.linuxsysadmins.com
The rescue environment will now attempt to find your Linux installation and
mount it under the directory : /mnt/sysimage. You can then make any changes
required to your system. Choose '1' to proceed with this step.
You can choose to mount your file systems read-only instead of read-write by
choosing '2'.
If for some reason this process does not work choose '3' to skip directly to a
shell.

1) Continue
2) Read-only mount
3) Skip to shell
4) Quit (Reboot)

Please make a selection from the above: 1

```

Select 1 to proceed rescue mode

## Doing Chroot

By running the below command access the chroot environment.

```

# chroot /mnt/sysimage
=====
Rescue

The rescue environment will now attempt to find your Linux installation and
mount it under the directory : /mnt/sysimage. You can then make any changes
required to your system. Choose '1' to proceed with this step.
You can choose to mount your file systems read-only instead of read-write by
choosing '2'.
If for some reason this process does not work choose '3' to skip directly to a
shell.

1) Continue
2) Read-only mount www.linuxsysadmins.com
3) Skip to shell
4) Quit (Reboot)

Please make a selection from the above: 1
=====

Rescue Mount

Your system has been mounted under /mnt/sysimage.

If you would like to make your system the root environment, run the command:

    chroot /mnt/sysimage
Please press <return> to get a shell.
When finished, please exit from the shell and your system will reboot.
sh-4.2#
sh-4.2#
sh-4.2# chroot /mnt/sysimage/
bash-4.2#
bash-4.2#
[Anaconda] 1:main* 2:shell 3:log 4:storage-log 5:program-log      Switch tab:
run chroot to get root environment

```

## Find and locate the /boot partition

Now it's time to recover the grub. Before that, we need to find under which disk /boot mount resides. Run # lsblk and # blkid command to verify and confirm the available partition and disk layouts.

```
bash-4.2#  
bash-4.2#  
bash-4.2# ls -lthr /boot/  
total 0  
bash-4.2#  
bash-4.2# lsblk  
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
sda        8:0    0   20G  0 disk  
|__sda1     8:1    0    1G  0 part /boot  
|__sda2     8:2    0   19G  0 part  
|   ├─rhel-root 253:2    0   17G  0 lvm  /  
|   └─rhel-swap 253:3    0    2G  0 lvm  [SWAP]  
sr0       11:0    1  3.5G  0 rom  
loop0      7:0    0 299M  1 loop  
loop1      7:1    0    2G  1 loop  
└─live-rw  253:0    0    2G  0 dm  
  └─live-base 253:1    0    2G  1 dm  
loop2      7:2    0 512M  0 loop  
└─live-rw  253:0    0    2G  0 dm  
zram0     252:0    0 976.5M  0 disk [SWAP]  
bash-4.2#  
bash-4.2#
```

find the /boot residing disk

## Install Grub

While installing a grub we should not install on a partition, instead install the grub on the whole disk. In my case /dev/sda1 is the partition used for /boot so I need to install the grub on /dev/sda.

Install the grub under /dev/sda by running below command.

```
# grub2-install /dev/sda
```

Once it installed we will get only the grub related files under /boot. But still, we need to follow more steps to get the vmlinuz, initramfs for a successful boot.

```

bash-4.2#
bash-4.2#
bash-4.2# grub2-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
bash-4.2#
bash-4.2# ls -lthr /boot/          www.linuxsysadmin.com
total 0
drwxr-xr-x. 5 root root 63 Jan 18 20:40 grub2
bash-4.2#
bash-4.2# ls -lthr /boot/grub2/
total 20K
drwxr-xr-x. 2 root root 4.0K Jan 18 20:40 locale
drwxr-xr-x. 2 root root 25 Jan 18 20:40 fonts
-rw-r--r--. 1 root root 1.0K Jan 18 20:40 grubenv
drwxr-xr-x. 2 root root 8.0K Jan 18 20:40 i386-pc
bash-4.2#
bash-4.2# _
```

grub2-install to install the grub

## Reinstalling Kernel

Mount the [RHEL](#) Disk/ISO under any location and reinstall the kernel. By reinstalling we will get back all the files like vmlinuz and initramfs.

```
# mount /dev/sr0 /mnt
# cd /mnt/Packages/
# yum reinstall kernel
```

## Saving or Regenerating GRUB Configuration

Finally, run the “grub2-mkconfig” to write the changes to a file. By running this command it will regenerate the required grub configuration file “grub.cfg”. The location will be under /boot/grub2/grub.cfg.

```

bash-4.2#
bash-4.2# grub2-mkconfig -o /boot/grub2/grub.cfg          www.linuxsysadmins.com
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.10.0-514.e17.x86_64
Found initrd image: /boot/initramfs-3.10.0-514.e17.x86_64.img
Found linux image: /boot/vmlinuz-0-rescue-d851f9e3aabc4690b96ad4787b40789c
Found initrd image: /boot/initramfs-0-rescue-d851f9e3aabc4690b96ad4787b40789c.img
done
bash-4.2#
[anaconda 1:main* 2:shell 3:log 4:storage-log 5:program-log      Switch tab: ]
grub2-mkconfig to save to config
At last, just list the files to verify. Hope, we are good to move forward now.
```

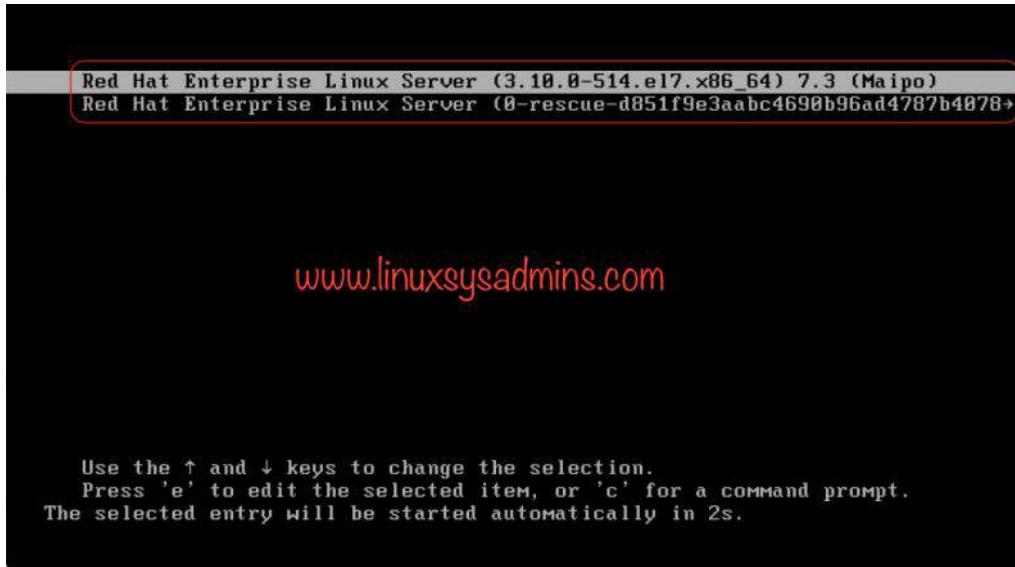
```

bash-4.2# ls -lthr /boot/grub2/
total 28K
drwxr-xr-x. 2 root root 4.0K Jan 18 20:40 locale
drwxr-xr-x. 2 root root 25 Jan 18 20:40 fonts
-rw-r--r--. 1 root root 1.0K Jan 18 20:40 grubenv
drwxr-xr-x. 2 root root 8.0K Jan 18 20:40 i386-pc
-rw-r--r--. 1 root root 4.2K Jan 18 20:53 grub.cfg
bash-4.2#
bash-4.2# ls -lthr /boot/
total 80M
-rw-----. 1 root root 3.0M Oct 19 2016 System.map-3.10.0-514.e17.x86_64
-rw-r--r--. 1 root root 135K Oct 19 2016 config-3.10.0-514.e17.x86_64
-rwrxr-xr-x. 1 root root 5.2M Oct 19 2016 vmlinuz-3.10.0-514.e17.x86_64
-rw-r--r--. 1 root root 272K Oct 19 2016 symvers-3.10.0-514.e17.x86_64.gz
-rw-----. 1 root root 20M Jan 18 20:58 initramfs-3.10.0-514.e17.x86_64.img
-rw-----. 1 root root 47M Jan 18 20:51 initramfs-0-rescue-d851f9e3aabc4690b96ad4787b40789c.img
-rwrxr-xr-x. 1 root root 5.2M Jan 18 20:51 vmlinuz-0-rescue-d851f9e3aabc4690b96ad4787b40789c
drwxr-xr-x. 5 root root 79 Jan 18 20:53 grub2
bash-4.2# _
```

All files are in the place

## Exiting Chroot/Rescue Shell

Type **exit** two times to exit from the chroot and rescue shell. Once you are exiting the rescue shell it will reboot the server. During the reboot, it will relabel SELinux for new files so avoid interrupting the process. Moreover, it will reboot automatically once again after the relabeling process.



Fixed Grub menu

After the reboot, we can see the grub menu and server boots normally.

## Recovering a UEFI based Machine

In UEFI based servers the configuration file location will be under /boot/efi/EFI/redhat/. In case, If we get "grub rescue" in UEFI based machine we can fix it with similar above steps. but the configuration location will be a little different from bios-based machines.

```
[root@server1 ~]# ls -l /boot/
total 99M
drwx----- 3 root root 4.0K Jan 1 1970 efi
-rw-r--r-- 1 root root 135K Oct 19 2016 config-3.10.0-514.el7.x86_64
-rw----- 1 root root 3.0M Oct 19 2016 System.map-3.10.0-514.el7.x86_64
-rw-r-xr-x 1 root root 5.2M Oct 19 2016 vmlinuz-3.10.0-514.el7.x86_64
-rw-r--r-- 1 root root 272K Oct 19 2016 symvers-3.10.0-514.el7.x86_64.gz
-rw-r--r-- 1 root root 599K Jan 18 05:09 initrd-plymouth.img
-rw----- 1 root root 47M Jan 18 05:13 initramfs-0-rescue-de38118e96024ea6a9193166a0a52701.img
-rw-rxr-xr-x 1 root root 5.2M Jan 18 05:13 vmlinuz-0-rescue-de38118e96024ea6a9193166a0a52701
-rw----- 1 root root 20M Jan 18 05:13 initramfs-3.10.0-514.el7.x86_64.img
-rw----- 1 root root 18M Jan 18 05:16 initramfs-3.10.0-514.el7.x86_64kdump.img
[root@server1 ~]# ls -l /boot/efi/EFI/redhat/
total 5.7M
-rwx----- 1 root root 1.3M Jul 20 2015 shim.efi
-rwx----- 1 root root 1.3M Jul 20 2015 shim-redhat.efi
-rwx----- 1 root root 1.3M Jul 20 2015 MokManager.efi
-rwx----- 1 root root 176 Jul 20 2015 BOOT.CSV
-rwx----- 1 root root 1001K Aug 29 2016 grubx64.efi
-rwx----- 1 root root 1001K Aug 29 2016 gcdx64.efi
drwx----- 2 root root 4.0K Jan 18 17:08 fonts
-rwx----- 1 root root 1.0K Jan 18 17:11 grubenv
-rwx----- 1 root root 4.2K Jan 18 17:11 grub.cfg
[root@server1 ~]#
```

Follow all the above steps except installing the packages and saving the grub configuration.

## Install the packages

In UEFI based machine we need to reinstall the below packages to fix the grub rescue prompt.

```
# yum reinstall grub2-efi kernel shim -y
```

```

Reinstall 3 Packages

Total size: 157 M
Installed size: 157 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : kernel-3.10.0-514.e17.x86_64 1/3
  Installing : 1:grub2-efi-2.02-0.44.e17.x86_64 2/3
  Installing : shim-0.9-2.e17.x86_64 3/3
  Verifying  : shim-0.9-2.e17.x86_64 1/3
  Verifying  : 1:grub2-efi-2.02-0.44.e17.x86_64 2/3
  Verifying  : kernel-3.10.0-514.e17.x86_64 3/3

Installed:
  grub2-efi.x86_64 1:2.02-0.44.e17          kernel.x86_64 0:3.10.0-514.e17          shim.x86_64 0:0.9-2.e17

Complete!
bash-4.2#
```

[www.linuxsysadmins.com](http://www.linuxsysadmins.com)

## Saving Grub configuration

Finally, save the configuration under below location.

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

To complete the rescue operation exit from chroot shell and rescue shell as mentioned in the earlier steps. Rebooting will take some time to complete the SELinux relabeling. Once done, it will be fine with booting the server.

# Disk

Monday, October 3, 2022 5:11 PM

# Ext2, Ext3 and Ext4

Wednesday, December 2, 2020 1:42 PM

## Ext2:

- ⌚ Ext2 stands for second extended file system.
- ⌚ It was to overcome limitation of legacy Ext file system
- ⌚ Journaling feature is not available.
- ⌚ It's being used for normally Flash based storage media like USB Flash drive, SD Card etc...

## Creating Ext2 File System

```
# mke2fs /dev/hdXX
```

## Ext3:

- ⌚ Ext3 stands for third extended file system.
- ⌚ Provide facility to upgrade from Ext2 to Ext3 file systems without having to back up and restore data.

## Creating Ext3 File System

```
# mke2fs -j /dev/hdXX OR  
# mkfs.ext3 /dev/hdXX  
-j option is used for journaling.
```

## Convert ext2 to ext3 :-

```
# umount /dev/sda2  
# tune2fs -j /dev/sda2  
# mount /dev/sda2 /var
```

## Ext4:

- ⌚ Ext4 stands for fourth extended file system.
- ⌚ Ext4 file system have **option to Turn Off journaling feature**.
- Other features like Sub Directory Scalability, Multiblock Allocation, Delayed Allocation, Fast FSCK etc.**
- ⌚ Ext4, the high-anticipated Ext3 successor

## Creating Ext4 File System

```
# mke2fs -t ext4 /dev/hdXX OR  
# mkfs.ext4 /dev/hdXX  
-t option to specify the file system type.
```

## Converting ext3 to ext4 :

```
( Warning :- Never try this live or production servers )  
# umount /dev/sda2  
# tune2fs -O extents,uninit_bg,dir_index /dev/sda2  
# e2fsck -pf /dev/sda2  
# mount /dev/sda2 /var
```

# XFS and btrfs

Wednesday, December 2, 2020 4:11 PM

## Managing the XFS and Btrfs File System

- XFS is default FS on RHEL 7
- Very versatile and flexible file system that allow you to work with big files and big file systems as well.
- It also uses b-tree database, info about files stored in very fast database.
- The new FS is coming in RHEL that is **btrfs**
- Ext4 old FS still there. It is based on h-tree index that's why it's slow

## Using XFS tools

Couple of Logical Volumes on RHEL7 System

```
[root@rhel7 ~]# lvs
  LV      VG     Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  root    centos -wi-ao---- <6.20g
  swap    centos -wi-ao---- 820.00m
  lvdisk1 vgdisk -wi-a----- 500.00m
  lvdisk2 vgdisk -wi-a----- 500.00m
  lvdisk3 vgdisk -wi-a----- 500.00m
  lvdisk4 vgdisk -wi-a----- 500.00m
  lvdisk5 vgdisk -wi-a----- 500.00m
  lvdisk6 vgdisk -wi-a----- 500.00m
```

Creating XFS file system on it.

```
[root@rhel7 ~]# mkfs.xfs /dev/vgdisk/lvdisk1
meta-data=/dev/vgdisk/lvdisk1      isize=512      agcount=4, agsize=32000 blks
                                  =                      sectsz=512  attr=2, projid32bit=1
                                  =                      crc=1      finobt=0, sparse=0
data      =                      bsize=4096   blocks=128000, imaxpct=25
          =                      sunit=0      swidth=0 blks
naming    =version 2            bsize=4096   ascii-ci=0 fttype=1
log       =internal log        bsize=4096   blocks=855, version=2
          =                      sectsz=512  sunit=0 blks, lazy-count=1
realtime  =none                extsz=4096   blocks=0, rtextents=0
```

Extend your LV and Grow XFS file system.

```
[root@rhel7 ~]# lvextend -L +200M /dev/vgdisk/lvdisk1
  Size of logical volume vgdisk/lvdisk1 changed from 500.00 MiB (125 extents) to 700.00 MiB (175 extents).
  Logical volume vgdisk/lvdisk1 successfully resized.
```

Make sure your LV is mounted before you grow your FS with XFS\_growfs

```

meta-data=/dev/mapper/vgdisk-lvdisk1 isize=512    agcount=4, agsize=32000 blks
          =                      sectsz=512  attr=2, projid32bit=1
          =                      crc=1    finobt=0 spinodes=0
data     =                      bsize=4096   blocks=128000, imaxpct=25
          =                      sunit=0    swidth=0 blks
naming   =version 2           bsize=4096   ascii-ci=0 ftype=1
log      =internal            bsize=4096   blocks=855, version=2
          =                      sectsz=512  sunit=0 blks, lazy-count=1
realtime =none                extsz=4096   blocks=0, rtextents=0
data blocks changed from 128000 to 179200

```

**xfs\_fsr** is a filesystem reorganizer, designed to be run regularly from a cron job to defragment XFS filesystems while they are mounted.

```
[root@rhel7 ~]# xfs_fsr /dev/vgdisk/lvdisk1
/mnt start inode=0
```

How to repair XFS. Make sure your file system is unmount.

```
[root@rhel7 ~]# xfs_repair /dev/vgdisk/lvdisk1
Phase 1 - find and verify superblock...
Phase 2 - using internal log
        - zero log...
        - scan filesystem freespace and inode maps...
        - found root inode chunk
Phase 3 - for each AG...
        - scan and clear agi unlinked lists...
        - process known inodes and perform inode discovery...
        - agno = 0
        - agno = 1
        - agno = 2
        - agno = 3
        - agno = 4
        - agno = 5
        - process newly discovered inodes...
        - agno = 0
        - agno = 1
        - agno = 2
        - agno = 3
        - agno = 4
        - agno = 5
        - process newly discovered inodes...
        - agno = 0
        - agno = 1
        - agno = 2
        - agno = 3
        - agno = 4
        - agno = 5
Phase 4 - check for duplicate blocks...
        - setting up duplicate extent list...
        - check for inodes claiming duplicate blocks...
        - agno = 0
        - agno = 1
        - agno = 2
        - agno = 3
        - agno = 4
        - agno = 5
Phase 5 - rebuild AG headers and trees...
        - reset superblock...
Phase 6 - check inode connectivity...
        - resetting contents of realtime bitmap and summary inodes
        - traversing filesystem ...
        - traversal finished ...
        - moving disconnected inodes to lost+found ...
Phase 7 - verify and correct link counts...
done
```

Sometime the XFS repair may not work because log corrupted then use **-L** option. This through away

log.

```
[root@rhel7 ~]# xfs_repair -L /dev/vgdisk/lvdisk1
Phase 1 - find and verify superblock...
Phase 2 - using internal log
    - zero log...
    - scan filesystem freespace and inode maps...
    - found root inode chunk
Phase 3 - for each AG...
    - scan and clear agi unlinked lists...
    - process known inodes and perform inode discovery...
    - agno = 0
    - agno = 1
    - agno = 2
    - agno = 3
    - agno = 4
    - agno = 5
    - process newly discovered inodes...
Phase 4 - check for duplicate blocks...
    - setting up duplicate extent list...
    - check for inodes claiming duplicate blocks...
    - agno = 0
    - agno = 1
    - agno = 2
    - agno = 3
    - agno = 4
    - agno = 5
Phase 5 - rebuild AG headers and trees...
    - reset superblock...
Phase 6 - check inode connectivity...
    - resetting contents of realtime bitmap and summary inodes
    - traversing filesystem ...
    - traversal finished ...
    - moving disconnected inodes to lost+found ...
Phase 7 - verify and correct link counts...
Maximum metadata LSN (1:31) is ahead of log (1:2).
Format log to cycle 4.
done
```

Dealing with XFS snapshots. You can take snapshot on mounted File system in XFS but not on ext4  
First, freeze your mounted FS so it will not be usable.

```
[root@rhel7 ~]# xfs_freeze -f /mnt/
[root@rhel7 ~]# xfs_freeze -u /mnt/
[root@rhel7 ~]#
```

## Btrfs Features

- Spanning multiple file systems
- Snapshots
- Checksums
- Copy on Write
- Online defragmentation
- Online file system growth and shrinking
- Online block device addition / removal
- Online balancing

## Understanding Subvolumes

- Subvolumes are not separate block device

## Creating btrfs file system on LV

```
[root@rhel7 ~]# mkfs.btrfs /dev/vgdisk/lvdisk3
btrfs-progs v4.9.1
See http://btrfs.wiki.kernel.org for more information.

Label:          (null)
UUID:           725ffddc-0ece-4d60-a08e-1b2180a384d7
Node size:      16384
Sector size:   4096
Filesystem size: 500.00MiB
Block group profiles:
  Data:          single     8.00MiB
  Metadata:      DUP       32.00MiB
  System:        DUP       8.00MiB
SSD detected:  no
Incompat features: extref, skinny-metadata
Number of devices: 1
Devices:
  ID      SIZE  PATH
    1    500.00MiB /dev/vgdisk/lvdisk3
```

Resize grow btrfs can be easily.

```
[root@rhel7 ~]# mount /dev/vgdisk/lvdisk3 /btrfs
[root@rhel7 ~]#
[root@rhel7 ~]# btrfs device add /dev/vgdisk/lvdisk4 /btrfs
[root@rhel7 ~]#
[root@rhel7 ~]# btrfs device stat /btrfs
[/dev/mapper/vgdisk-lvdisk3].write_io_errs 0
[/dev/mapper/vgdisk-lvdisk3].read_io_errs 0
[/dev/mapper/vgdisk-lvdisk3].flush_io_errs 0
[/dev/mapper/vgdisk-lvdisk3].corruption_errs 0
[/dev/mapper/vgdisk-lvdisk3].generation_errs 0
[/dev/mapper/vgdisk-lvdisk4].write_io_errs 0
[/dev/mapper/vgdisk-lvdisk4].read_io_errs 0
[/dev/mapper/vgdisk-lvdisk4].flush_io_errs 0
[/dev/mapper/vgdisk-lvdisk4].corruption_errs 0
[/dev/mapper/vgdisk-lvdisk4].generation_errs 0
[root@rhel7 ~]#
```

Best feature – you can spread btrfs file system over multiple devices.

```
[root@rhelserver ~]# mkfs.btrfs /dev/sdb1 /dev/sdb2
failed to open /dev/fd0: No such device or address
failed to open /dev/fd0: No such device or address

WARNING! - Btrfs v3.12 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using

Turning ON incompat feature 'extref': increased hardlink limit per file to 65536
adding device /dev/sdb2 id 2
fs created label (null) on /dev/sdb1
      nodesize 16384 leafsize 16384 sectorsize 4096 size 2.00GiB
Btrfs v3.12
[root@rhelserver ~]#
```

## Adding device into btrfs file system

```
[root@rhelserver ~]# btrfs device add /dev/sdb3 /data
[root@rhelserver ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/rhel-root  18G  2.9G   15G  17% /
devtmpfs        908M    0  908M   0% /dev
tmpfs          918M  148K  917M   1% /dev/shm
tmpfs          918M  8.9M  909M   1% /run
tmpfs          918M    0  918M   0% /sys/fs/cgroup
/dev/sdal       497M  119M  379M  24% /boot
/dev/sr0        3.5G  3.5G    0 100% /run/media/user/RHEL-7.0 Server.x86_64
/dev/sdbl       3.0G  384K  2.6G   1% /data
```

## Balance your data on btrfs

```
[root@rhel7 ~]# btrfs filesystem balance /btrfs
WARNING:

      Full balance without filters requested. This operation is very
      intense and takes potentially very long. It is recommended to
      use the balance filters to narrow down the balanced data.
      Use 'btrfs balance start --full-balance' option to skip this
      warning. The operation will start in 10 seconds.
      Use Ctrl-C to stop it.
10 9 8 7 6 5 4 3 2 1
Starting balance without any filters.
Done, had to relocate 3 out of 3 chunks
```

## Analyze on btrfs File System

```
[root@rhel7 ~]# btrfs device stats /btrfs
[/dev/mapper/vgdisk-lvdisk3].write_io_errs      0
[/dev/mapper/vgdisk-lvdisk3].read_io_errs       0
[/dev/mapper/vgdisk-lvdisk3].flush_io_errs      0
[/dev/mapper/vgdisk-lvdisk3].corruption_errs    0
[/dev/mapper/vgdisk-lvdisk3].generation_errs    0
[/dev/mapper/vgdisk-lvdisk4].write_io_errs      0
[/dev/mapper/vgdisk-lvdisk4].read_io_errs       0
[/dev/mapper/vgdisk-lvdisk4].flush_io_errs      0
[/dev/mapper/vgdisk-lvdisk4].corruption_errs    0
[/dev/mapper/vgdisk-lvdisk4].generation_errs    0
```

## Delete device from btrfs mount point

```
[root@rhelserver ~]# btrfs device delete /dev/sdb3 /data
[root@rhelserver ~]#
```

## Mounting btrfs FS on boot

```
# /etc/fstab
# Created by anaconda on Sat Jun 14 07:50:44 2014
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
/dev/mapper/rhel-root  /          xfs  defaults        1 1
UUID=d5f7227d-ea3b-4769-9c2d-a74a05ab3748 /boot      xfs  defaults        1 2
/dev/mapper/rhel-swap   swap       swap  defaults        0 0
/dev/sdb1      /data      btrfs  device=/dev/sdb1,device=/dev/sdb2  0 0
~
```

# Space not being freed from disk

Monday, October 3, 2022 4:56 PM

## Resolution

### Graceful shutdown of relevant process

First, obtain a list of deleted files which are still held open by applications:

```
$ lsof | egrep "deleted|COMMAND"
COMMAND      PID    TID TASKCMD     USER   FD  TYPE   DEVICE    SIZE/OFF      NODE NAME
ora        25575  8194 oracle    oracle   33    REG    65,65  4294983680  31014933 /oradata/DATAPRE/file.dbf (deleted)
```

Note: check either the filesystem path within NAME field or the device number under DEVICE to match the filesystem of interest.

The lsof output shows the process with pid 25575 has kept file /oradata/DATAPRE/file.dbf open with file descriptor (fd) number 33.

After a file has been identified, free the file used space by shutting down the affected process.

If a graceful shutdown does not work, then issue the kill command to forcefully stop it by referencing the PID.

### Root Cause

if the file is still open (in use by a running process) it will still be accessible to this process and will continue to occupy space on disk. Therefore such processes may need to be restarted before that file's space will be cleared up on the filesystem.

**Q5. / mount point usage is 99% however did not find any files to delete, how did you clear up the / usage and bring back to normal usage?**

» When lot of processes are running in Linux server there are lot of .TMP files will be created in background for support. You can't identify use of / file system in this case using lsof would help.

##To check deleted Temp Files

```
# lsof |grep deleted
```

##Clear Temp Files by killing PID's

```
# lsof |grep deleted |grep ".TMP" | awk '{ print $2 }' | xargs kill -9
```

##To delete Temp Files

```
# lsof |grep deleted |grep ".TMP" |awk '{print $9}' | xargs rm -f
```

# Scan new LUN

Monday, 7 November 2022 4:49 PM

Install sg3\_utils

```
# sg_scan -l
```

```
# echo "---" > /sys/class/scsi_host/host0/scan
```

```
# ls /sys/class/scsi_host/ | while read host; do echo "---" > /sys/class/scsi_host/$host/scan; done
```

"---" = wild cards for channel ID, SCI target ID and LUV

# LVM XFS

Sunday, April 4, 2021 10:35 PM

```
# pvscan  
# vgscan  
# lvscan  
  
147 pvcreate /dev/sdb  
148 vgcreate vgdata /dev/sdb  
149 lvcreate -L 100%FREE -n lvdata vgdata  
150 lvcreate -l 100%FREE -n lvdata vgdata  
  
151 mkfs.xfs /dev/vgdata/lvdata  
152 mkdir /data01  
153 mount /dev/vgdata/lvdata /data01
```

## Reduce LV

```
# umount <file system mount point>  
# e2fsck <device or partition name>  
# lvreduce -L - <Size of in MB></dev/vgname/lvname>  
# resize2fs </dev/vgname/lvname>  
# mount -a
```

## LV Extend

```
# lvextend -L 700M /dev/vg/test_lv  
# resize2fs /dev/vg/test_lv
```

Deactivate the LV

```
# lvchange -an /dev/vg_name/lv_name
```

Activate the LV

```
# lvchange -ay /dev/vg_name/lv_name
```

Disable the Volume Group

```
# vgchange -an volume_group_name
```

Enable Volume group

```
# vgchange -ay volume_group_name
```

# WWN

Sunday, 13 November 2022 2:46 PM

## Purpose of wwn number

To add storage to the host, server has to be mapped with storage device by zoning the WWN of both host and storage in Fabric switch.

Once the zone is created, the storage team can assign LUNs to a specific Linux host, and new LUN can be discovered by [scanning the storage LUN](#) ID at the host end.

- WWN – World Wide Name
- WWNN – World Wide Node Name
- WWPN – World Wide Port Name
- WWID – World Wide Identifier
- OUI – Organizationally Unique Identifier

## HBA card information

```
# lspci -nn | grep -i hba
```

## Checking wwn number

```
# ls -l /sys/class/fc_host
```

The ‘fc\_transport’ determines the correct host, channel, and target information from currently presented LUN:

```
# ls -lrt /sys/class/fc_transport/
```

list of ‘wwn’ number of the fc host

```
# cat /sys/class/fc_host/host?/port_name
```

specific fc host wwn number:

```
# cat /sys/class/fc_host/host1/node_name
```

status of HBA ports

```
# more /sys/class/fc_host/host?/port_state
```

## Method-2: Checking wwn number using systool command

The **systool** is a tool that uses APIs provided by **libsysfs** to gather information. It allows you to view system device information by bus, class, and topology.

When you run **systool** without parameters, it will present all available bus types, device classes, and root devices.

### How to install systool in Linux

**systool** can be easily installed from the distribution official repository.

For RHEL/CentOS 6/7 systems, use the [yum command](#) to install systool:

```
$ sudo yum install -y sysfsutils
```

For RHEL/CentOS 8 and Fedora systems, use the [dnf command](#) to install systool:

```
$ sudo dnf install -y sysfsutils
```

Once the **sysfsutils** package is installed on the Linux system, run the following command to find the WWN number of fc host:

```
# systool -c fc_host -v | grep port_name
```

```
port_name      = "0x500143802426baf2"  
port_name      = "0x500143802426baf3"  
port_name      = "0x500143802426baf4"  
port_name      = "0x500143802426baf5"
```

Run the following command to check the state of HBA ports:

```
# systool -c fc_host -v | grep port_state
```

```
port_state     = "Online"  
port_state     = "Online"  
port_state     = "Online"  
port_state     = "Online"
```

If you want to check wwn number of a specific fc host , run the following command:

```
# systool -c fc_host -v -d host2 | grep port_name
```

# Rescan ISCSI

Wednesday, December 2, 2020 10:38 PM

## Check the number of attached disks

You use the below commands to identify existing LUNs and how to add newly mapped LUNs to Linux.

```
# cat /proc/scsi/scsi | egrep -i 'Host:' | wc -l  
7
```

You can use the following command to have better output of all the disks

```
# fdisk -l 2>/dev/null | egrep '^Disk' | egrep -v 'dm-|type|identifier'  
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors  
Disk /dev/sdb: 16.1 GB, 16106127360 bytes, 31457280 sectors  
Disk /dev/sdc: 21.5 GB, 21474836480 bytes, 41943040 sectors  
Disk /dev/mapper/centos-root: 18.8 GB, 18756927488 bytes, 36634624 sectors  
Disk /dev/mapper/centos-swap: 2147 MB, 2147483648 bytes, 4194304 sectors  
Disk /dev/sdd: 10.7 GB, 10737418240 bytes, 20971520 sectors  
Disk /dev/sde: 10.7 GB, 10737418240 bytes, 20971520 sectors  
Disk /dev/sdf: 7516 MB, 7516192768 bytes, 14680064 sectors
```

or

```
# fdisk -l | grep sd  
Disk /dev/sda: 21.5 GB, 21474836480 bytes, 41943040 sectors  
/dev/sda1 * 2048 1026047 512000 83 Linux  
/dev/sda2 1026048 41943039 20458496 8e Linux LVM  
Disk /dev/sdb: 16.1 GB, 16106127360 bytes, 31457280 sectors  
Disk /dev/sdc: 10.7 GB, 10737418240 bytes, 20971520 sectors  
Disk /dev/sdd: 14.0 GB, 13958643712 bytes, 27262976 sectors
```

## 1) Using /sys class file

You can use the [echo command](#) to scan each scsi host device as below. Now to rescan the bus, use the following command

```
# echo "---" > /sys/class/scsi_host/host0/scan
```

The three dash ("---") of the command act as wildcards meaning rescan everything. Remember that the three values normally stand for channel, SCSI target ID, and LUN.

```
# echo "c t l" > /sys/class/scsi_host/hosth/scan
```

where

- *h* is the HBA number
- *c* is the channel on the HBA
- *t* is the SCSI target ID
- *l* is the LUN.

If you don't have the host bus number, you must list all the existing host bus number on your system with the command

```
# ls /sys/class/scsi_host  
host0 host1 host2
```

Then you will scan every iscsi disk found and scan after every scanning if the new disk was detected. It means

```
# echo "---" > /sys/class/scsi_host/host0/scan  
# echo "---" > /sys/class/scsi_host/host1/scan  
# echo "---" > /sys/class/scsi_host/host2/scan
```

or

```
# for host in `ls /sys/class/scsi_host/`; do
```

```
echo "--" >/sys/class/scsi_host/${host}/scan;
done
```

It may look like very simple as we perform this operation but the system has much work to do in the background when you execute storage scanning commands.

## Method to find Channel Routes

If we know the channel, target ID and LUN address, we can scan using that. Here, we have 4 HBA emulex cards 0, 1, 2 and 3.

```
server1:/proc/scsi/lpfc# ls
0 1 2 3
```

Through card 0 and 2,

```
server1:/proc/scsi/lpfc# cat 0
lpfc0t00 DID d200ef WWPN 50:06:01:68:39:a0:43:65 WWNN 50:06:01:60:b9:a0:43:65
lpfc0t01 DID d201ef WWPN 50:06:01:61:39:a0:43:65 WWNN 50:06:01:60:b9:a0:43:65
server1:#/proc/scsi/lpfc# cat 1
server1:/proc/scsi/lpfc# cat 2
lpfc2t01 DID ed0cef WWPN 50:06:01:69:39:a0:43:65 WWNN 50:06:01:60:b9:a0:43:65
lpfc2t00 DID ed0def WWPN 50:06:01:60:39:a0:43:65 WWNN 50:06:01:60:b9:a0:43:65
server1:/proc/scsi/lpfc# cat 3
```

It has the same WWNN (World Wide Node Name) for all the 4 WWPN (World Wide Port Name).

```
server1:/proc/scsi/lpfc# cat /sys/class/fc_transport/*/node_name
0x50060160b9a04365
0x50060160b9a04365
0x50060160b9a04365
0x50060160b9a04365
```

We can do a depth research by filtering the WWPN (World Wide Port Name) to have more information

```
server1:/proc/scsi/lpfc# grep 50060160b9a04365 /sys/class/fc_transport/*/*node_name
/sys/class/fc_transport/target0:0:0/node_name:0x50060160b9a04365
/sys/class/fc_transport/target0:0:1/node_name:0x50060160b9a04365
/sys/class/fc_transport/target2:0:0/node_name:0x50060160b9a04365
/sys/class/fc_transport/target2:0:1/node_name:0x50060160b9a04365
```

This indicates there are four Fibre Channel routes to this target.

```
Line 1 : Thru host 0 channel 0 target 0
Line 2 : Thru host 0 channel 0 target 1
Line 3 : Thru host 2 channel 0 target 0
Line 4 : thru host 2 channel 0 target 1
```

So now, you can scan for LUNs as follows and addresss "8" is given by storage team.

```
echo "0 0 8" > /sys/class/scsi_host/host0/scan
echo "0 1 8" > /sys/class/scsi_host/host0/scan
echo "0 0 8" > /sys/class/scsi_host/host2/scan
echo "0 1 8" > /sys/class/scsi_host/host2/scan
```

The output of SCSI file is illustrated for your reference:

```
Host: scsi0 Channel: 00 Id: 00 Lun: 08
Vendor: DGC Model: RAID 5 Rev: 0326
Type: Direct-Access ANSI SCSI revision: 04
Host: scsi0 Channel: 00 Id: 01 Lun: 08
Vendor: DGC Model: RAID 5 Rev: 0326
Type: Direct-Access ANSI SCSI revision: 04
Host: scsi2 Channel: 00 Id: 00 Lun: 08
Vendor: DGC Model: RAID 5 Rev: 0326
Type: Direct-Access ANSI SCSI revision: 04
Host: scsi2 Channel: 00 Id: 01 Lun: 08
Vendor: DGC Model: RAID 5 Rev: 0326
Type: Direct-Access ANSI SCSI revision: 04
```

## Another Method

If you don't have the host bus number, you must list all the existing host bus number on your system with the command

```
# ls /sys/class/scsi_host  
host0 host1 host2
```

or try

```
# grep mpt /sys/class/scsi_host/host?/proc_name  
/sys/class/scsi_host/host0/proc_name:mptspi
```

On the output, *host0* is the relevant field. As we have said earlier, we need to have the host bus number to determine what to scan

Then you will scan every iscsi disk found and scan after every scanning if the new disk was detected. It means

```
# echo "---" > /sys/class/scsi_host/host0/scan  
# echo "---" > /sys/class/scsi_host/host1/scan  
# echo "---" > /sys/class/scsi_host/host2/scan
```

If you have too many hosts (from host0 to host20 for example), you can use the command below

```
# for host in `ls /sys/class/scsi_host/`; do  
echo "---" >/sys/class/scsi_host/${host}/scan;  
done
```

or you can try (this one for Fibre Channel)

```
# for host in `ls /sys/class/fc_host/` ; do  
echo "1" >/sys/class/fc_host/${host}/issue_lip;  
done
```

Can devices be rescanned in Linux OS without reloading the Linux driver?

There is a procedure which forces the driver to rescan the targets and to allow a new device which is to be added. This triggers the driver to initiate a LUN discovery process.

To force a rescan from the command line, type the following command:

```
# echo "scsi-qlascan" > /proc/scsi//
```

where,

- = qla2100, qla2200, qla2300 (2.4 kernel drivers) or qla2xxx (2.6 kernel drivers)

- = the instance number of the HBA

After executing this command, force the SCSI mid layer to do its own scan and build the device table entry for the new device by typing the following command:

```
# echo "scsi add-single-device 0 1 2 3" >/proc/scsi/scsi
```

where,

- "0 1 2 3" = your "Host Channel ID LUN"

The scanning must be done in the above-mentioned order; first the driver (qla2300/qla2200 driver, etc.) and then the Linux SCSI mid-layer (i.e. OS scan).

## 2) Scan lun with multipath/powermt

You can check current multipath setup using multipath or powermt command.

```
# multipath -l  
mpath2 (36006016015501c0018c07c18e0d8dc11)  
[size=68 GB][features="1 queue_if_no_path"][hwhandler="1 emc"]
```

```

\_ round-robin 0 [active]
\ 0:0:0:2 sdc 8:32 [active]
\_ round-robin 0 [enabled]
\ 1:0:0:2 sdi 8:128 [active]

mpath1 (36006016015501c0084227c0ee0d8dc11)
[size=68 GB][features="1 queue_if_no_path"][hwandler="1 emc"]
\_ round-robin 0 [active]
\ 1:0:0:1 sdh 8:112 [active]
\_ round-robin 0 [enabled]
\ 0:0:0:1 sdb 8:16 [active]

```

.....  
.....

```

mpath3 (36006016015501c0019c07c18e0d8dc11)
[size=68 GB][features="1 queue_if_no_path"][hwandler="1 emc"]
\_ round-robin 0 [active]
\ 1:0:0:3 sdj 8:144 [active]
\_ round-robin 0 [enabled]
\ 0:0:0:3 sdd 8:48 [active]

```

If EMC powerpath is installed, check the disk/multipath status as below:

```

# powermt display dev=all | more
Pseudo name=emcpowerb
CLARiiON ID=APM00080503154 [cl10083]
Logical device ID=6006016023041F003AB6ED708885DD11 [LUN 3]
state=alive; policy=CLAROpt; priority=0; queued-IOs=0
Owner: default=SP A, current=SP A Array failover mode: 1
=====
----- Host ----- - Stor - -- I/O Path - -- Stats ---
### HW Path      I/O Paths  Interf. Mode State Q-IOs Errors
=====
3 qla2xxx      sde   SP A2  active alive  0   1
3 qla2xxx      sdh   SP B3  active alive  0   0
6 qla2xxx      sdk   SP A3  active alive  0   1
6 qla2xxx      sdn   SP B2  active alive  0   0
-----  
-----  
-----  
----- Host ----- - Stor - -- I/O Path - -- Stats ---
### HW Path      I/O Paths  Interf. Mode State Q-IOs Errors
=====
3 qla2xxx      sdd   SP A2  active alive  0   1
3 qla2xxx      sdg   SP B3  active alive  0   0
6 qla2xxx      sdj   SP A3  active alive  0   1
6 qla2xxx      sdm   SP B2  active alive  0   0

```

Multipath daemon will automatically add attached devices if it's configured properly. For powermt, we need to run below command manually.

```
# powermt config
```

### 3) Using Script

You can rescan using the SCSI rescan script which will detect new luns and add it to your server automatically. You can find this script in sg3\_utils package.

```
# /us/bin/rescan-scsi-bus -a
Scanning SCSI subsystem for new devices
Scanning host 0 for SCSI target IDs 0 1 2 3 4 5 6 7, all LUNs
.....
.....
1 new or changed device(s) found.
[0:0:5:0]
0 remapped or resized device(s) found.
0 device(s) removed.
```

## Conclusion

For storage (Netapp,equallogic) that use iscsi target, command `iscsiadm -m session --rescan` could be used to rescan when new lun added to the target. I would recommend always to use vendor-specific script or tools to scan Luns.

I hope you have enjoyed reading and let us know if you found any other method to scan storage disk devices.

From <<https://linoxide.com/storage/scandetect-luns-redhat-linux-outputs-remember/>>

# Multipath

Friday, 3 February 2023 1:22 PM

Multiple physical connections between a server and a storage array into one virtual device to aggregate storage bandwidth for improved performance.

Linux 7 supports multipathing using the **dm-multipath** subsystem. This uses the kernel device mapper system to generate virtual devices, managed by the multipathd daemon and the multipath command-line tool.

multipath devices are created under **/dev/mapper**.

**/etc/multipath.conf** configuration file

## Configuring multipathing

To configure multipathing, first make sure that the device-mapper-multipath package is installed.

```
# yum -y install device-mapper-multipath
```

Once the device-mapper-multipath package is installed, a configuration file must be created for the multipath daemon, **/etc/multipath.conf**. The easiest way to create this file is to use the **mpathconf** utility.

If there already is a file called **/etc/multipath.conf**, the mpathconf command will edit that file. If no such file exists, mpathconf will copy the default configuration from **/usr/share/doc/device-mapper-multipath-\*/multipath.conf**. If that file does not exist, mpathconf will create a new configuration file from scratch.

To create a default configuration, and then start and enable the multipathd daemon, use the following command:

```
# mpathconf --enable --with_multipathd y --with_chkconfig y
```

**Note:** In the default configuration file created by mpathconf, user-friendly names are enabled with the **user\_friendly\_names** option. User-friendly names will result in multipathed devices being named **mpathN**. While this can be useful if there is only one multipathed device, it can become confusing when there are multiple multipathed devices. To disable user-friendly names, use the **-user\_friendly\_names n** option to mpathconf. This will result in multipathed devices being named after their WWIDs.

If fine-tuning multipath configuration is desired before starting the multipathd daemon, use the mpathconf command with just the **--enable** option:

```
# mpathconf --enable
```

After editing the configuration file, enable and start the multipathd daemon as normal with the **systemctl** command.

## The multipath.conf configuration file

The multipath.conf configuration file consists of five sections:

multipath.conf section	Description
blacklist'{} blacklist_exceptions {}	This section defines which devices should be excluded from the multipath topology discovery.
defaults {} devices {}	This section defines the default settings to be used for all multipaths, unless explicitly overridden in the devices {} or multipaths {} section.
multipaths {}	This section contains overrides for the defaults {} section for specific types of devices, unless overridden from the multipaths {} section. Devices are identified based on their vendor, product, and revision keywords (regular expressions matching information from sysfs).

An easy way to remember overrides is: **multipaths > devices > defaults**.

## Blacklisting

Devices can be blacklisted in the configuration file using the **blacklist {}** section of multipath.conf. If blacklisting using wildcards, individual devices can be exempted from the blacklist using the **blacklist\_exceptions {}** section. Devices can be blacklisted using either their device node or their WWID. The following example shows an example of both:

```
blacklist {
```

```
devnode "^cciss"
wwid 1234567890abcdef
}
```

To determine the WWID of a disk device, use the `scsi_id` utility.

```
/usr/lib/udev/scsi_id -g -u /dev/sdN
360014053bd9ea2a35914e39a556051cf
```

## Defaults

Defaults for all multipaths can be set in the `defaults {}` section of `multipath.conf`. The complete list of all built-in defaults can be found in the file `/usr/share/doc/devicemapper-multipath-*/multipath.conf.defaults`. Some of the most interesting settings are:

- **path\_selector**: The algorithm that determines which path inside a priority group to use for the next I/O. The default of “**round-robin 0**” distributes I/O over all paths in the group. The number of request to be sent using one path before switching to the next is determined by the `rr_min_io_rq` setting. The alternatives are “**queue-length 0**”, which will send the next I/O request to the path with the shortest queue of outstanding requests, and “**service-time 0**”, which will send the next I/O request to the path that has the shortest estimated service time.
- **path\_grouping\_policy**: This setting defines how multiple paths are combined into priority groups. In the default of **failover**, every path will be put into a separate group. On the other hand, with the **multibus** policy, all possible paths are aggregated into a single group. Before configuring a multipath device to use the multibus policy, make sure that the storage controller supports active-active connections.
- **path\_checker**: This setting determines how the multipathd daemon will determine if a path is healthy. Other than the hardware independent options of **directio** and **readsector0**, there are a number of hardware independent checkers. Although the default for this option is directio, it is typically overridden in one of the default devices specified in the `devices {}` section.
- **features**: This option specifies the multipath features to enable. Syntax is the form of `num list`, where num represents the number of features being enabled and list represents list of features being enabled. The two available features are `queue_if_no_path` and `no_partitions`.
- **user\_friendly\_names**: This setting determines whether multipaths without a defined alias will be named `mpathN` (when set to yes), or if they will be named after their WWID.

**Note:** If the `queue_if_no_path` feature is enabled with the setting, features “`1 queue_if_no_path`”, and paths fail, processes issuing I/O will hang until the paths are restored. This behavior is undesirable in cluster implementations since one node stuck blocking on I/O to a failed storage device can block the rest of the cluster from accessing the storage resource. To avoid this situation, specify a value of `fail` for the `no-path_retry` parameter. Doing so will fail I/O immediately back up to higher layers rather than blocking on I/O indefinitely until paths are recovered.

**Note:** The commented defaults {} section in the `multipath.conf` produced by `mpathconf` does not reflect the actual built-in defaults of the multipath daemon.

## The `devices {}` section

In the `devices {}` section, the defaults for specific devices can be overridden. Inside the `devices {}` section, there are individual device {} subsections detailing settings for specific devices. Most common storage hardware already have their own section defined in the built-in defaults for the multipath daemon. If a hardware is not (yet) listed, a section for the hardware can be manually added. Following is an example definition for a nonexistent piece of storage hardware. The device itself is selected using a combination of **vendor**, **product**, and **revision**.

```
devices {
    device {
        vendor "MegaHyperSuperStorage"
        product "BAS"
        revision "513/B"
        features "1 queue_if_no_path"
        path_grouping_policy multibus
        path_checker tur
    }
}
```

In the preceding examples, the `features` line indicates that new I/O requests will be accepted and queued even when no paths are currently available.

## The `multipaths {}` section

In the `multipaths {}` section, overrides can be defined for specific multipaths. This can be used to set different `path_grouping` policies for a specific multipath. One of the other common uses of the `multipaths {}` section is to define an alias for a multipath. When an alias is set, the name for the device node in `/dev/mapper/` for this multipath will be based on the alias, making it easier to distinguish between different multipaths.

As an example, the following configuration will set an alias of `clusterstorage` for the multipath with a WWID of “`1234567890abcdef`”, as well as a `path_selector` of `queue-length`.

```
multipaths {  
    multipath {  
        wwid "1234567890abcdef"  
        alias "clusterstorage"  
        path_selector "queue-length 0"  
    }  
}
```

## Adding partitions

To add a partition on a multipathed device, use the following steps:

1. Create the partition on the multipathed device using a partition editor, e.g., **fdisk /dev/mapper/mpath0**.
2. Run the **partprobe** command to update the kernel's view of the partition table on all devices (including the devices collated into a multipath).
3. Run the command, **kpartx -a**, on the multipath device to create device mapper devices for the newly created partition(s).

## Removing a multipath

After removing all paths for a multipath, remove the multipathed device by running the command **multipath -f [device]**. If the multipathd daemon has been stopped and there are still device nodes for multipathed devices, flush all multipathed devices by running **multipath -F**. This can be useful when testing out different configurations and see remnants of old configurations lingering around.

Filed Under: [CentOS/RHEL 6](#), [CentOS/RHEL 7](#), [Fedora](#), [Linux](#)

# Moving VG to another system

Wednesday, December 2, 2020 1:40 PM

## 13.6. Moving a volume group to another system

It is quite easy to move a whole volume group to another system if, for example, a user department acquires a new server. To do this we use the `vgexport` and `vgimport` commands. `vgexport/vgimport` is not necessary to move drives from one system to another. It is an administrative policy tool to prevent access to volumes in the time it takes to move them.

### 13.6.1. Unmount the file system

First, make sure that no users are accessing files on the active volume, then unmount it

```
# umount /mnt/design/users
```

### 13.6.2. Mark the volume group inactive

Marking the volume group inactive removes it from the kernel and prevents any further activity on it.

```
# vgchange -an design  
vgchange -- volume group "design" successfully deactivated
```

### 13.6.3. Export the volume group

It is now necessary to export the volume group. This prevents it from being accessed on the ``old'' host system and prepares it to be removed.

```
# vgexport design  
vgexport -- volume group "design" successfully exported
```

When the machine is next shut down, the disk can be unplugged and then connected to its new machine

### 13.6.4. Import the volume group

When plugged into the new system it becomes `/dev/sdb` so an initial `pvscan` shows:

```
# pvscan  
pvscan -- reading all physical volumes (this may take a while...)  
pvscan -- inactive PV "/dev/sdb1" is in EXPORTED VG "design" [996 MB / 996 MB free]  
Moving a volume group to another system http://tldp.org/HOWTO/LVM-HOWTO/recipemovevgtonewsys.html  
1 of 2 12/22/2011 1:36 PM
```

```
pvscan -- inactive PV "/dev/sdb2" is in EXPORTED VG "design" [996 MB / 244 MB free]
pvscan -- total: 2 [1.95 GB] / in use: 2 [1.95 GB] / in no VG: 0 [0]
```

We can now import the volume group (which also activates it) and mount the file system.

If you are importing on an LVM 2 system, run:

```
# vgimport design
Volume group "vg" successfully imported
```

If you are importing on an LVM 1 system, add the PVs that need to be imported:

```
# vgimport design /dev/sdb1 /dev/sdb2
vgimport -- doing automatic backup of volume group "design"
vgimport -- volume group "design" successfully imported and activated
```

## 13.6.5. Activate the volume group

You must activate the volume group before you can access it.

```
# vgchange -ay design
```

## 13.6.6. Mount the file system

```
# mkdir -p /mnt/design/users
# mount /dev/design/users /mnt/design/users
```

The file system is now available for use.

[Prev](#) [Home](#) [Next](#)

Removing an Old Disk [Up](#) Splitting a volume group

Moving a volume group to another system <http://tldp.org/HOWTO/LVM-HOWTO/recipemovevgtonewsys.html>

2 of 2 12/22/2011 1:36 PM

# FS Security

Wednesday, December 2, 2020 3:03 PM

## Lesson 2 Managing file system security properties

---

### \* Encrypting your disk.

Someone can access your hard disk by connecting it to there system without any promoting username password. that's why you encrypt the disk.

In order to mount and access encrypt device you need passphrase.

/dev/sdb --> luksformat (going to create encryption layer) --> luksopen (Will create new device) --> /dev/mapper/secret (manage by device mapper) --> mapper (mkfs.ext4) --> mount (/dev/mapper/secret on somewhere

### Raw device

/dev/sdb1

LuksFromat on the device and it's going to create encryption layer

### Device Mapper

LuksOpen it will bring you on different layer. Where you going to work with encrypted device. so this will create new device. this new device managed by mapper.

### Device mapper going to map

Let's call it : /dev/mapper/secret

mkfs.ext4 now make FS on this level

mount now you can mount /dev/mapper/secret on somewhere

fdisk /dev/sdb1

cat /proc/partitions

cryptsetup luksFormat /dev/sdb1

cryptsetup luksOpen /dev/sdb1 secret

cat /proc/partitions

cd /dev/mapper/

ls

ls -ltr

mkfs.ext4 /dev/mapper/secret

mkdir /secret

mount /dev/mapper/secret /secret

mount

df -hP

umount /secret

cryptsetup luksClose /dev/mapper/secret

---

### \* Creating encrypted volumes

In order to mount the device we need luks Key. Otherwise you have to put passphrase to mount the device. LuksOpen need to be automated

/etc/crypttab (need to create crypt config file)

mount (2nd part is to automated is mount - /etc/fstab)

Create key file first

# touch urandom

# dd if=urandom of=/root/lukskey bs=4096 count=1

```
# cryptsetup luksAddKey /dev/sdb1 /root/lukskey
# cat /etc/crypttab
secret /dev/sdb1    /root/lukskey
# tail /etc/fstab
/dev/mapper/secret  /secret      ext4 defaults 1 2
```

---

#### \* Using security related mount options.

nodev (no device can be accessible from this filesystem)  
nosuid (no set user ID programs, dangerous permission)  
noexec (cannot execute from specific directory)

```
# nano /etc/fstab
# mount -o remount,noexec /dev/mapper/secret
# cd /secret/
# nano script.sh
# chmod +x script.sh
# sh script.sh
```

---

#### \* Monitoring file system changes AIDE

It's part of OS and it's static part. don't try it on /home or FS that containing large number of files.

```
# yum search aide
# yum install aide -y
Configuration file is under /etc/aide.conf
# aide --init (execute monitoring aide)
# aide --init
AIDE, version 0.15.1
### AIDE database at /var/lib/aide/aide.db.new.gz initialized.
# ls /var/lib/aide/
aide.db.new.gz
# ls -ltr /var/lib/aide/
total 1844
-rw----- 1 root root 1884552 Jul 24 12:49 aide.db.new.gz
# cd /var/lib/aide/
# mv aide.db.new.gz aide.db.gz (rename it)
# aide --check (checking the changes on system)
```

EXAMPLE:

```
# useradd linda (added new user)
# passwd linda
# aide --check
AIDE 0.15.1 found differences between database and filesystem!!
Start timestamp: 2018-07-24 12:53:31
```

Summary:

```
Total number of files: 51227
Added files: 0
Removed files: 0
Changed files: 4
```

---

Changed files:

---

```
changed: /etc/group
changed: /etc/gshadow
changed: /etc/passwd
```

changed: /etc/shadow

---

Detailed information about changes:

---

File: /etc/group

SHA256 : GodOUErZlIHkP1yiGyMkPbwO+6ghcuGl , X3r4kO/QsEEcR013cwyU+wiXrp930DHe

File: /etc/gshadow

SHA256 : B1ZStbRy5ElpY+5Q2l9wuLP/B6zq7u/l , zaYFC7LKFhVX3Tk9J6KL8KljLpHWdEkO

File: /etc/passwd

SHA256 : kw1rFlpG1J/mPdV91HQYvFic7V+WxK08 , UShdvgc2IFaj0B3o/xHfq4OpA+peFqq2

File: /etc/shadow

SHA256 : 90UzJAeleMk52Lw/4xPF9WGSn+2mcDDw , +NZTehiQ7mzUQ203sxnnmmaFU9Y0iTEO

---

# RAID

Wednesday, December 2, 2020 1:45 PM

RAID (redundant array of independent disks; originally redundant array of inexpensive disks) is a **way of storing the same data in different places on multiple hard disks to protect data in the case of a drive failure**. However, not all RAID levels provide redundancy.

## RAID controller

A RAID controller can be used as a level of abstraction between the OS and the physical disks, presenting groups of disks as logical units. Using a RAID controller can improve performance and help protect data in case of a crash.

## Standard RAID levels

### RAID 0:

This configuration has striping, but no redundancy of data.

It offers the best performance, but no fault tolerance.

### RAID 1:

Also known as *disk mirroring*,

this configuration consists of at least two drives that duplicate the storage of data.

There is no striping.

Read performance is improved since either disk can be read at the same time.

Write performance is the same as for single disk storage.

### RAID 5:

**This level is based on** block-level striping with parity.

The parity information is striped across each drive, allowing the array to function even if one drive were to fail.

The array's architecture allows read and write operations to span multiple drives.

This results in performance that is usually better than that of a single drive, but not as high as that of a RAID 0 array.

RAID 5 requires at least three disks, but it is often recommended to use at least five disks for performance reasons.

RAID 5 arrays are generally considered to be a poor choice for use on write-intensive systems because of the performance impact associated with writing parity information. When a disk does fail, it can take a long time to rebuild a RAID 5 array. Performance is usually degraded during the rebuild time, and the array is vulnerable to an additional disk failure until the rebuild is complete.

**RAID 6:** This technique is similar to RAID 5, but includes a second parity scheme that is distributed across the drives in the array. The use of additional parity allows the array to continue to function even if two disks fail simultaneously. However, this extra protection comes at a cost. RAID 6 arrays have a higher cost per gigabyte (GB) and often have slower write performance than RAID 5 arrays.

## Nested RAID levels

Some RAID levels are referred to as *nested RAID* because they are based on a combination of RAID levels. Here are some examples of nested RAID levels.

### RAID 10 (RAID 1+0):

Combining RAID 1 and RAID 0

This level is often referred to as RAID 10

which offers higher performance than RAID 1,

but at a much higher cost. In RAID 1+0, the data is mirrored and the mirrors are striped.

**RAID 01 (RAID 0+1):** RAID 0+1 is similar to RAID 1+0, except the data organization method is slightly different. Rather than creating a mirror and then striping the mirror, RAID 0+1 creates a stripe set and then mirrors the stripe set.

**RAID 03 (RAID 0+3, also known as RAID 53 or RAID 5+3):** This level uses striping (in RAID 0 style) for RAID 3's virtual disk blocks. This offers higher performance than RAID 3, but at a much higher cost.

**RAID 50 (RAID 5+0):** This configuration combines RAID 5 distributed parity with RAID 0 striping to improve RAID 5 performance without reducing data protection.

# Ext2,3,4 & XFS

Sunday, 13 November 2022 1:47 PM

## 14. What are differences between the ext2, ext3, ext4 and xfs file systems?

S.No.	Ext2	Ext3	Ext4
1.	<i>Stands for Second Extended file system.</i>	<i>Stands for Third Extended file system.</i>	<i>Stands for Fourth Extended file system.</i>
2.	<i>Does not have Journaling feature.</i>	<i>Supports Journaling feature.</i>	<i>Supports Journaling feature.</i>
3.	<i>Max. file size can be from 16 GB to 2 TB.</i>	<i>Max. file size can be from 16 GB to 2 TB.</i>	<i>Max. file size can be from 16 GB to 16 TB.</i>
4.	<i>Max. file system size can be from 2 TB to 32 TB</i>	<i>Max. file system size can be from 2 TB to 32 TB</i>	<i>Max. file system size can be from 2 TB to 1 EB *1EB = 1024 Peta bytes.]</i>

- XFS is default FS on RHEL 7
- Very versatile and flexible file system that allow you to work with big files and big file systems as well.
- It also uses b-tree database, info about files stored in very fast database.

## Certified and [maximum] individual file size

File system	RHEL 3	RHEL 4	RHEL 5	RHEL 6	RHEL 7	RHEL 8
<b>Ext2/3</b>	1TiB (3.0) 2TiB (3.5+)	2TiB	2TiB	2TiB	2TiB	2TiB
<b>Ext4</b>	n/a	n/a	16TiB (5.6+) <sup>2</sup>	16TiB	16TiB	16TiB
<b>GFS1</b>	2TiB	16TiB [8EiB]	16TiB [8EiB]	n/a	n/a	n/a
<b>GFS2<sub>1</sub></b>	n/a	n/a	100TiB (5.3+) [8EiB]	100TiB [8EiB]	100TiB [8EiB]	100TiB [8EiB]
<b>XFS<sub>3</sub></b>	n/a	n/a	100TiB [8EiB]	100TiB [8EiB]	500TiB [8EiB]	8EiB

## Certified and [maximum] file system size

File system	RHEL 3	RHEL 4	RHEL 5	RHEL 6	RHEL 7	RHEL 8
<b>Ext2/3</b>	1TiB (3.0) 2TiB (3.5+) [8TiB]	8TiB	8TiB (5.0), 16TiB (5.1+) <sup>4</sup>	16TiB	16TiB	16TiB
<b>Ext4</b>	n/a	n/a	16TiB [1EiB] (5.6+) <sup>2</sup>	16TiB [1EiB]	50TiB [1EiB]	50TiB [1EiB]
<b>GFS</b>	2TiB	16TiB [8EiB]	16TiB [8EiB]	n/a	n/a	n/a
<b>GFS2<sub>1</sub></b>	n/a	n/a	100TiB (5.3+) [8EiB]	100TiB [8EiB]	100TiB [8EiB]	100TiB [8EiB]
<b>XFS<sub>3</sub></b>	n/a	n/a	100TiB [16EiB]	300TiB [16EiB] <sup>5</sup>	500TiB [16EiB]	1PiB

# Swap Extend

Sunday, 5 February 2023 11:32 PM

## Commands

```
# dd
    # dd if=/dev/zero of=/newswap bs=1M count=1024
    # chmod go-r or 600 newswap
# mkswap /newswap
# swapon /newswap
# swapoff
```

=====

```
# lvextend -L 900M /dev/centos/swap
# swapon -s
# swapoff -v /dev/centos/swap
# mkswap /dev/centos/swap
# swapon -va
```

# Permission

Saturday, 12 November 2022 2:42 PM

The diagram illustrates the calculation of file permissions. At the top, 'u g o' (User, Group, Other) are shown above the octal value '754'. Below this, three binary columns represent the permissions for each level:

	User (u)	Group (g)	Other (o)
access	r w x	r w x	r w x
binary	4 2 1	4 2 1	4 2 1
enabled	1 1 1	1 0 1	1 0 0
result	4 2 1	4 0 1	4 0 0
total	7	5	4

The final row shows the total permissions: 7 (User), 5 (Group), and 4 (Other).

## Linux File Permissions

 blog.bytebytego.com

Binary	Octal	String Representation	Permissions
000	0 (0+0+0)	---	No Permission
001	1 (0+0+1)	--x	Execute
010	2 (0+2+0)	-w-	Write
011	3 (0+2+1)	-wx	Write + Execute
100	4 (4+0+0)	r--	Read
101	5 (4+0+1)	r-x	Read + Execute
110	6 (4+2+0)	rw-	Read + Write
111	7 (4+2+1)	rwx	Read + Write + Execute

The diagram shows the breakdown of file permissions into three levels: Owner, Group, and Other. Above the table, the permissions are represented as a string: 'r w x' for Owner, 'r w -' for Group, and 'r - x' for Other.

	Owner	Group	Other
<b>r</b>	Read	4	7
<b>w</b>	Write or Edit	2	6
<b>x</b>	Execute	1	5

OS

Thursday, 2 February 2023 9:22 PM

# RHEL 6, 7, 8

Monday, 14 November 2022 3:51 PM

## General Information

S.No	Description	RHEL 8	RHEL 7	RHEL 6
1	General Availability Date	14-Nov-18	10-Jun-14	10-Nov-10
2	Code Name	Ootpa	Maipo	Santiago
3	Kernel Version	4.18	3.10.0-123	2.6.32-71
4	End of Full Support	May-2024	Q4 2019	10-May-16
5	End of Maintenance Support 1	N/A	Q4 2020	10-May-17
6	End of Maintenance Support 2	May-2029	30-Jun-2024	30-Nov-20
7	End of Extended Life cycle Support	TBD	N/A	30-Jun-24
8	Last Minor Release	TBD	7.7	6.10

## Boot Process Related Changes

S.No	Description	RHEL 8	RHEL 7	RHEL 6
1	Boot Loader: The GRUB2 looks very similar to GRUB but there are many features added.	GRUB 2	GRUB 2	Legacy GRUB
2	Runlevel: Runlevels are referred as target but there is no difference but they merged runlevel 2,3,4 into one.	runlevel0.target -> poweroff.target runlevel1.target -> rescue.target runlevel2.target -> multi-user.target runlevel3.target -> multi-user.target runlevel4.target -> multi-user.target runlevel5.target -> graphical.target runlevel6.target -> reboot.target	runlevel0.target -> poweroff.target runlevel1.target -> rescue.target runlevel2.target -> multi-user.target runlevel3.target -> multi-user.target runlevel4.target -> multi-user.target runlevel5.target -> graphical.target runlevel6.target -> reboot.target	runlevel 0 runlevel 1 runlevel 2 runlevel 3 runlevel 4 runlevel 5 runlevel 6
3	To view runlevel/target	systemctl get-default	systemctl get-default	runlevel
4	To change runlevel/target	systemctl isolate [Name.target]	systemctl isolate [Name.target]	init [runlevel]
5	To configure default runlevel/target	systemctl set-default [Name.target]	systemctl set-default [Name.target]	/etc/inittab
6	To break root password or Boot into single user mode	Append rd.break or init=/bin/bash to kernel cmdline	Append rd.break or init=/bin/bash to kernel cmdline	Append 1 or s or init=/bin/bash to kernel cmdline

7	KDUMP	Kdump is enabled by default and will run without any problems if the system has too much RAM.	Kdump is enabled by default and will run without any problems if the system has too much RAM (up to 3 TB).	Kdump is enabled by default and will run without any problems if the system has too much RAM.
---	-------	---	--	---

### Major Package Changes

S.No	Description	RHEL 8	RHEL 7	RHEL 6
1	System Manager: Systemd is a new init system and system manager which was adapted by most of the major distribution.	systemd	systemd	upstart
2	Service Manager	systemctl command	systemctl command	service command
3	Enable Service on Boot	systemctl command	systemctl command	chkconfig
4	Network Time Synchronization	Only Chrony (faster time sync and useful for the systems which are not online all the time)	It supports Chrony and ntp	ntp
5	Network Bonding	Teamd	Teamd	Bonding
6	To view ports/sockets	ss and lsof	ss and lsof	netstat, ss and lsof
7	Cluster Resource Manager	Pacemaker	Pacemaker	Rgmanager
8	GUI Interface (Desktop)	Gnome 3.28	Gnome 3	Gnome 2
9	Default Display Server	Wayland	X.Org	X.Org
10	Default Database	MySQL 8.0, MariaDB 10.3, PostgreSQL 10 and 9.6, and Redis 5.0	MariaDB	MySQL
11	Default Firewall	Firewalld, it uses nftables framework in the backend	Firewalld, it uses Iptables framework in the backend	Iptables
12	Temporary Files Management	systemd-tmpfiles	systemd-tmpfiles	tmpwatch
13	Load Balancer Technology	Keepalived and HAProxy	Keepalived and HAProxy	Piranha
14	Python	Python 3	Python 2.7.5	Python 2.0
15	PHP	PHP 7.2	PHP 5.4	PHP 5.3
16	Compiler	GCC 8.2.1	GCC 4.8.2	GCC 4.4

### File System Related Changes

S.No	Description	RHEL 8	RHEL 7	RHEL 6
1	Default File System	XFS	XFS	EXT4
2	File System Check	xfs_repair	xfs_repair	e2fsck

3	File System Extend: xfs_growfs (This doesn't allow you to reduce a filesystem)	xfs_growfs	xfs_growfs	resize2fs
---	--	------------	------------	-----------

## Other Changes

S.No	Description	RHEL 8	RHEL 7	RHEL 6
1	First Process owned by	systemd (PID 1)	systemd (PID 1)	init (PID 1)
2	Network Interface Name	enpXXX (enp0s3)	enpXXX (enp0s3)	eth0
3	Host Name Change	It needs to be defined in /etc/hostname file	/etc/hostname	It's defined in /etc/sysconfig/network file
4	UID Allocation Change	0-999 UIDs are reserved for system and application users.	0-999	0-499
5	Max Supported (Individual) File & Filesystem Size	XFS= 500TB XFS= 1024TB	XFS= 500TB XFS= 500TB	EXT4= 16TB EXT4= 16TB
6	ISO Image	Only 64-Bit	Only 64-Bit	32-Bit and 64-Bit
7	Mount Options Change	By default user_xattr and acl mount options are enabled	By default user_xattr and acl mount options are enabled	Need to enable them manually
8	Default Repos	Repo ID: rhel-8-for-x86_64-appstream-rpms Repo Name: Red Hat Enterprise Linux 8 for x86_64 – AppStream (RPMS) Repo ID: rhel-8-for-x86_64-baseos-rpms Repo Name: Red Hat Enterprise Linux 8 for x86_64 – BaseOS (RPMS)	Repo ID: rhel-7-server-rpms Repo Name: Red Hat Enterprise Linux 7 Server (RPMS)	Repo ID: rhel-6-server-rpms Repo Name: Red Hat Enterprise Linux 6 Server (RPMS)
9	Package Management	By default both are installed, YUM symbolic link to DNF	By default only YUM and DNF can be installed from the Extra repo	Only YUM
10	Max. RAM Supported	24 TB on x86_64 architecture	12 TB on x86_64 architecture	12 TB on x86_64 architecture
11	Directories Change	Directories /bin, /sbin, /lib and /lib64 are now all under the /usr directory	All these under the /usr directory	All these under the / directory
12	Logging	rsyslog and journal	rsyslog and journal	Only rsyslog
13	Minimum required disk space	10GB minimum, 20GB recommended	10GB minimum, 20GB recommended	1GB minimum, 5GB recommended
14	Is the upgrade possible?	Yes	Yes, RHEL 7.7 to RHEL 8	Yes, RHEL 6.10 to RHEL 7.7
15	Virtual Machines Management	cockpit	virt-manager	virt-manager

# MEMORY PROCESSES

Friday, September 24, 2021 2:17 AM

## ### ZOMBIE PROCESS ###

Let's say you one running process on system called (P1) and it is completed the execution and has to give exit status to Parent Process (P0). so let's say the parents process (P0) is on sleep mode then the (P1) will move to process table that call Zombie process until it exist in process table.

## ### PROCESSES ###

The Processes is any program that is running on your system.

# ps aux - will give you the list of processes (aux)

## ### THREAD ###

A Thread is the unit of execution within a process. A process can have anywhere from just one thread to many threads. So when the program is running threads comes to life. Previously one process can have only one process but now one process can have multiple threads. so when process running there are certain unit of execution and each unit known as a thread. Program --> Process --> Thread (Process Explorer)

## ### MEMORY LEAK ###

Memory - RAM on your computer

Leak - Loss of Memory

Memory leak is not for permanent and your not going to lose your RAM.

First thing how you gonna find out this issue is Computer slowness, less response or entire system stuck.

The main reason is Program holding memory but not actually using it - MEMORY LEAK.

The proper way is when program required memory then system provides it. but after the job done program has to return the memory back to the system.

Check the process page and note which process ID is using more RAM.

Garbage Collection: this program will run it will look for memory that is not in use and release it back to the system.

# Root Password

Monday, July 12, 2021 2:34 PM

```
rd.break
# mount -o remount,rw /sysroot
# chroot /sysroot
# passwd
# touch /.autorelabel
# mount -o remount,ro /
```

# /etc/passwd - Understand

Thursday, February 18, 2021 12:13 PM

## Understanding /etc/passwd file fields

The /etc/passwd contains one entry per line for each user (user account) of the system. All fields are separated by a colon (:) symbol. Total of seven fields as follows. Generally, /etc/passwd file entry looks as follows:

The diagram shows a line of text: "oracle:x:1021:1020:Oracle user:/data/network/oracle:/bin/bash". Seven vertical arrows point downwards from the colon-separated fields to the numbers 1 through 7 respectively. The fields are: 1. Username (oracle), 2. Password (x), 3. User ID (UID) (1021), 4. Group ID (GID) (1020), 5. User ID Info (Oracle user), 6. Home directory (/data/network/oracle), and 7. Command/shell (/bin/bash).

(Fig.01: /etc/passwd file format – click to enlarge)

### /etc/passwd Format

From the above image:

1. **Username:** It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password:** An x character indicates that encrypted password is stored in /etc/shadow file. Please note that you need to use the passwd command to compute the hash of a password typed at the CLI or to store/update the hash of the password in /etc/shadow file.
3. **User ID (UID):** Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID):** The primary group ID (stored in /etc/group file)
5. **User ID Info:** The comment field. It allows you to add extra information about the users such as user's full name, phone number etc. This field is used by finger command.
6. **Home directory:** The absolute path to the directory the user will be in when they log in. If this directory does not exist then user's directory becomes /
7. **Command/shell:** The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell.

For example, sysadmin can use the nologin shell, which acts as a replacement shell for the user accounts. If shell set to `/sbin/nologin` and the user tries to log in to the Linux system directly, the `/sbin/nologin` shell closes the connection.

From <<https://www.cyberciti.biz/faq/understanding-etcpassword-file-format/>>

# SSH Password Less

Monday, August 9, 2021 3:31 PM

```
# ssh-keygen  
# ssh-copy-id -i .ssh/id_rsa.pub root@192.168.100.80  
# ssh 192.168.100.80
```

**Troubleshooting: SSH public key added but still prompt for password**

```
$ chmod 600 ~/.ssh/authorized_keys
```

## Set Hostname

Wednesday, July 14, 2021 10:08 PM

```
# hostnamectl set-hostname serverb.example.com
```

# umask

Monday, 7 November 2022 5:02 PM

## Q9. What is umask and how umask will be calculated?

When a new files/directories created in Linux which are applying a permissions "mask" called the umask.

```
Max allowed Permissions: 666  
umask value: 022  
File Permissions: 644
```

In case of directories:

```
Max allowed permissions: 777  
umask value: 022  
Directory permissions: 755
```

The user file-creation mode mask (umask) is used to determine the file permissions for newly created files or directories.

Linux assigns default permissions to a file or a directory at the time of creation.

Default permissions are calculated based on the umask (user mask) permission value subtracted from a preset value called "initial" permissions.

Default permission = umask - initial

Example:

Create a file as a regular user

```
#- touch file  
#- ll  
-rw-rw--r--. 1 root root 0 May 10 17:04 file  
6 6 4
```

readwrite-readwrite--read-- (default permissions)

- To find umask value:

```
#- umask  
002
```

For regular users:

- Initial permissions = 666 for files
- Initial permissions = 777 for directories

Files:

- Initial permissions - umask = Default permission

$$666 - 002 = 664$$

```
-rw-rw--r--. 1 root root 0 May 10 17:04 file  
6 6 4
```

Directories:

- Initial permissions - umask = Default permission  
777 - 002 = 775

Example:

```
#- mkdir 2 (as regular user)
#- ll 2
drwxrwxr-x. 2 1 1 6 May 10 17:20 2
7 7 5
```

# Logrotate

Monday, 4 April, 2022 5:41 PM

Logfile > (scrape) filebeat > (filter) logstash > (store) elasticsearch > (display) kibana

## Logrotation example

```
# cat /etc/cron.hours/logrotate
#!/bin/sh

/usr/sbin/logrotate -s /var/lib/logrotate/logrotate.status /etc/logrotate.conf
EXITVALUE=$?
if [ $EXITVALUE != 0 ]; then
    /usr/bin/logger -t logrotate "ALERT exited abnormally with [$EXITVALUE]"
fi
```

=====  
logrotate command

You can experiment with the log rotation (outside of the usual cron job) by forcing an execution of the **logrotate** in the absence of any log files to rotate. To do so, use the **-f** option and specify the configuration file you wish to use.

```
# logrotate -f /etc/logrotate.d/linuxserver
```

If you experience any problems, and wish to debug, you can use the **-d** option with **logrotate**. This will simulate a “test run” and not actually make any changes. Instead, it will only output debug messages to help with troubleshooting.

```
# logrotate -d /etc/logrotate.d/linuxserver
```

Use the **-v** option to turn on verbosity. This will display messages during rotation so you can see exactly what's going on.

```
# logrotate -v /etc/logrotate.d/linuxserver
```

- **weekly:** Rotate logs once per week. Available options are daily, weekly, monthly, and yearly
- **missingok:** It's OK if no \*.log files are found
- **rotate #:** Keep specified number of files before deleting older log files
- **compress:** Compress (gzip) log files
- **delaycompress:** Delays compression until second time around
- **compresscmd:** Set which command to used to compress. Defaults to gzip
- **uncompresscmd:** Set the command to use to uncompress. Defaults to gunzip
- **notifempty:** Don't rotate empty files
- **create 640 root adm:** Create new log files with set permissions/owner/group
- **postrotate:** Scripts to run after rotating is done
- **prerotate:** Scripts to run before log rotating begins
- **size:** Rotate when the file size reaches a particular limit

Install elastic stack  
<https://www.elastic.co/guide/en/beats/filebeat/current/setup-repositories.html>

# Cpu Mem

Monday, October 3, 2022 5:22 PM

```
# top
```

```
# free -m
```

```
# vmstat
```

# Load Average + Inode + Hard Link

Wednesday, December 2, 2020 2:55 PM

## 25) What are hard links?

Hard links point directly to the physical file on disk, and not on the path name. This means that if you rename or move the original file, the link will not break, since the link is for the file itself, not the path where the file is located.

**Flags** indicate that a certain property, characteristic, or feature is active or is present. Certain flags are only supported on particular filesystems and partitioning tables. The flags (on GPT) are actually 128-bit GUIDs on the GUID partitioning table. Each flag has its own GUID.

## SAN vs NAS

SAN is a dedicated network of storage devices(can include tape drives storages, raid disk arrays etc) all working together to provide an excellent block level storage. WhileNAS is a single device/server/computing appliance, sharing its own storage over the network.

## Q:30 What is load average in a linux?

Ans: Load Average is defined as the average sum of the number of process waiting in the run queue and number of process currently executing over the period of 1,5 and 15 minutes. Using the 'top' and 'uptime' command we find the load average of a linux sever.

## What is an inode?

An inode is a data structure on a filesystem on Linux and other Unix-like operating systems that stores all the information about a file except its name and its actual data. like location of file on the disk, access mode, ownership, file type etc. it's large numbers that used in index and linux kernel used inode numbers to access the content inside the file system metadata (times of last change,[2] access, modification)

Linux Permission

Read = 4

Write = 2

Execute = 1

User. Group. Everyone

# Directory Hierarchy

Saturday, 12 November 2022 11:37 AM

/

Contains all other files and dir.

**/bin**

Essential binary executables commands for use by all users. Cd, cp, ls, ping, ps.

**/boot**

To successfully start the computer during the boot process.

**/dev**

Hold device files, such as Physical devices attached to the computer such as hard drives, sound devices and communication ports.

**/etc**

Configurations files that control the operating of program. Also contains scripts used to startup and shutdown individual program.

**/etc/cron.d**

Contain scripts which are executed on regular schedule by the crontab process.

**/etc/rc.d**

Contain the files required to control system services and configure the mode of operation for computer.

**/home**

Each requires space store information specific to them that is done via home directory

**/lib**

Contains shared library that supports the executable files located under /bin and /sbin. Also holds the kernel modules (drivers) responsible for giving Linux a great deal of versatility to add or remove functionality as needs dictate.

**/lost+found**

Contain potentially recoverable data that might be produced if the file system undergoes an improper shut-down due to crash or power failure.

**/media**

For temporarily mount removable devices.

**/opt**

To install third party or additional optional software that is not part of the default installation.

**/proc**

Contains information about running process

**/var/tmp**

Temp store for data that needs to be held between reboots

**/var/spool**

Data stored for later processing. Printers which will queue print jobs in spool file for eventual printing and then deletion when the resource becomes available.

**/var/log**

Holds log files from a range of programs and services.

**/var/lib**

This directory holds dynamic state information that programs typically modify while they run.

**/var**

Contains variable data files. These are files that are expected to grow under normal circumstances.

**/sbin**

Similar to the /bin directory. Holds binary executables / commands

**/srv**

provide a location for storing data for specific services.

**/usr/bin**

where user programs and data are stored and shared. contains several subdirectories that mirror those in the root (/) directory to make organisation more consistent.

# Add Repo

Wednesday, July 14, 2021 10:06 PM

## Install Epel Repo

```
# # dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

## Add Repo

```
# dnf config-manager --add-repo http://repo.eight.example.com
```

## List Repo

```
# dnf repository-packages epel list
```

# Repo Local

Wednesday, December 2, 2020 11:03 PM

Creating a local repository and configuring the GUI

```
# mkdir /mnt/cdrom  
  
# mount /dev/sr0 /mnt/cdrom  
  
# mkdir /var/www/html/rhel6/Packages/  
  
# cp -r /mnt/cdrom/Packages/* /var/www/html/rhel6/Packages/
```

2. Copy other necessary subdirectories by entering the following commands:

```
# cp -r /mnt/cdrom/isolinux /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/images /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/Server/repo /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/HighAvailability /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/ScalableFilesystem /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/LoadBalancer /var/www/html/rhel6/  
  
# cp -r /mnt/cdrom/ResilientStorage /var/www/html/rhel6/
```

3. To create a repository description for yum, create a new file named local.repo inside /etc/yum.repos.d. Add the following contents to the new file:

```
# nano /etc/yum.repos.d/local.repo
```

```
[local]  
name=RHEL6 - local  
baseurl=file:///var/www/html/rhel6  
enabled=1  
gpgcheck=0
```

```
ctrl+O > Save Your file  
ENTER > ENTER  
ctrl+x > Exit
```

6. Enter the command:

```
#yum clean all
```

# History

Sunday, 13 November 2022 11:34 AM

```
# history | more  
# !4  
# !ps  
# history -c  
# export HISTTIMEFORMAT=`%F %T`  
# hist  
ory | more  
# alias h1='history 10'  
# alias h2='history 20'  
# alias h3='history 30'  
  
# vi ~/.bash_profile  
HISTSIZE=450  
HISTFILESIZE=450  
  
# export HISTCONTROL=ignoredups  
# export HISTCONTROL=erasedups
```

# Set Timezone

Sabtu, 25 Februari 2023 7:24 PTG

```
# timedatectl list-timezones  
# timedatectl set-timezone "Asia/Kuala_Lumpur"  
# timedatectl set-ntp yes
```

```
# dnf install chrony  
# systemctl status chronyd  
  
# cat /etc/chrony.conf
```

```
# timedatectl  
Local time: Wed 2021-07-14 22:52:30 +08  
Universal time: Wed 2021-07-14 14:52:30 UTC  
RTC time: Wed 2021-07-14 14:52:29  
Time zone: Asia/Kuala_Lumpur (+08, +0800)  
System clock synchronized: yes  
NTP service: active  
RTC in local TZ: no
```

```
# timedatectl set-time [YYYY-MM-DD]  
# timedatectl set-time [HH:MM:SS]  
# timedatectl set-local-rtc yes  
# timedatectl set-local-rtc no  
  
date +%r
```

You can make an alias for date in .bashrc by typing alias date=date +%r and then doing source .bashrc, if you want to change date default format.

# User

Sunday, 13 November 2022 12:47 PM

## The types of users in Linux and their attributes:

Type of User	Example	User ID	Group ID	Home Directory	Default Shell
Super User	Root	0	0	/root	/bin/bash
Normal User	ram, raju, gopal, ...etc.,	500 - 60000	500 - 60000	/home/<user name>	/bin/bash
System User	ftp, ssh, apache, nobody, ...etc.,	1 - 499	1 - 499	/vat/ftp, ...etc	/sbin/nologin
Network User	Remote user like LDAP user	Same as normal users	Same as normal users	/home/guests/ldap user	/bin/bash
Sudo User	Normal users with admin privileges	Same as normal users	Same as normal users	/home/<user name>	/bin/bash

System user - Admin who controlled the system

Normal user - Created by admin user

System user - created when app or soft installed

- ⌚ User names and user id are stored in /etc/passwd file.
- ⌚ User's passwords are stored in /etc/shadow file in an encrypted form.
- ⌚ Users are assigned a home directory and a shell to work with the O/S.
- ⌚ And some user environmental files like .bash\_logout, .bash\_profile, .bashrc , ...etc., are also copied from /etc/skell to his/her home directory (/home/<username>).

.bash\_logout : is a user's logout ending program file. It will execute first whenever the user is logout.  
.bash\_profile : is user's login startup program file. It will execute first whenever the user is login. It consists the user's environmental variables.

.bashrc : This file is used to create the user's custom commands and to specify the umask values for that user's only.

## /etc/passwd

<user name> : x : <uid> : <gid> : <comment> : <user's home directory> : <login shell> (where 'x' means link to password file ie., /etc/shadow file)

- ⌚ /etc/passwd ----> Stores user's information like user name, uid, home directory and shell ...etc.,
- ⌚ /etc/shadow ----> Stores user's password in encrypted form and other information.
- ⌚ /etc/group -----> Stores group's information like group name, gid and other information.
- ⌚ /etc/gshadow ---> Stores group's password in encrypted form.
- ⌚ /etc/passwd---> Stores the /etc/passwd file backup copy.
- ⌚ /etc/shadow---> Stores the /etc/shadow file backup copy.
- ⌚ /etc/default/useradd ----> Whenever the user created user's default settings taken from this file.
- ⌚ /etc/login.defs ----> user's login defaults settings information taken from this file.
- ⌚ /etc/skell -----> Stores user's all environmental variables files and these are copied from this directory to user's home directory.

## # usermod <options><user name>

- \* The options are, -L ----> lock the password
- U ----> unlock the password
- o ----> creates duplicate user modify the user's id same as other user
- u ----> modify user id
- g ----> modify group id
- G ----> modify or add the secondary group
- c ----> modify comment
- d ----> modify home directory
- s ----> modify user's login shell
- l ----> modify user's login name
- md ----> modify the users home directory and the old home directory

### **Assign the password to user**

```
# useradd <user name> (to create the user)
# passwd -S <user name> (to see the status of the password of that user. if root user is not assigned the password then the password status is locked)
# passwd -d <user name> (then delete the password for that user)
# chage -d 0 <user name> (it will change the password age policy)
# su - <user name> (Try to switch to that user then it will display the following message)
Newpassword : (type new password for that user)
Retype password : (retype the password again)
```

### **Restrict users**

- (i) By removing (deleting) the user we can restrict the user from login.
- (ii) Put the user's hostnames as entries in /etc/hosts.deny file (applying TCP wrappers).
- (iii) #passwd -l <user name> (by locking his password we can restrict the users).

### **User profiles**

Profile is a file to enter some settings about users working environment

#### **Global profile :**

- (1) Only root user can set and applicable to all the users.
- (2) Only global parameters can entered in this profile.
- (3) The location of the global profile is /etc/bashrc

#### **Local profile :**

- (1) Every user has his/her own profile.
- (2) The settings entered in this profile are only for that user.
- (3) The location of the profile is .bash\_profile (hidden file) in that particular user's home directory.

# Adding user

Sunday, 13 November 2022 12:57 PM

## Normal user

```
# useradd -u <uid> -g <gid> -G <secondary group> -c <comment> -d <home directory> -s <shell><username>
Example : # useradd -u 600 -g 600 -G java -c "oracle user" -d /home/raju -s /bin/bash raju
```

## Check user

```
# id mali
uid=1001(mali) gid=1001(mali) groups=1001(mali)
```

## Create duplicate user

```
# useradd -o -u 0 -g root <user name>
```

## Recover user

```
# pwunconv (It creates the users according /etc/passwd file and deletes the /etc/shadow file)
```

## Find and Kill the user

```
# fuser -cu (to see who are login)
#fuser -ck <user login name> (to kill the specified user)
```

## User with no login access

```
# useradd -s /sbin/nologin sap
```

# PAM

Sunday, 13 November 2022 1:42 PM

pam\_tally.so module maintains a count of attempted accesses, can reset count on success, can deny access if too many attempts fail. Edit /etc/pam.d/system-auth file, enter:

```
auth required pam_tally.so no_magic_root
account required pam_tally.so deny=3 no_magic_root lock_time=180 Where,
deny=3 : Deny access if tally for this user exceeds 3 times.
```

# Profile

Friday, 3 February 2023 11:30 AM

(i) Profile is a file to enter some settings about users working environment. ie., we can set user home directory, login shell, path, ...etc.,  
Profiles are two types.

- (a) Global profile
- (b) Local profile

Global profile :

- (1) Only root user can set and applicable to all the users.
- (2) Only global parameters can entered in this profile.
- (3) The location of the global profile is /etc/bashrc

Local profile :

- (1) Every user has his/her own profile.
- (2) The settings entered in this profile are only for that user.
- (3) The location of the profile is .bash\_profile (hidden file) in that particular user's home directory.

# ACL

Sunday, 13 November 2022 1:37 PM

permissions to files and directories. Using Access Control list we assign the permissions to some particular users to access the files and directories.

Create a partition using # fdisk command.

Format the above partition with ext4 file system using # mkfs.ext4 <partition name> command.  
Create the mount point using # mkdir /<mount point> command.

Mount that file system on the mount point using # mount -o acl <partition name><mount point> command.

Mount the partition permanently using

# vim /etc/fstab (open this file and make an entry as below)  
<partition name><mount point><file system type> defaults, acl 0 0

Save and exit this file.

If the partition is already mounted then just add acl after defaults in /etc/fstab file and execute the below command

# mount -o remount <partition name>

## Check ACL

# getfacl <options><file or directory name>

The options are, -d -----> Display the default ACLs.

-R -----> Recurses into subdirectories.

## Assign ACL

# setfacl <options><argument> : <username>: <permissions><file or directory name>

The options are, -m -----> Modifies an ACL.

-x -----> Removes an ACL.

-b -----> Remove all the ACL permissions on that directory.

-R -----> Recurses into subdirectories.

The arguments are, u -----> user

g -----> group

o -----> other

## Assign r+w to user , group

# setfacl -m u : <user name> : <permissions><file or directory>

# setfacl -m g : <user name> : <permissions><file or directory>

# setfacl -m o : <user name> : <permissions><file or directory>

# setfacl -m u : <user name> : <permissions>, g : <user name> : <permissions>, o : <user name> : <permissions><file or directory>

Useful commands :

# setfacl -x u : <user name><file or directory name> (to remove the ACL permissions from the user)

# setfacl -x g : <user name><file or directory name> (to remove the ACL permissions from group)

# setfacl -x o : <user name><file or directory name> (to remove the ACL permissions from other)

# setfacl -b <file or directory> (to remove all the ACL permissions on that file directory)

# Processes with Signals

Sunday, 5 February 2023 11:57 PM

Controlling Processes with Signals:

A system may have hundreds or thousands of processes running simultaneously on it.

It is sometimes necessary to alert a process of an event.

We can accomplish that by send a control signal to the process.

Processes can send a signal to alert each other as well

A process will halt its execution as soon as it gets the signal and take appropriate action as per the enclosed instructions in that signal.

The signal can instruct the process to terminate gracefully, kill it abruptly or force it to re-read its configuration.

There are many signals available for use but we will mostly deal with only a few of them.

Each signal is associated with a unique numeric identifier, a name and an action.

- List of available signals can be displayed by using the `kill -l` command with `-l` option:

```
#- kill -l
```

Examples:

1- SIGHUP:

Hang up signal causes a process to disconnect itself from a closed terminal it was tied to.

Can also instruct a running daemon to re-read its configuration.

2- SIGINT:

The `^c` signal issued on the controlling terminal can

interrupt the execution of a process.

9- SIGKILL:

Kills a process abruptly.

15- SIGTERM:

Sends the signal to stop a process in an orderly way.  
This is the default signal and can be ignored by a process.

Commands:

The commands kill and pkill are used to send a signal to a process. Regular users can kill processes that they own and root user can kill any process.

- pkill:

The pkill command needs process name(s) to send a signal to.

Examples:

- To kill with default soft termination signal:

#- pkill top

- To kill forcefully right away use:

#- pkill -9 top

- kill

The kill command needs PID(s) to send a signal.

Examples:

- To kill with default soft termination signal:

#- kill top

- To kill forcefully right away use:

#- kill -9 top

- The command killall can be used to terminate all processes that match provided criteria.

Example:

#- killall top



# Sticky bit

Sunday, 13 November 2022 1:33 PM

It protects the data from other users when all the users having full permissions on one directory.

Example : # chmod o+t <directory name> (to set the sticky bit permission on that directory)

# ls -ld <directory name>

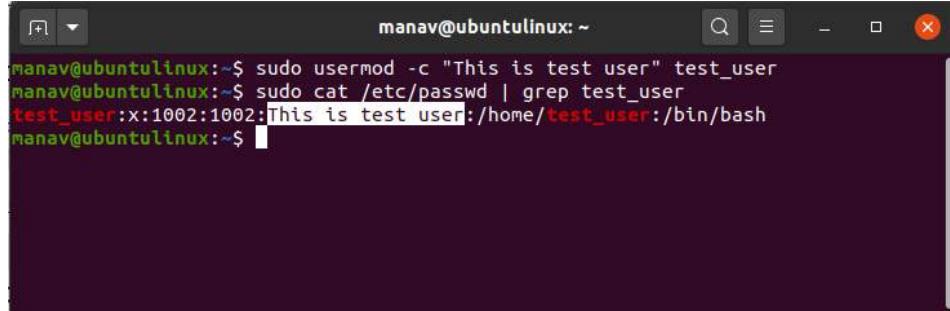
r w x r w x r w t <directory name> (where 't' is called the sticky bit)

# Usermod

Thursday, February 18, 2021 12:16 PM

## 1. To add a comment for a user

```
sudo usermod -c "This is test user" test_user
```



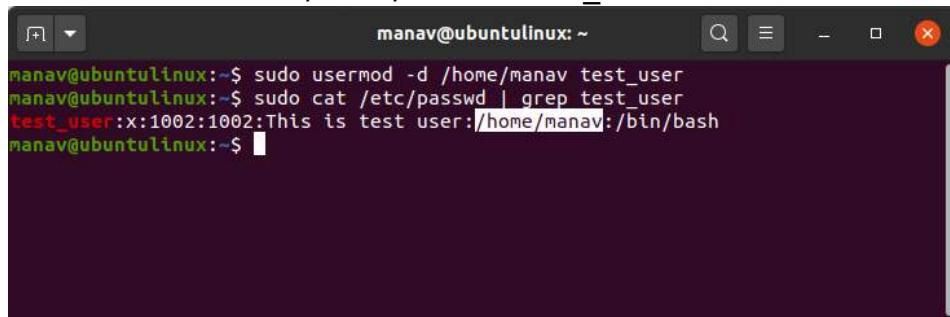
A terminal window titled "manav@ubuntulinux: ~". The command "sudo usermod -c \"This is test user\" test\_user" is run, followed by "sudo cat /etc/passwd | grep test\_user". The output shows the user entry with the new comment.

```
manav@ubuntulinux:~$ sudo usermod -c "This is test user" test_user
manav@ubuntulinux:~$ sudo cat /etc/passwd | grep test_user
test_user:x:1002:1002:This is test user:/home/test_user:/bin/bash
manav@ubuntulinux:~$
```

This will add a comment about the user or a short description related to the user.

## 2. To change the home directory of a user

```
sudo usermod -d /home/manav test_user
```



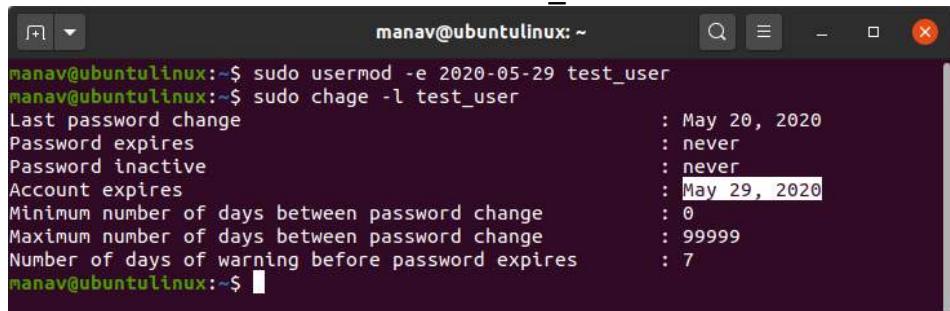
A terminal window titled "manav@ubuntulinux: ~". The command "sudo usermod -d /home/manav test\_user" is run, followed by "sudo cat /etc/passwd | grep test\_user". The output shows the user entry with the new home directory.

```
manav@ubuntulinux:~$ sudo usermod -d /home/manav test_user
manav@ubuntulinux:~$ sudo cat /etc/passwd | grep test_user
test_user:x:1002:1002:This is test user:/home/manav:/bin/bash
manav@ubuntulinux:~$
```

This will change the home directory of the user to /home/manav.

## 3. To change the expiry date of a user

```
sudo usermod -e 2020-05-29 test_user
```



A terminal window titled "manav@ubuntulinux: ~". The command "sudo usermod -e 2020-05-29 test\_user" is run, followed by "sudo chage -l test\_user". The output shows the account's password history, expiration, and warning period.

```
manav@ubuntulinux:~$ sudo usermod -e 2020-05-29 test_user
manav@ubuntulinux:~$ sudo chage -l test_user
Last password change : May 20, 2020
Password expires     : never
Password inactive   : never
Account expires      : May 29, 2020
Minimum number of days between password change : 0
Maximum number of days between password change : 99999
Number of days of warning before password expires: 7
```

This will change the expiration date of account "test\_user"

## 4. To change the group of a user

```
sudo usermod -g manav test_user
```



```
manav@ubuntulinux:~$ sudo usermod -g manav test_user
manav@ubuntulinux:~$ id test_user
uid=1002(test_user) gid=1000(manav) groups=1000(manav)
manav@ubuntulinux:~$
```

This command will now change the group of test user from test\_user to manav

## 5. To change user login name

```
sudo usermod -l test_account test_user
```



```
manav@ubuntulinux:~$ sudo usermod -l test_account test_user
manav@ubuntulinux:~$ id test_account
uid=1002(test_account) gid=1000(manav) groups=1000(manav)
manav@ubuntulinux:~$ id test_user
id: 'test_user': no such user
manav@ubuntulinux:~$
```

This will now change the login name of the user “test\_user”.

## 6. To lock a user

```
sudo usermod -L test_user
```

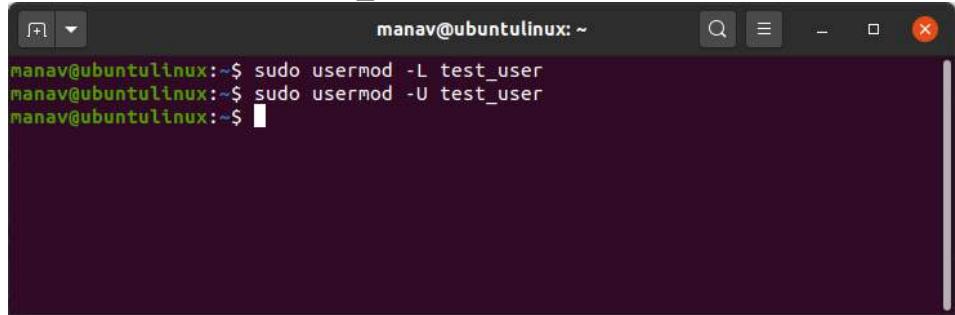


```
manav@ubuntulinux:~$ sudo usermod -L test_user
manav@ubuntulinux:~$ sudo usermod -U test_user
manav@ubuntulinux:~$
```

This will lock the “test\_user” account and will display a! sign in shadow file before the username

## 7. To unlock a user

```
sudo usermod -U test_user
```

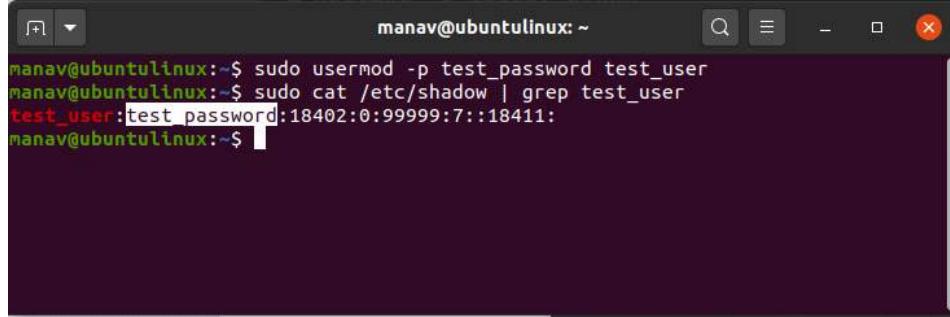


```
manav@ubuntulinux:~$ sudo usermod -L test_user
manav@ubuntulinux:~$ sudo usermod -U test_user
manav@ubuntulinux:~$
```

This will unlock the “test\_user” which was locked by the previous command

## 8. To set an unencrypted password for the user

```
sudo usermod -p test_password test_user
```

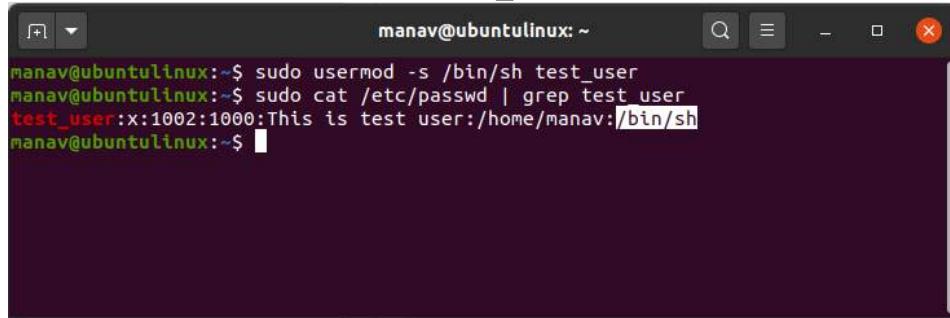


```
manav@ubuntulinux:~$ sudo usermod -p test_password test_user
manav@ubuntulinux:~$ sudo cat /etc/shadow | grep test_user
test_user:test_password:18402:0:99999:7::18411:
manav@ubuntulinux:~$
```

This will set the password “test\_password” in the unencrypted form for the user “test\_user”

#### 9. To create a shell for the user

```
sudo usermod -s /bin/sh test_user
```

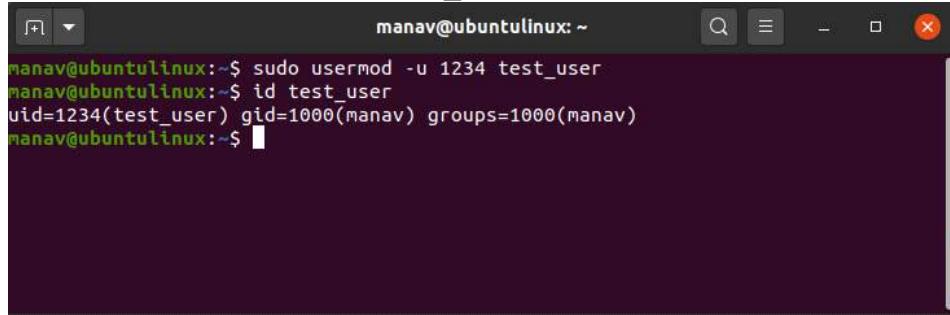


```
manav@ubuntulinux:~$ sudo usermod -s /bin/sh test_user
manav@ubuntulinux:~$ sudo cat /etc/passwd | grep test_user
test_user:x:1002:1000:This is test user:/home/manav:/bin/sh
manav@ubuntulinux:~$
```

This command will now create a shell for the user “test\_user” from /bin/sh

#### 10. To change the user id of a user

```
sudo usermod -u 1234 test_user
```



```
manav@ubuntulinux:~$ sudo usermod -u 1234 test_user
manav@ubuntulinux:~$ id test_user
uid=1234(test_user) gid=1000(manav) groups=1000(manav)
manav@ubuntulinux:~$
```

This command will change the user id of “test\_user” to 1234

# Password rules

Friday, June 12, 2020 11:02 AM

## Set Password Rules

2011/07/11

Set Password Policy to let users Comply rules.

- [1] Set number of days for password Expiration. Users must change their password within the days.

This setting impact only when creating a user, not impact to existing users.

If set to existing users, run the command "chage -M (days) (user)".

```
[root@dlp ~]#  
  
vi /etc/login.defs  
# line 17: set 60 for Password Expiration  
PASS_MAX_DAYS  
60
```

- [2] Set Minimum number of days available of password.

Users must use their password at least this days after changing it.

This setting impact only when creating a user, not impact to existing users.

If set to existing users, run the command "chage -m (days) (user)".

```
[root@dlp ~]#  
  
vi /etc/login.defs  
# line 18: set 2 for Minimum number of days available  
PASS_MIN_DAYS  
2
```

- [3] Set number of days for warnings before expiration.

This setting impact only when creating a user, not impact to existing users.

If set to existing users, run the command "chage -W (days) (user)".

```
[root@dlp ~]#  
  
vi /etc/login.defs  
# line 20: set 7 for number of days for warnings  
PASS_WARN_AGE  
7
```

- [4] Limit using a password that was used in past.

Users can not set the same password within the generation.

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 16: prohibit to use the same password for 5 generation in past  
password  
sufficient  
  
pam_unix.so sha512 shadow nullok try_first_pass use_authtok \
```

## remember=5

- [5] Set minimum password length.

Users can not set their password length less than set this parameter. ( minlen=N )

This setting linkages to other settings, so it need to set other settings like below.

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: set 8 for minimum password length  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type= \  
  
minlen=8 dccredit=0 uccredit=0 lccredit=0 occredit=0
```

- [6] In addition to the setting above, Set dccredit that forces users to include numbers in their password. ( dccredit=-N )

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: require to include 2 numbers in users password  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type= \  
  
minlen=8 dccredit=-2 uccredit=0 lccredit=0 occredit=0
```

- [7] In addition to the setting above, Set uccredit that forces users to include Capital characters in their password. ( uccredit=-N )

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: require to include 1 capital character  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type= \  
  
minlen=8 dccredit=-2 uccredit=-1 lccredit=0 occredit=0
```

- [8] In addition to the setting above, Set lccredit that forces users to include Lower cases in their password. ( lccredit=-N )

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: require to include 1 Lower case  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type= \  
  
minlen=8 dccredit=-2 uccredit=-1 lccredit=-1 occredit=0
```

```
minlen=8 dccredit=-2 uccredit=-1 lccredit=-1 ocredit=0
```

- [9] In addition to the setting above, Set ocredit that forces users to include Symbols in their password. ( ocredit=N )

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: require to include 1 Symbol  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type= \  
  
minlen=8 dccredit=-2 uccredit=-1 lccredit=-1 ocredit=-1
```

- [10] Set difok that forces more than N words in password before change are different from the one after change. ( difok=N )

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# near line 15: require at least 3 words are different from before change  
password  
requisite  
  
pam_cracklib.so try_first_pass retry=3 type=  
difok=3
```

- [11] Set number of login failure. Users' account will be locked after failing to login without a break.

```
[root@dlp ~]#  
  
vi /etc/pam.d/system-auth  
# add like follows ( this example sets login failure for 5 times. ( deny=5 ) )  
#%PAM-1.0  
# This file is auto-generated.  
# User changes will be destroyed the next time authconfig is run.  
auth  
required  
  
pam_env.so  
auth  
required  
pam_tally2.so deny=5  
auth  
sufficient  
  
pam_fprintd.so  
auth  
sufficient  
  
pam_unix.so nullok try_first_pass  
auth  
requisite
```

```
pam_succeed_if.so uid >= 500 quiet
auth

required

pam_deny.so
account
required

pam_unix.so
account
required
pam_tally2.so
account
sufficient

pam_localuser.so
account
sufficient

pam_succeed_if.so uid < 500 quiet
account

required

pam_permit.so
# make sure the number of failure of login about a user
[root@dlp ~]#
pam_tally2 -u cent
Login
Failures

Latest failure

From
cent

7

04/27/11 13:10:26

ttyS0
# unlock a locked user
[root@dlp ~]#

pam_tally2 -r -u cent
* sshd refers not to "system-auth" but to "password-auth", so if you apply login
failure setting for SSH, apply the same settings with above in
"/etc/pam.d/password-auth", too.
```

[12] Change password encryption algorithm.

This setting impact only when creating a user, not impact to existing users.

If set to existing users, run the command "chage -d 0 (user)" and let us change their password forcibly on next login.

```
# make sure current algorithm
[root@dlp ~]#
authconfig --test | grep hashing
password hashing algorithm is md5

# change algorithm to sha512
[root@dlp ~]#
authconfig --passalgo=sha512 --update
[root@dlp ~]#
authconfig --test | grep hashing
password hashing algorithm is sha512
```

From <[https://www.server-world.info/en/note?os=CentOS\\_6&p=password](https://www.server-world.info/en/note?os=CentOS_6&p=password)>

# Special Permission

Sunday, 5 February 2023 11:55 PM

- Linux offers 3 types of special permission bits that maybe set on executable files or directories to respond differently for certain operations.

These special permission bits are:

- setuid (set user identifier) bit
- setgid (set group identifier) bit
- sticky bit

- The setuid and setgid:

- These 2 bits can provide non-owners and non-group owners the ability to run executables with same access as the owner and group owner.

- Setuid bit on executable files:

Setuid bit flag can provide regular users the ability to run the same access as the owner of the executable file.

It is represented by an s in the owner's permission class.

Example:

```
#- ll /usr/bin/su  
-rwsr-xr-x. 1 root root 32096 Dec 1 18:28 /usr/bin/su  
s = setuid bit
```

- Login with a regular user and run su command.

- Lets remove the setuid flag

```
#- chmod u-s /usr/bin/su  
#- ll /usr/bin/su  
-rwxr-xr-x. 1 root root 32096 Dec 1 18:28 /usr/bin/su
```

Setuid bit has been removed, let try to run su again as a regular user.

su: Authentication failure

- To re-add setuid bit:

```
#- chmod u+s /usr/bin/su  
#- ll /usr/bin/su  
-rwsr-xr-x. 1 root root 32096 Dec 1 18:28 /usr/bin/su  
re-added setuid bit represented by s.
```

- You can also use numbers to apply setuid bit.

```
#- chmod u+s /usr/bin/su
```

or

```
#- chmod 4755 /usr/bin/su
```

4 adds the the setuid bit.

- To search for all files with setuid bit permissions:

```
#- find / -perm -4000
```

-----

Setgid bit on executable files:

- Setgid bit is set on executable files at the group level. Setgid bit flag can provide regular users the ability to run the same access as the group members of the executable file.

It is represented by an s in the group's permission class.

Example:

```
#- ll /usr/bin/wall
```

```
-r-xr-sr-x. 1 root tty 15344 Jun 9 2014 /usr/bin/wall
```

s is the setuidbit flag in group permissions

To remove setgid bit:

```
#- chmod g-s /usr/bin/wall
```

To add setgid bit:

```
#- chmod g+s /usr/bin/wall
```

or

```
#- chmod 2555 /usr/bin/wall
```

- To search for all files with setgid bit permissions:

```
#- find / -perm -2000
```

-----

- Stickybit:

Sticky bit is set on public writeable directories to prevent moving or deletion by regular users.

Example:

```
#- ll -d /tmp
```

```
drwxrwxrwt. 7 root root 93 May 11 04:00 /tmp/
```

t = stickybit in other's permissions

- The stickybit can be set by following command:

- Create a directory

```
#- mkdir dir1
```

-Add stickybit with rwx permissions to all

```
#- chmod o+t /tmp
```

```
#- chmod 777 /tmp
```

or

```
#- chmod 1777
```

```
#- ll dir1
```

- To remove stickybit

```
#- chmod o-t dir1
```

or

```
#- chmod 777 dir1
```

1 sets the sticky bit.

- To search for all files with stickybit permissions:

```
#- find / -type d -perm -1000
```

# Groups

Sunday, 13 November 2022 1:05 PM

Primary Group

Secondary group

A user can be assigned to max. 16 groups. ie., 1 primary group and 15 secondary groups

# Adding group

Sunday, 13 November 2022 1:06 PM

```
# groupadd <options><group name>
The options are, -f -----> add the group forcefully
-g -----> group id no.
-o ----->non-unique (duplicate group id)
-p -----> group password
-r -----> system group
-R -----> root group
```

```
# groupmod <options><group name>
The options are, -g -----> group id
-n -----> new name for existing one, ie., rename the group
-o -----> non-unique (duplicate group id)
-p -----> group passwd
-R ----->root group
```

## Assign Password

```
# gpasswd <group name> (to assign a password to the group without
any options)
# gpasswd <options><group name>
The options are, -a ----->add users to the group
-d -----> delete the user from the group
-r -----> remove the group password
-R -----> restrict to access that group
-A -----> set the list of Administrative users
-M -----> set the list of group members
```

# SUID GUID

Sunday, 13 November 2022 1:30 PM

## SUID

Allow the user to execute the root users cmd, applied for user level and is applicable for files only.

```
# chmod u+s <file name> (to set the uid on that file)
# chmod u-s <file name> (to remove the uid from that file)
# ls -l (if 'x' is replaced with 's' in owner's level permissions that means uid is applied on that
file)
- r w s r w x r w x <file name> (here 's' is called set uid or uid)
Example : # chmod u+s /usr/sbin/init 6 (then any user can restart the system using this
command #init 6)
# chmod u+s /sbin/fdisk (then any user can run the fdisk command)
```

## GUID

allow all the users of one group to get the group ownership permissions

Example: # chmod g+s <directory name> (to set the sgid on that directory)
# chmod g-s <directory name> (to remove the sgid from that directory)

# Links

Saturday, 12 November 2022 2:18 PM

making a connection between files.

1. The filename and it's associated inode number
2. An inode that describes the attributes of the file
3. The data associated with the file.

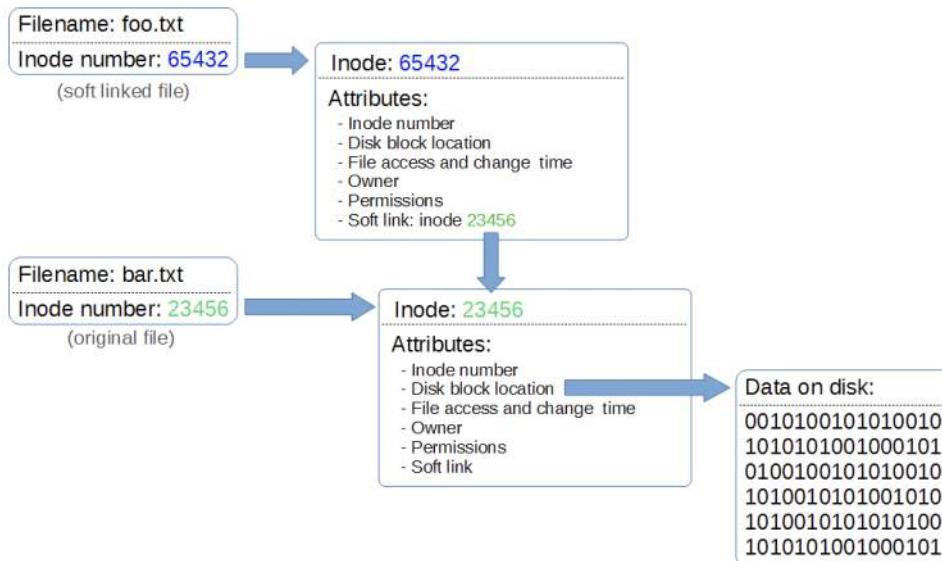
The file name and inode number point to an inode that has all the information about the file and  
in turn points to the data stored on the hard drive.

# Soft Links

Saturday, 12 November 2022 2:37 PM

## Soft Links (Symbolic links, Symlinks)

A symlink has a file name and inode number that points to its own inode,



```
$ ln -s [original filename] [link name]
```

inodes are restricted to the partition they are created on, a symbolic link allows redirecting to a path that can cross partitions.

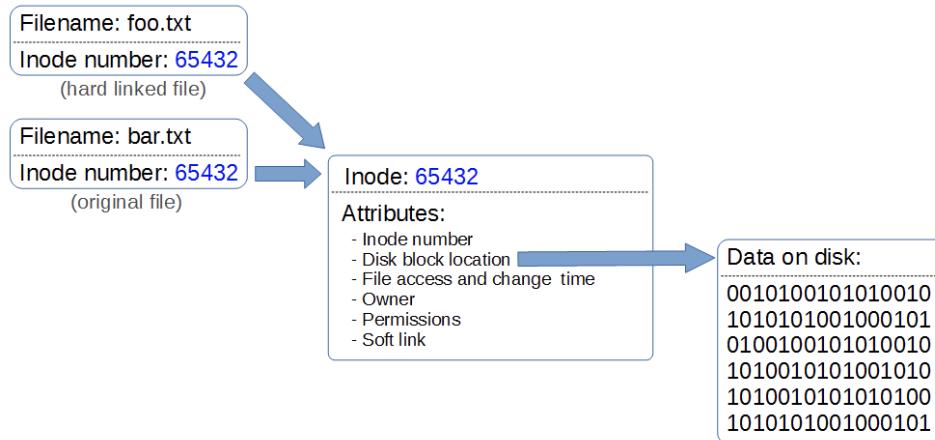
If we edit the linked file, the original file will be edited. If we delete the linked file, the original file will not be deleted. But if we delete the original file without deleting the link, we will be left with an orphaned link.

Soft links can link to directories and files but if the file is moved the link will no longer work.

# Hard Links

Saturday, 12 November 2022 2:38 PM

A hard link is one where more than one file name links to an inode.



```
$ ln [original filename] [link name]
```

This means that two file names are essentially sharing the same inode and data block, however they will behave as independent files. For example if we then delete one of the files the link is broken, but the inode and data block will remain until all the file names that link to the inode are deleted.

Hard links only link to a file (no directories), cannot span partitions and will still link to a file even if it is moved.

## Hard links

- Will only link to a file (no directories)
- Will not link to a file on a different hard drive / partition
- Will link to a file even when it is moved
- Links to an inode and a physical location on the disk

## Soft links (or symbolic links or symlinks)

- Will link to directories or files
- Will link to a file or directory on a different hard drive / partition
- Links will remain if the original file is deleted
- Links will not connect to the file if it is moved
- Links connect via abstract (hence symbolic) conventions, not physical locations on the disk. They have their own inode

# TLS / SSL

Saturday, 12 November 2022 3:32 PM

**TLS**, or transport layer security, and its predecessor **SSL**, which stands for secure sockets layer, are web protocols used to wrap normal traffic in a **protected, encrypted wrapper**.

- **openssl**: This is the basic command line tool for creating and managing OpenSSL certificates, keys, and other files.
- **req**: This subcommand specifies that you want to use X.509 certificate signing request (CSR) management. The “X.509” is a public key infrastructure standard that SSL and TLS adheres to for its key and certificate management. You want to create a new X.509 cert, so you are using this subcommand.
- **-x509**: This further modifies the previous subcommand by telling the utility that you want to make a self-signed certificate instead of generating a certificate signing request, as would normally happen.
- **-nodes**: This tells OpenSSL to skip the option to secure your certificate with a passphrase. You need Apache to be able to read the file, without user intervention, when the server starts up. A passphrase would prevent this from happening because you would have to enter it after every restart.
- **-days 365**: This option sets the length of time that the certificate will be considered valid. You set it for one year here.
- **-newkey rsa:2048**: This specifies that you want to generate a new certificate and a new key at the same time. You did not create the key that is required to sign the certificate in a previous step, so you need to create it along with the certificate. The **rsa:2048** portion tells it to make an RSA key that is 2048 bits long.
- **-keyout**: This line tells OpenSSL where to place the generated private key file that you are creating.
- **-out**: This tells OpenSSL where to place the certificate that you are creating.

sslcheck.myftp.org

```
sudo mkdir /etc/ssl/private  
sudo chmod 700 /etc/ssl/private
```

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out  
/etc/ssl/certs/apache-selfsigned.crt
```

```
sudo openssl dhparam -out /etc/ssl/certs/dhparam.pem 2048
```

```
# nano /etc/httpd/conf.d/ssl.conf
```

```
<VirtualHost *:443>  
    ServerName sslcheck.myftp.org  
    DocumentRoot /var/www/html  
    SSLEngine on  
    SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt  
    SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key  
</VirtualHost>
```

```
[root@ip-172-31-1-214 ~]# apachectl configtest  
Syntax OK  
[root@ip-172-31-1-214 ~]# apachectl restart
```

# Linux Cheat Sheet

Wednesday, December 2, 2020 10:28 PM

SYSTEM		FILE PERMISSION RELATED	
uname -a	=>Display linux system information	chmod octal file-name	=>Change the permissions of file to octal
uname -r	=>Display kernel release information	Example	
uptime	=>Show how long the system has been running + load	chmod 777 /data/test.c	=>Set rwx permission for owner,group,world
hostname	=>Show system host name	chmod 755 /data/test.c	=>Set rwx permission for owner,rx for group and world
hostname -i	=>Display the IP address of the host	chown owner-user file	=>Change owner of the file
last reboot	=>Show system reboot history	chown owner-user:owner-group file-name	=>Change owner and group owner of the file
date	=>Show the current date and time	chown owner-user:owner-group directory	=>Change owner and group owner of the directory
cal	=>Show this month calendar		
w	=>Display who is online		
whoami	=>Who you are logged in as		
finger user	=>Display information about user		
HARDWARE		NETWORK	
dmesg	=>Detected hardware and boot messages	ip addr show	=>Display all network interfaces and ip address (a iproute2 command, powerful than ifconfig)
cat /proc/cpuinfo	=>CPU model	ip address add 192.0.1 dev eth0	=>Set ip address
cat /proc/meminfo	=>Hardware memory	ethtool eth0	=>Linux tool to show ethernet status
cat /proc/interrupts	=>Lists the number of interrupts per CPU per I/O device	miitool eth0	=>Linux tool to show ethernet status
lshw	=>Displays information on hardware configuration of the system	ping host	=>Send echo request to test connection
lsblk	=>Displays block device related information in Linux	whois domain	=>Get who is information for domain
free -m	=>Used and free memory (-m for MB)	dig domain	=>Get DNS information for domain
lspci -tv	=>Show PCI devices	dig -x host	=>Reverse lookup host
lsusb -tv	=>Show USB devices	host google.com	=>Lookup DNS ip address for the name
dmidecode	=>Show hardware info from the BIOS	hostname -i	=>Lookup local ip address
hdparm -i /dev/sda	=>Show info about disk sda	wget file	=>Download file
hdparm -T /dev/sda	=>Do a read speed test on disk sda	netstat -tulp	=>Listing all active listening ports
badblocks -s /dev/sda	=>Test for unreadable blocks on disk sda		
USERS		COMPRESSION / ARCHIVES	
id	=>Show the active user id with login and group	tar cf home.tar home	=>Create tar named home.tar containing home/
last	=>Show last logins on the system	tar xf file.tar	=>Extract the files from file.tar
who	=>Show who is logged on the system	tar czf file.tar.gz files	=>Create a tar with gzip compression
groupadd admin	=>Add group "admin"	gzip file	=>Compress file and renames it to file.gz
useradd -c "Sam Tomshi"	=>g admin -m sam #Create user "sam"		
userdel sam	=>Delete user sam		
adduser sam	=>Add user "sam"		
usermod	=>Modify user information		
FILE COMMANDS		INSTALL PACKAGE	
ls -al	=>Display all information about files/ directories	rpm -i pkgname.rpm	=>Install rpm based package
pwd	=>Show the path of current directory	rpm -e pkgname	=>Remove package
mkdir directory-name	=>Create a directory		
rm file-name	=>Delete file		
rm -r directory-name	=>Delete directory recursively		
rm -f file-name	=>Forcefully remove file		
rm -rf directory-name	=>Forcefully remove directory recursively		
cp file1 file2	=>Copy file1 to file2		
cp -r dir1 dir2	=>Copy dir1 to dir2, create dir2 if it doesn't exist		
mv file1 file2	=>Rename source to dest / move source to directory		
In -s /path/to/file-name link-name	=>#Create symbolic link to file-name		
touch file	=>Create or update file		
cat > file	=>Place standard input into file		
more file	=>Output contents of file		
head file	=>Output first 10 lines of file		
tail file	=>Output last 10 lines of file		
tail -f file	=>Output contents of file as it grows starting with the last 10 lines		
gpg -c file	=>Encrypt file		
gpg file.gpg	=>Decrypt file		
wc	=>print the number of bytes, words, and lines in files		
xargs	=>Execute command lines from standard input		
PROCESS RELATED		INSTALL FROM SOURCE	
ps	=>Display your currently active processes	/configure	
ps aux   grep 'telnet'	=>Find all process id related to telnet process	make	
pmap	=>Memory map of process	make install	
top	=>Display all running processes		
kill pid	=>Kill process with mentioned pid id		
killall proc	=>Kill all processes named proc		
pkill process-name	=>Send signal to a process with its name		
bg	=>Resumes suspended jobs without bringing them to foreground		
fg	=>Brings the most recent job to foreground		
fg n	=>Brings job n to the foreground		
FILE TRANSFER		SEARCH	
scp		grep pattern files	=>Search for pattern in files
scp file.txt server2:/tmp		grep -r pattern dir	=>Search recursively for pattern in dir
rsync -a /home/apps /backup/		locate file	=>Find all instances of file
		find /home/tom -name 'index*'	=>Find files names that start with "index"
		find /home -size +10000k	=>Find files larger than 10000k in /home
LOGIN (SSH AND TELNET)		DISK USAGE	
ssh user@host		df -h	=>Show free space on mounted filesystems
ssh -p port user@host		df -i	=>Show free inodes on mounted filesystems
telnet host		fdisk -l	=>Show disks partitions sizes and types
DIRECTORY TRAVERSE		FILE TRANSFER	
cd ..	=>To go up one level of the directory tree	du -ah	=>Display disk usage in human readable form
cd	=>Go to \$HOME directory	du -sh	=>Display total disk usage on the current directory
cd /test	=>Change to /test directory	findmnt	=>Displays target mount point for all filesystem
		mount device-path mount-point	=>Mount a device

# Archive & Compression

Sunday, 13 November 2022 11:03 AM

# ZIP

Sunday, 13 November 2022 11:11 AM

Zip files

```
# zip filename filename
```

Zip Dir

```
# zip -r var-log-dir /var/log/
```

Unzip

```
# unzip var-log.zipn
```

List a content of zip file

```
# unzip -l var-log.zip
```

Validate a zip archive

```
# unzip -t log.zip
```

Password protected

```
# zip -P mysecurepasswd var-log-passwd.zip /var/log/*
```

Assign password without showing

```
# zip -e var-log-passwd.zip /var/log/*
```

# TAR Tape Archive

Sunday, 13 November 2022 11:11 AM

Single backup file

```
# tar cvf /tmp/dir.tar /home/mali
```

View all files

```
# tar tvf /tmp/dir.tar
```

Extract

```
# tar xvf /tmp/dir.tar
```

Extract tar.gz to specific directory

```
# tar xvzf /tmp/dir.tar.gz -C /home/mali
```

Combine gzip, bzip with tar

```
# tar cvfz /tmp/dir.tar.gz /home/mali
```

```
# tar xvzf /tmp/dir.tar.gz
```

```
# tar tvfz /tmp/dir.tar.gz
```

Use bzip2 with tar

```
# tar cvfj /tmp/dir.tar.bz2 /home/mali
```

```
# tar xvfh /tmp/dir.tar.bz2
```

```
# tar tvfh /tmp/dir.tar.bz2
```

# Memory Process, Zombie, Thread, Leak

Friday, June 19, 2020 3:50 PM

## ### ZOMBIE PROCESS ###

Let's say you one running process on system called (P1) and it is completed the execution and has to give exit status to Parent Process (P0). so let's say the parents process (P0) is on sleep mode then the (P1) will move to process table that call Zombie process until it exist in process table.

## ### PROCESSES ###

The Processes is any program that is running on your system.

# ps aux - will give you the list of processes (aux)

## ### THREAD ###

A Thread is the unit of execution within a process. A process can have anywhere from just one thread to many threads. So when the program is running threads comes to life. Previously one process can have only one process but now one process can have multiple threads. so when process running there are certain unit of execution and each unit known as a thread. Program --> Process --> Thread (Process Explorer)

## ### MEMORY LEAK ###

Memory - RAM on your computer

Leak - Loss of Memory

Memory leak is not for permanent and your not going to lose your RAM.

First thing how you gonna find out this issue is Computer slowness, less response or entire system stuck.

The main reason is Program holding memory but not actually using it - MEMORY LEAK.

The proper way is when program required memory then system provides it. but after the job done program has to return the memory back to the system.

Check the process page and note which process ID is using more RAM.

Garbage Collection: this program will run it will look for memory that is not in use and release it back to the system.

# Load Average + Inode + Hard Link

Wednesday, December 2, 2020 2:55 PM

## 25) What are hard links?

Hard links point directly to the physical file on disk, and not on the path name. This means that if you rename or move the original file, the link will not break, since the link is for the file itself, not the path where the file is located.

**Flags** indicate that a certain property, characteristic, or feature is active or is present. Certain flags are only supported on particular filesystems and partitioning tables. The flags (on GPT) are actually 128-bit GUIDs on the GUID partitioning table. Each flag has its own GUID.

## SAN vs NAS

SAN is a dedicated network of storage devices(can include tape drives storages, raid disk arrays etc) all working together to provide an excellent block level storage. WhileNAS is a single device/server/computing appliance, sharing its own storage over the network.

## Q:30 What is load average in a linux?

Ans: Load Average is defined as the average sum of the number of process waiting in the run queue and number of process currently executing over the period of 1,5 and 15 minutes. Using the 'top' and 'uptime' command we find the load average of a linux sever.

## What is an inode?

An inode is a data structure on a filesystem on Linux and other Unix-like operating systems that stores all the information about a file except its name and its actual data. like location of file on the disk, access mode, ownership, file type etc. it's large numbers that used in index and linux kernel used inode numbers to access the content inside the file system metadata (times of last change,[2] access, modification)

### Linux Permission

Read = 4

Write = 2

Execute = 1

User. Group. Everyone

# Proxy

Wednesday, 14 December 2022

4:49 PM

## Differences Between Forward Proxy and Reverse Proxy

The main difference between the two is that **forward proxy is used by the client** such as a web browser whereas **reverse proxy is used by the server** such as a web server. Forward proxy can reside in the same internal network as the client, or it can be on the Internet.

**Forward proxy can be used by the client to bypass firewall restrictions** in order to visit websites that are blocked by schools, governments, companies etc. If a website blocked an IP range from visiting the website, then a person in that IP range can use forward proxy to hide the real IP of the client so that person can visit the website and maybe leave some spam comments. However forward proxy might be detected by the website administrator. There are some paid proxy services that have numerous proxy systems around the world so that they can change your IP address every time you visit a new web page and this makes it harder for website administrators to detect.

## Reverse Proxy

Reverse proxy is mainly used by server admins to achieve **load balancing and high availability**. A website may have several web servers behind the reverse proxy. The reverse proxy server takes requests from the Internet and forwards these requests to one of the back-end web servers. Most visitors don't know websites are using reverse proxy because they usually lack the knowledge and tools to detect it or they simply don't care about it.

# Squid Proxy server

Wednesday, December 2, 2020 3:04 PM

Squid proxy server

```
mount -t iso9660 -o loop /home/iso/rhel-server-7.5-beta-1-x86_64-dvd.iso /mnt/iso
```

## 1 Understanding Proxy cache

- A forward cache accelerates client access to specific service
  - The client connects to the internet through the cache (not directly)
  - the cache remembers web pages which help making it faster
  - security can be applied also
  - proxy caching TLS/SSL is problematic and therefore not common
  - squid and apache can be used forward caches
- A reverse cache sits in front of web server (farm)
  - it makes connecting to the web server faster
    - squid, apache and Nginx can be used as reversed cache

## 2 Configuring squid for cache usage

- # yum search squid

```
# yum install squid
```

```
# cd /etc/squid (configuration dir)
```

```
# vim squid.conf
```

3128 is default port for squid

3 configuring browser to use squid

4 configuring squid ACLs

# Virtual host

Sunday, 5 February 2023 1:25 AM

What is meant by virtual host?

Virtual hosting is a **web server that appears as more than one host on the Internet**; the apparent host names distinguishes one host from another one. Using virtual hosting you can run multiple web services, each with a different host name and URL, that appear to be separate sites.

What are the types of virtual hosts?

There are two primary forms of virtual hosts: IP-based virtual hosts, where each virtual host has its own unique IP address; and name-based virtual hosts, where more than one virtual host runs on the same IP address but with different names.

# Cronjob

Sunday, 5 February 2023 12:01 PM

Logs /var/log/cron



```
# crontab -e  
# crontab -l
```

# Security

Monday, 6 February 2023 12:15 AM

## Introduction:

In this section we will cover things we can do to make your server more secure.

Keep in mind there is no such thing as 100% secure system and that is why you want to include regular backups and auditing as part of your security plan.

If your server is connected to the internet there is always a possibility that its security can be compromised so the idea behind security is to minimize the attacks on our system as much as possible.

If your security is tight then an attacker may move on to an easier target.

- Below are things you want to implement and consider when planning your system's security:

- Physical Security

- Keep the software up to date  
# yum update

- Employ the Principle of Least Privilege

- Use Encryption

- Avoid Non-Secure Protocols

- Clean up your systems  
# rpm -qa > /tmp/installedpackages.txt

- Minimize number of services per system

- Enforce a good password policy  
# chage  
# /etc/login.defs

- Disable root login  
# /etc/ssh/sshd\_config

- Disable Unneeded Services  
# chkconfig --list (RHEL 6)  
# systemctl list-unit-files (RHEL 7)

- Delete X Windows

```
# yum groupremove "X Windows System"
```

- Implement a Firewall

```
# firewalld daemon for RHEL 7 and iptables
```

Covered in detail in the Firewall section.

- Separate Partitions

```
/home  
/tmp  
/var  
/  
/boot
```

- Block SSH attacks

```
# /var/log/auth.log  
# /etc/hosts.deny  
# yum install denyhosts
```

Change default SSH port 22.

- Perform Security Scans and Audits

- nmap (yum install nmap)
- Nessus

- Use Sudo

```
# /etc/sudoers  
# visudo
```

- Use lastlog

```
# lastlog  
# lastlog -u username
```

Shows recent login info

- last command shows list of last logged in users

```
# last  
# last username
```

- Additional Security Tools

Wireshark  
Snort  
Aircrack  
John the Ripper

- Regular Backups

Free open source backup tools:

- Bacula
- rsnapshot
- DRBD
- Amanda

# OS Hardening

Thursday, 2 February 2023 11:04 PM

To checking our system is reaching to standards required by the organization.  
That is minimum password length, minimum size of root partition.  
Minimum free space and password expiry and all other security standards.

## Decommission and recommission

Thursday, 2 February 2023 11:06 PM

- (i) Normally servers should be changed every 5 - 6 years because of performance degradation as per standards of the company.
- (ii) Decommission means the process of removing the old system from the production environment and Recommission means the process of putting the new system into the production environment.
- (iii) We are not dedicated for decommission. We do decommission along with our routine work.
- (iv) Login as root though console.
- (v) First inform or raise the ticket to monitoring team to ignore the alerts.
- (vi) Stop the application and databases.
- (vii) Stop the cluster and Volume Manager.
- (viii) Unmount the file system.
- (ix) After that we should put the system for one week.
- (x) We will inform or raise the ticket to the network team to release the ports belonging to that system.
- (xi) Finally we inform to the data centre people to remove the cables from that system.

# Backup Policy

Thursday, 2 February 2023 11:08 PM

## Backup Procedure :

- (i) Deport the disk group on production server.
- (ii) Import the disk group on backup (media) server.
- (iii) Join the disk group with media server.
- (iv) Sync the data with production server.
- (v) Take the backup.
- (vi) split the disk group from media server.
- (vii) Join the disk group with production server.
- (viii) Deport the disk group from media server.
- (ix) Import the disk group on production server.

## Backup policy :

- (i) Complete (full) backup (every month ie., once in a month).
- (ii) Incremental backup (Daily).
- (iii) Differential or cumulative backup (every week end).

# File System is full?

Thursday, 2 February 2023 11:10 PM

- (i) First check whether the file system is O/S or other than O/S.
- (ii) If it is other than O/S, then inform to that respective teams to house keep the file system (ie., remove the unnecessary files in those file system).
- (iii) If not possible to house keep then inform to different teams (raise the CRQ (Change Request)) for increasing the file system.
  - (a) First take business approval and raise the CRQ to monitoring team to ignore the alerts from the system, Ask the application team to stop the application and database team to stop the database.
  - (b) Normally team lead or tech lead or manager will do this by initiate the mail thread.
  - (c) We will do this on weekend to reduce the business impact.
- (iv) First take a backup of the file system then unmount the file system.
- (v) Remove that partition and again create that file system with increased size, then mount again that file system and restore the backup.
- (vi) If the file system belongs to system log files or other log files and not to delete then they requested us to provide one Repository server (only for log files). Normally one script will do automatically redirect the log files to that repository server.
- (vii) Sometimes we will delete file contents not the files to reduce the file sizes. For that we execute the command # cat /dev/null ><file name with path> ie., nullifying the files.
- (ix) If it is root file system or O/S file system,**
  - (a) may be /opt full or may be /var full or may be /tmp full
  - (b) In /var/log/secure or /var/log/system or /var/tmp files may be full. If those files are important then redirect them to other central repository server or backup those files and nullifying those files.
  - (c) If /home directory is present in root ( / ) file system then this file system full will occur. Generally /home will be separated from root file system and created as separate /home file system.  
If /home is in root ( / ) as a directory then create a separate file system for /home and copy those files and directories belongs to /home and remove that /home directory.
  - (d) If root ( / ) is full then cannot login to the system. So, boot with net or CDROM in single user mode and do the above said.
- (x) Normally if file system is other than O/S then we will inform to that respective manager or owner and take the permissions to remove unnecessary files through verbal permission or CRQ .

## CPU utilization full?

Thursday, 2 February 2023 11:17 PM

- (a) Normally we get these scenarios on weekends because backup team will take heavy backups.
- (b) First check which processes are using more CPU utilization by # top and take a snap shot of that user processes and send the snap shot and inform to that user to kill the unnecessary process.
- (b) If those processes are backups then inform to the backup team to reduce the backups by stopping some backups to reduce the CPU utilization.
- (c) Sometimes in peak stages (peak hours means having business hours) CPU utilization will full and get back to the normal position automatically after some time (within seconds). But ticket raised by monitoring team. So, we have to take a snap shot of that peak stage and attach that snap shot to the raised ticket and close that ticket.
- (d) Sometimes if heavy applications are running and not to kill (ie., business applications), then if any spare processor is available or other low load CPUs available then move those heavy application processes to those CPUs.
- (e) If CPUs are also not available then if the system supports another CPU then inform to the data centre people or CPU vendor to purchase new CPU though Business approval and move some processes to the newly purchased CPUs.

## System is slow?

Thursday, 2 February 2023 11:19 PM

- (a) System slow means the end users response is slow.
- (b) Check the Application file system, CPU utilization, memory utilization and O/S file system utilization.
- (c) If all are ok, then check network statistics and interfaces whether the interfaces are running in full duplex mode or half duplex mode and check whether the packets are missing. If all are ok from our side then.
- (d) Inform to network team and other respective teams to solve this issue.

# Node is down?

Thursday, 2 February 2023 11:21 PM

- (a) Check pinging the system. If pinging, then check whether the system is in single user mode or not.
- (b) If the system is in single user mode then put the system in multi user mode ie., default run level by confirming with our team whether system is under maintenance or not.
- (c) Check in which run level the system is running. If it is in init 1 it will not be able to ping. If it is in init s then it will ping.
- (d) In this situation also if it is not pinging then try to login through console port. If not possible then inform to data centres people to hard boot the system.
- (d) If connected through console port then we may get the console prompt.

# Memory utilization full?

Thursday, 2 February 2023 11:23 PM

- (a) Check how much memory is installed in the system by # dmidecode -t memory command.
- (b) Check the memory utilization by # vmstat -v command.
- (c) Normally application or heavy backups utilize more memory. So, inform to application team or backup team or other teams which team is utilizing the more memory to reduce the processes by killing them or pause them.
- (d) Try to kill or disable or stop the unnecessary services.
- (e) If all the ways are not possible then inform to team lead or tech lead or manager to increase the memory (swap space). If it is also not possible then taking higher authority's permissions to increase the physical memory. For those we contact the server vendor and co-ordinate with them through data centre people to increase the RAM size.

# Failed hard disk?

Thursday, 2 February 2023 11:25 PM

- (a) Check whether the disk is failed or not by # iostat -En | grep -i hard/soft command.
- (b) If hard errors are above 20 then we will go for replacement of the disk.
- (c) If the disk is from SAN people then we will inform to them about the replacement of the disk. If it is internal disk then we raise the CRQ to replace the disk.
- (d) For this we will consider two things.
  - (i) whether the system is within the warranty.
  - (ii) without warranty.
- (f) We will directly call to the toll free no. of the system vendor and raise the ticket. They will issue the case no. This is the no. we have to mention in all correspondences to vendor regarding this issue.
- (g) If it is having warranty they ask rack no. system no. and other details and replace the hard disk with co-ordinate of the data centre people.
- (h) If it is not having warranty, we have to solve the problem by our own or re-agreement to extend the warranty and solve that problem.

# DB Patching

Friday, 3 February 2023 10:20 AM

(i) DBA team :

This is the team to apply the patches to the databases.

(ii) Linux team :

This team is also involved if any problems occur. If the database volume is having a mirror we should first break the mirror and then the DBA people will apply the patches. After 1 or 2 days there is no problem again we need sync the data between mirrored volume to patch applied volume. If there is no space for patch we have to provide space to DBA team.

(iii) Monitoring team :

This team should receive requests or suggestions to ignore any problems occurs. After applied the patch if the system is automatically rebooted then monitoring team will raise the ticket "Node down" to system administrators team. So, to avoid those type of tickets we have to sent requests to ignore those type alerts.

(iv) Application team :

For applying any patches, the databases should not be available to application. So, if suddenly database is not available then application may be crashed. So, first the application should be stopped. This will be done by application team.

# Kernel rollback

Friday, 3 February 2023 5:10 PM

# Logs

Sunday, 5 February 2023 4:41 PM

```
# journalctl
```

```
# journalctl -f
```

Kernel messages

```
# dmesg
```

```
# journalctl -k
```

Installation logs

```
# /var/log/anaconda/anaconda.log
```

# Limits

Sunday, 5 February 2023 4:46 PM

## To improve performance

```
# ulimit -a
```

```
# ulimit -u unlimited
```

Resource limit

# /etc/security/limits.conf

# Sar utility

Sunday, 5 February 2023 4:52 PM

```
# sar -u 2 3 (Shows realtime cpur
```

```
# sar -r memory utilizatio
```

```
# sar -S swap space
```

# Troubleshooting Best Practice

Thursday, June 25, 2020 6:02 PM

## Follow Policies and standards

- Communication (Why, What happen, When)
- Involve the right people
- Open up a ticket
- Resolution techniques or methods
- Maintenance schedules (M-F or weekends)
- Approval process

## Documentations or Ticketing Process

- Look for any existing documentation
- Wiki
- Ticketing system
- Post Mortem or Incident Report
- Root Cause Analysis
- Document Vendor recommendations
- Training.

## Patience To Work with Users / Groups.

- Do NOT panic.
- Involve others in your group to seek help (DB team, Leaders, Network, Application System)
- Work patiently with other groups
- Explain the situation when others get panic or annoyed.
- Setup a conference call to bring everyone together
- Handover to Someone with more experience (Do not take it personal)
- Do not let others derail you from the issue (Do not take off from the subject. Fucus on what is current going on.)

## Get Online Help

- Select a preferred search engine
- Look for specific error messages (Do not be too generic)
- Sign up for Linux community forums
- Ask questions if you cannot find your answer

## •Be aware of security issues

- Do not let anyone login to your system
- Hide IP address or hostname
- Do not search online from the server which is having issues
- Do not send any documents that has company information
- Do not download a file or script and run in your environment
- Make sure to add a thank you note to a helpful article.

## Understand the issue before making a decision

### **Ask yourself questions**

- Who is the source
- Who is the target
- What is the port number
- Who will be impacted
- Who do you need to notify
- Trace the issue.

### **Involve vendors if needed**

- Call for critical issues
- Have someone from your team reach to vendors
- Upload logs and allow vendors to look through the logs
- Allow access to login to the system
- Setup a call between the vendors to avoid finger pointing
- Record all vendors ticket notes and numbers.

### **Check for Logs**

- Systems
- Application
- Hardware
- Networking
- Trace for logs
  - e.g. webserver → application server → database server
- Copy the log error and search online
- Save the logs before system reboot.

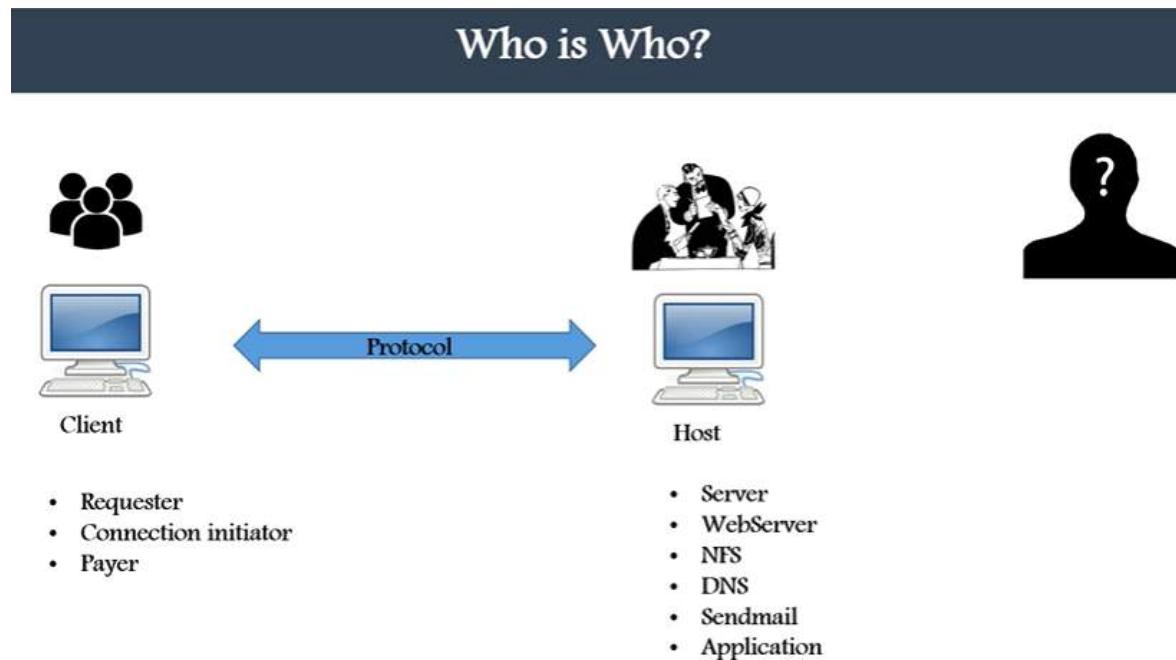
### **Be Honest and Ask Questions**

- Your mistake = admit it
- Don't make up if you have not find the problem
- Don't be afraid to ask questions
- What is it stands for
- How the issue was reported
- Why do you think its Linux related issue
- How the issue is resolved.'

# Conceptual Troubleshooting

Sunday, November 29, 2020 8:44 PM

## Who is Who ?



### Cannot connect to the server?

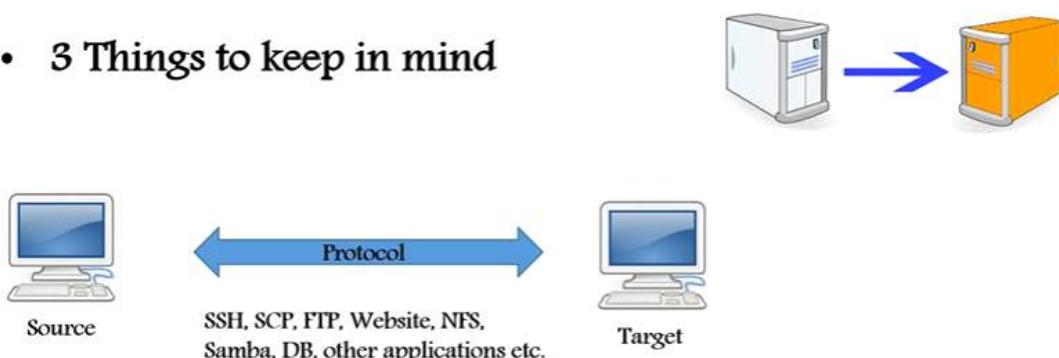
1. Find out the source from where they trying to connect.
2. What is the target. from where to where.
3. Which protocol they are using to connect source to target
  - SSH, SCP, FTP, NFS, SMB, DB, Website or other Application etc.

### Troubleshoot.

1. Is it the source or destination.
  - **Source** = Check physical connection, Check your own server network connectivity.
  - **Destination** = Try another server on the same network or different network.
  - **Protocol** = Check service functionality from S to D login to D check service functionality.
  - **Tool** = ping, telnet, curl etc.

## CANNOT ACCESS SERVER

- 3 Things to keep in mind



## VM running Slow

There are many reason for VM

- Memory allocation = CM
  - CPU assignment = VM
  - Network = VM (default 1 GB of network)
  - Storage I/O

## Cannot install Linux

- Look for the exact error message and search for it online.
  - If an error message specific to the vendor then lookup at vendor site for any KB articles known issues.
  - Validate install media (CD/DVD)
  - Make sure computer have enough resources. Eg RAM, Disk Space.
  - Try to reboot the computer.
  - If VM then try to re-create the VM.
  - If physical then make sure the ROM drive can read it or virtual drive is attached properly.

## Virtual Machine Run Slow

If VM then try to re-create the VM.

# Access Troubleshooting

Sunday, November 29, 2020 11:32 PM

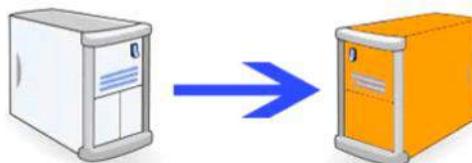
## System Access Troubleshooting

### INDEX

- **Server is not Reachable**
- **Cannot Connect to the Website or an application**
- **Cannot SSH as root or specific User.**
- **Firewall Issue**
- **Terminal Client is not working**
- **Cannot Connect using Putty to a VirtualBox VM**

### 1. Server is Not Reachable.

- Ping the destination server name.
  - If the server name not pingable
    - Ping with IP
- Use nslookup (Will return the server name and IP address)
- Check /etc/hosts file
- Check /etc/resolve.conf (DNS server details)
- Check /etc/nsswitch.conf (Tells you where to find the file)
  - ex: hosts: files dns myhostname
- Ping another server In same network.
- Check if your server has an IP address.
  - # ifconfig
  - # ip -a
  - # netstat -rnv (Check Gateway)
- Check physical cable connection
- Ping the destination server name
  - If server name is NOT pingable
- Ping the destination server by IP
  - If IP is pingable = *Name resolution issue*
    - Check /etc/hosts file
    - Check /etc/resolv.conf
    - Check /etc/nsswitch.conf
  - If IP is NOT pingable
    - Ping another server by name and then by IP
    - Checking if your server has an IP address
    - Ping your gateway/modem IP
    - Check physical cable connection

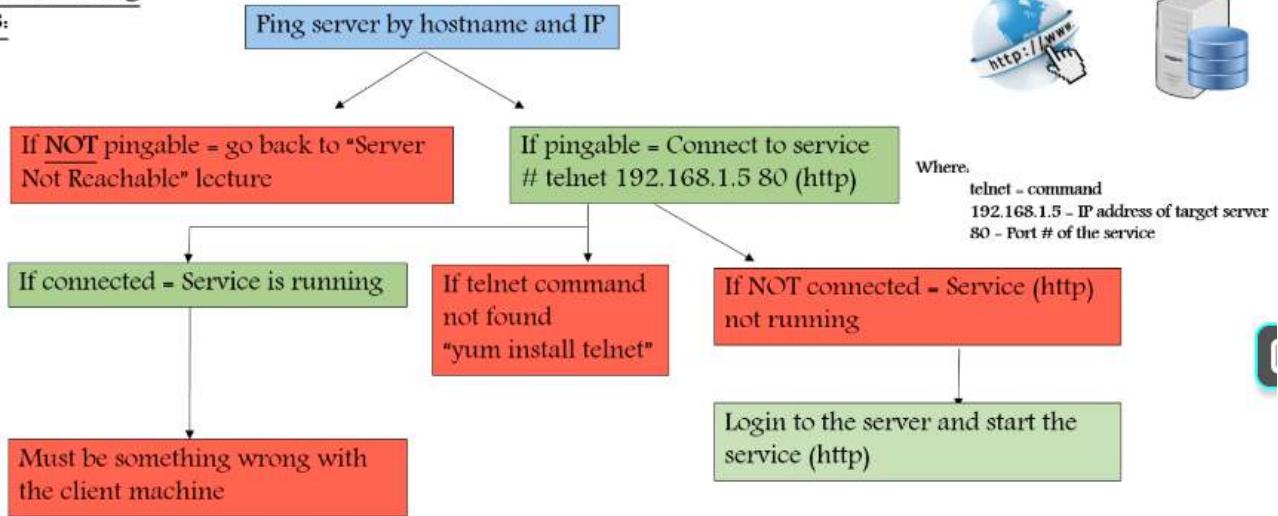


## 2. Cannot Connect to the Website or an application

- Ping the Server name / hostname or IP
- Telnet 192.168.1.5 80 (http)
- Check httpd service on server
- # ps -ef | grep -l network
- # ps -ef | grep ssh

### Troubleshooting

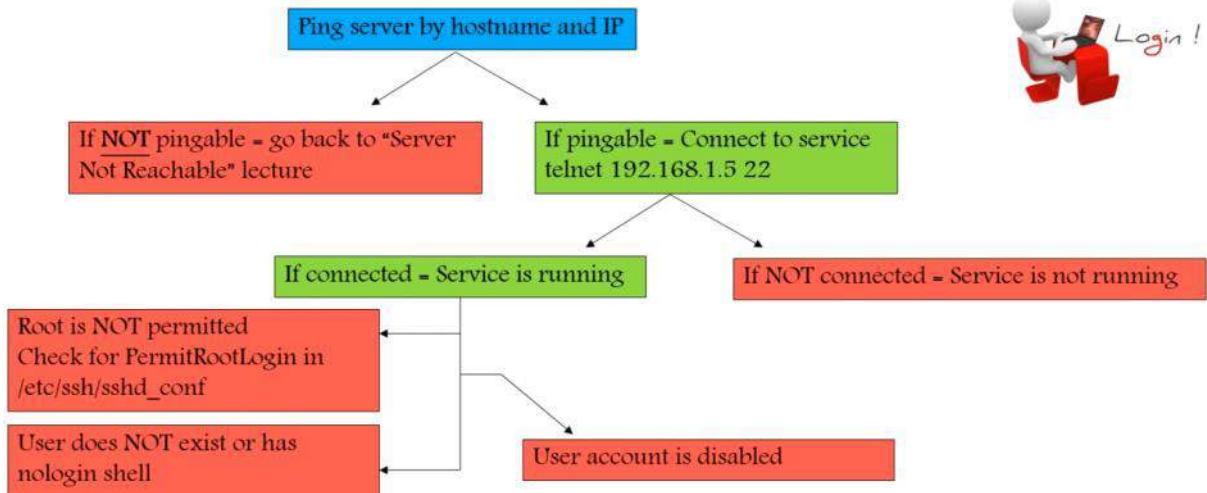
#### Steps:



## 3. Cannot SSH as root or specific User.

- Ping the Server name / hostname or IP
- If pingable = connect to service telnet IP 22
- If connect = check service is running
- cd /var/log | more secure (logs for login)
- User does not exist or has nologin shell check using "id iafzal"
- Check user entry in /etc/passwd
- User account is disabled check in /etc/passwd

### Troubleshooting Steps:

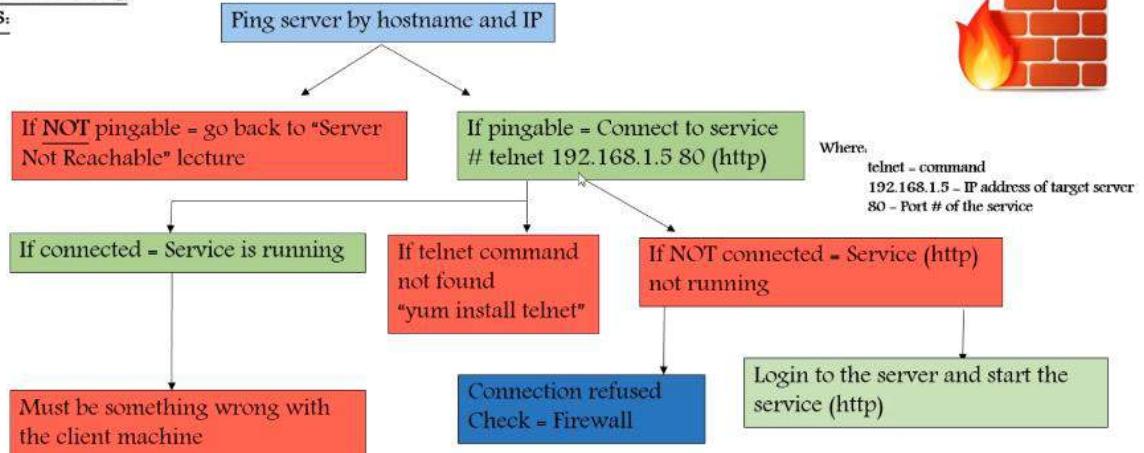


### 4. Firewall Issue

- Firewall is a wall that keeps away the fire
- Disable the firewall or do the configuration in firewall to allow the traffic
- # service iptables status
- # systemctl status iptables.service
- # systemctl status firewalld
- If any of the service is running you can configure it or disable it to allow the traffic.
- # systemctl disable firewalld

### Firewall

#### Troubleshooting Steps:



Where,  
telnet = command  
192.168.1.5 = IP address of target server  
80 = Port # of the service

### 5. Terminal Client is not working

- Client: Putty, CRS, terratorm, to connect machine remotely
- # ssh -l username X.X.X.X

# Filesystem Troubleshooting

Tuesday, December 1, 2020 9:54 AM

1. Cannot cd into a directory
2. Cannot Open a file or Run script
3. Having trouble finding files and directory
4. Cannot Create Links
5. Cannot Write to file
6. Cannot Delete, Copy, Move or rename a File
7. Cannot Change file permission or view other users files
8. Disk space full or add more disk space
9. Add disk and create standard partition
10. Add Disk and create LVM Partition
11. Extend disk with LVM
12. How to delete Old files
13. Script to delete old files
14. Filesystem is corruption
15. Corruption in /etc/fstab

## 1. Cannot cd into a directory

- Directory doesn't exist or check for Case sensitive
- Absolute Path - Path to directory correct or not
- Permissions problem on directory or file
  - -rwxrwxrwx
  - U | G | O
- File Type
- Parent directory Permission
- Hidden Directory

# Cannot open file or run script

Wednesday, 1 February 2023 12:51 AM

- File doesn't exist
- File content is there
- Absolute vs Relative Paths
- Permissions
- File type
- Hidden Directory

# Trouble Finding files and Dir

Wednesday, 1 February 2023 12:53 AM

- Run the right command - **find**
- Command syntax

```
# find / -name "susan"
```

- Additional command - **locate**

```
# updatedb  
# locate susan
```

- File/Directory exist

# Cannot create links

Wednesday, 1 February 2023 12:59 AM

## Link - Shortcuts

- Command and Syntax (Source first and then target)
- Complete paths  
`# ln -s /home/source.txt /tmp/target.txt`
- Permissions
- Source or target file/Dir missing or doesn't exist.

# Cannot Write to file

Wednesday, 1 February 2023 1:06 AM

- File doesn't exist
- Absolute vs. Relative Paths
- File Type (File or Dir or dev)
- Permissions and ownership
- Parent dir permissions
- Hidden File
- File is already open by another user

Run command "getfact" (Get file access control lists).

```
# getfac susan
```

# Delete, Copy, Rename or Move a file

Wednesday, 1 February 2023 1:12 AM

- File doesn't exist
- Absolute vs. Relative Paths
- File Type (File or Dir or dev)
- Permissions and ownership
- Parent directory permissions
- Hidden File
- Command syntax (Source and then target)

## **Delete old files**

```
# find /path/to/files/ -type f -name '*.jpg' -mtime +30 -exec rm {} \;
```

## **Create old date file**

```
# touch -d "Thu, 1 March 2018 12:30:00" a
```

## **Script**

```
#!/bin/bash
# this script will delete file older than 90 days
# find /home/iafzal/ps -mtime +90 -exec ls -l {} \;
# find /home/iafzal/ps -mtime +90 -exec rm -rf {} \;
# find /home/iafzal/ps -mtime +90 -exec mv {} {}.old \;
```

# Change File Permission / Ownership

Wednesday, 1 February 2023 1:20 AM

- File doesn't exist
- Absolute vs. Relative Paths
- File ownership (user and group)
- Permissions problem on file
  - -rwxrwxrwx
  - U | G | O
- Parent directory permissions
- Hidden File
- Command syntax (User and then file) - chown/chgrp user/group filename
- Command with recursive option (-R)
- Only root can change the file ownership

# Disk space full - Add more

Wednesday, 1 February 2023 1:29 AM

- What = df
- Actions
  - o Find out what is causing the issue (du)  
# du -a / | sort -nr | more / | du -sh
  - o Free up disk space by deleting old or big files
  - o Compress the old files (tar -czvf archive.tar.gz file)
  - o Move files to another partition or another server
  - o Check harddisk status  
# badblocks -v /dev/DISK or /dev/sda  
# iostat
  - o Create a link to another filesystem or directory
  - o Add additional disk and extend using LVM

```
# du -S | sort -n | more
```

## Script to Monitor

```
$ df -H | grep -vE '^Filesystem|tmpfs|cdrom' | awk '{ print $5 " " $1 }'  
$ df -H | grep -vE '^Filesystem|tmpfs|cdrom|loop|udev' | awk '{ print $5 " " $1 }'
```

```
#!/bin/sh  
# Purpose: Monitor Linux disk space and send an email alert to $ADMIN  
ALERT=90 # alert level  
ADMIN="you@cyberciti-biz" # dev/sysadmin email ID  
df -H | grep -vE '^Filesystem|tmpfs|cdrom' | awk '{ print $5 " " $1 }' | while read -r output;  
do  
echo "$output"  
usep=$(echo "$output" | awk '{ print $1}' | cut -d'%' -f1 )  
partition=$(echo "$output" | awk '{ print $2 }')  
if [ $usep -ge $ALERT ]; then  
echo "Running out of space \"$partition ($usep%)\" on $(hostname) as on $(date)" |  
mail -s "Alert: Almost out of disk space $usep%" "$ADMIN"  
fi  
done
```

```
=====
```

```
#!/bin/sh  
# set -x  
# Shell script to monitor or watch the disk space  
# It will send an email to $ADMIN, if the (free available) percentage of space is >= 90%.  
# -----  
# Set admin email so that you can get email.  
ADMIN="root"  
# set alert level 90% is default  
ALERT=90  
# Exclude list of unwanted monitoring, if several partitions then use "|" to separate the partitions.  
# An example: EXCLUDE_LIST="/dev/hdd1|/dev/hdc5"  
EXCLUDE_LIST="/auto/ripper|loop"
```

<https://www.unisys.com>

[http://www.gmx.com/vern/programming\\_and\\_scripting/192096/the\\_system\\_exceed\\_alert\\_script.htm](http://www.gmx.com/vern/programming_and_scripting/192096/the_system_exceed_alert_script.htm)

<https://www.zuaygeek.com/shell-script-monitor-disk-space-usage-linux/>

# Adding disk | Partition

Wednesday, 1 February 2023 1:38 AM

Purpose? = Out of space, Additional apps etc.

Commands for disk partition

- Df
- fdisk

# File system corrupted

Wednesday, 1 February 2023 12:31 PM

## Types of FS

- Ext3, ext4, xfs, NTFS etc.

## FS layout and partition

- /var, /etc, /root, /home etc.

## Checking FS

- # df
- # fdisk -l

## Troubleshooting steps.

- Check /var/log/messages or /var/log/syslog
- Run fsck on block device (/dev/sda) Not on mount point
- Unmount filesystem and run fsck

## Rescue

- Boot linux
- Troubleshoot
- Rescue a linux system
- Choose 3 to shell mode
- # fsck.xfs -y /dev/sda1
- # xfs\_repair /dev/sda1

# Corruption in fstab

Wednesday, 1 February 2023 12:41 PM

- OS filesystem table
- Device, Mount point, FS type, option, backup, filesystem check order

Checking FS

```
# df
```

Checking or verifying /etc/fstab. Take backup before editing

```
# cat, more, vi, vim etc
```

Issue - System is not booting

- Incorrect entry in /etc/fstab
- Accident deletion of /etc/fstab

Troubleshooting

- Boot in rescue mode by mounting CD / CD ISO image
- Choose option 1 to mount root filesystem
- Fix the /etc/fstab
- For deleted file - run blkid

# System Administration

Wednesday, 1 February 2023 12:58 PM

# Running out of Memory

Wednesday, 1 February 2023 1:01 PM

## Memory (RAM)

- Random-access memory is a form of computer data storage that stores data and machine code currently being used

## Swap (virtual memory)

- Memory carved out of a hard disk. It functions just like RAM but it is slower than RAM

## Cache

- This memory is typically integrated directly into the CPU chip or placed on a separate chip that has a separate bus interconnect with the CPU. The purpose of cache memory is to store program instructions and data that are used repeatedly

## Troubleshooting Commands:

- **free -m**(mega bytes) or **-h** (human readable)
- **top**
- **vmstat**
- **dmesg | grep -i "Out of memory" (/var/log/messages OR /var/log/syslog)**
- Memory commitment in configuration file **/etc/sysctl.conf**.

## Troubleshooting Steps:

- Identify the process causing the memory usage
- top, ps
- Kill or restratthe process causing high memory utilization
- **kill, service, systemctl**
- Prioritize the process
- nice
- Add new swap space
- Create a dedicate file for swap
- Assign it to swap
- Enable swap
- Extend swap space
- Add additional memory from virtualization
- Add additional physical memory.

## Commands

```
# dd
    # dd if=/dev/zero of=/newswap bs=1M count=1024
    # chmod go-r or 600 newswap
# mkswap /newswap
# swapon /newswap
# swapoff
```

# System Rebooted or Process Restarted

Wednesday, 1 February 2023 1:13 PM

## System reboot / crash

- Memory stress
- CPU stress
- Kernel panic
- Hardware issue

## Process restart

- System reboot
- Process restarted itself (`systemctl status process`)
- Watchdog application

## Troubleshooting Steps.

- Login through SSH and the following commands to troubleshoot  
**Uptime, top, dmesg, iostat -xz 1, journalctl**
- **Go through the logs**  
**/var/log/messages, /var/log/syslog, /var/log/boot.log**  
**For application, check app logs**
- Login through console (iLO, iDRAC, Virtual console etc)
- Reach out to the vendor (Redhat, SUSE, Oracle etc)
- For hardware, check logs from the console and reach out to the vendor

# Unable to get an IP Address

Wednesday, 1 February 2023 5:24 PM

- Troubleshooting Steps:

- Check your DHCP server (Modem at home)
- Check network setting at virtualization level
- Check interface at the hardware level
  - `lspci | egrep -i 'eth|wifi|wireless'`
  - `nmcli -p dev`
- Check your interface (`ifconfig` or `ip addr`)
- Check whether you are connected as wireless or wired network
- Difference between `ifup <interface>` and `ifconfig up <interface>`

- Restart network services `systemctl restart network`

- For static IP verify the network configuration files

- `/etc/sysconfig/network-scripts/ifcfg-enp0s3` or `ifcfg-eth0`

- `DEVICE=eth0`

- `BOOTPROTO=none`

- `ONBOOT=yes`

- `PREFIX=24`

- `IPADDR=192.168.1.200`

# IP Assigned but not Reachable

Wednesday, 1 February 2023 5:28 PM

- Troubleshooting Steps:
- Check if you are on the correct network interface (**ifconfig**)
- Check to see if you got the right subnet mask or gateway
- Ping the gateway
- Check if the gateway is assigned (netstat-rnv)
- Check with network team if the correct vLANs assigned on the switch side
- Run **ethtool** or **mii-tool** to check the NIC status
- Run **ifup<interface>** command to bring the NIC port up
- Restart network **systemctl restart network**
- Check on the status of NIC by running **ifconfig** or **ipaddr** command
- Check to see if the IP is assigned to some other device (IP conflict)
- Turn off firewall

# Password issue

Wednesday, 1 February 2023 5:36 PM

- Two important files for password  
`(/etc/passwd and /etc/shadow)`
- Issues:
- User is added manually in `/etc/passwd` file and `/etc/shadow`

file has no information (`passwduser`)

- `/etc/shadow` file is corrupted
- `/etc/shadow` file is missing (`pwconv` will recreate the file)

- Troubleshooting Steps:
  - Correct user
  - Provide current password first before entering new password
  - You have to be root to set other user passwords
- Make sure to specify the user after `passwd` command
- Make sure to run `pwconv` after creating a user in `/etc/passwd` file
- Fix the `/etc/shadow` file
- Run the `pwconv` command to recreate the `/etc/shadow` file. (Remember you will have to setup passwords for each user again).

# Broadcast message

Sunday, 5 February 2023 4:06 PM

```
[root@jenkins tmp]# wall -n "Server is going down for maintenance in 5 minutes, Please save  
your data"
```

```
[root@jenkins tmp]#
```

```
Remote broadcast message (Sun Feb 5 16:07:33 2023):
```

```
Server is going down for maintenance in 5 minutes, Please save your data
```

# Reboot in 15 min

Sunday, 5 February 2023 4:10 PM

Reboot in 15 minutes  
# shutdown -r +15

Reboot at 11 PM  
# shutdown -r 23:00

# Full Duplex / Half Duplex

Friday, 3 February 2023 5:40 PM

# Network

Monday, October 3, 2022 5:12 PM

## IP Address

```
# ip a  
# ip addr show  
# ifconfig
```

## 2nd Layer

```
# ip link show ens03
```

## ARP table

```
# ip neigh show  
# arp -n
```

## Routing table

```
# ip route  
# route  
# netstat -r
```

## Ports

```
# netstat -tulpn
```

## Remote Port

```
# telnet
```

## Network traffic

```
# tcpdump -l ens33 -w ./test.pcap
```

# DNS / DHCP

Friday, June 19, 2020 3:51 PM

## Managing a DNS Server

### 1 Understanding DNS Name Resolution

- DNS (Domain name system), that resolve domain names to IP address.
- So if you type in web address in your computer.
- The DNS server will search through its database to find a matching IP address for that domain name
- When it finds it, it will resolve that domain name to the IP address of your website.
- It works like phone directory so when you want to search for any number you search for the name.

### ISP or Resolver

- Search for any domain name
- Your system doesn't find it in its own cache memory
- It will send the query to next level that is called resolvers server.
- It is your network provider that's call ISP (Internet Service provider)
- So when resolver receive your query then it will find it on its own cache memory to find IP address of that website.
- If it can't find it then it will send that query to next level which is the root server.

### Root Server

- The root server are the top or root of DNS hierarchy.
- There are 13 sets of these root servers and they are placed around the world.
- Operated by 12 different organizations.
- Each set has their own unique IP address.
- Root server don't know the IP address of your website but it does know where to send the resolver to help it find the IP address.
- root server will send it to the TLD, the top-level domain server for the dot com domain.

### TLD (Top level domain)

- it contains the top-level domain name such as .com .net or .org.
- This particular TLD server manages the dot com domain which amazon.com is the part of.
- So when TLD receive the query for the IP address for your amazon.com the TLD server is not going to know what the IP address for amazon.com
- Then TLD will redirect the resolver request to the next and final level authoritative name servers.

### Authoritative name server

- they are responsible for knowing everything about the domain.
- Includes the IP Address, they are the final authority, so, when the authoritative name server receives the query from the resolver, the name server will respond with the IP address for your website. finally, the resolver will tell your computer the IP address of your website and it will retrieve your webpage.
- Once the resolver receive the IP address it will store it in its cache memory in case it receives another query for same domain name. So it doesn't have to go through all the process again.

## DNS PROPAGATION

DNS Propagation refers to the time for any DNS changes to transmit across the Internet. Please remember that DNS changes in general can take up to 24-48 hours to fully propagate.

## DNS RECORDS

The root domain (also sometimes referred to as the "parent," "naked," or "apex" domain) is the primary entry for the domain without any subdomains. The NAME field typically remains blank as this would define a subdomain. This type of record should usually be an A record, with the value set to the destination IP address. Using a CNAME for the root domain can cause other DNS functions, such as MX records, to route incorrectly. It is standard practice to set the A record for the root domain to that of the ["www"](#) subdomain.

### CNAME or "Canonical Name"

CNAME Records are used to define an alias hostname. A CNAME record takes this format:

alias.domain.name IN CNAME otherhost.domain.name.

This defines alias.domain.name as an alias for the host whose canonical (standard) name is otherhost.domain.name.

### A Record

An A record gives you the IP address of a domain. That way, users that try to go to [www.example.com](#) will get to the right IP address. An A record or "Address Record" maps a hostname to a 32-bit IPv4 address. An "A" Record takes this format (example):

Name TTL TYPE DATA  
[ftp.domain.com](#) 43200 A IP Address

Media Temple DNS Zone files are written with a "wildcard" entry, that looks like this:

\*.domain.com IN A xxx.xxx.xxx.xxx

The x's represent your particular IP address. The star takes "anything" .domain.com and points it to your server's IP address. This way, if someone mistakenly types too many or too few w's, they'll still see your website. This is also useful for setting up subdomains on your server, relieving you of the duty of adding an additional "A" record for the subdomain.

### MX Record

Mail Exchange Record: Maps a domain name to a list of mail exchange servers for that domain. A zone can have one or more Mail Exchange (MX) records. These records point to hosts that accept mail messages on behalf of the host. A host can be an 'MX' for itself. MX records need not point to a host in the same zone. An 'MX' record takes this format:

host.domain.name IN MX 10 otherhost.domain.name.  
IN MX 20 otherhost2.domain.name.

### PTR Record / Pointer Record

Maps an IPv4 address to the canonical name for that host. Setting up a PTR record for a hostname in the in-addr.arpa. domain that corresponds to an IP address implements reverse DNS lookup for that address. For example, at the time of writing, [www.icann.net](#) has the IP address 192.0.34.164, but a PTR record maps 164.34.0.192.in-addr.arpa to its canonical name.

### NS Record or "Name Server Record"

Maps a domain name to a list of DNS servers authoritative for that domain. In this case, for (mt) Media Temple purposes

## DHCP (Dynamic Host Configuration Protocol)

- Every Computer or device on network has to have an IP address for Communication purpose.
- IP is an identifier for computer or device on network.
- There are two ways that computer can be assigned an IP address.
- That could be done by using a static IP or dynamic IP.
- A Static IP where you assign the IP manually with subnet, gateway and DNS server.
- In large network it's bit difficult to do this manually typing work.
- Better way to assign the IP address that's is dynamic IP.
- Computer get IP address automatically from DHCP server.
- In Dynamic setting computer broadcast a request for an IP on network then DHCP will assign the IP from its pool and deliver to the computer.
- DHCP assign the IP from its scope and scope is the range of IP that DHCP server can hand out. It is customizable.
- Computer doesn't own this IP. It's actually a lease to make sure DHCP doesn't run out of IP in its scope.
- Address reservation

would be:

ns1.mediatemple.net  
ns2.mediatemple.net  
SOA Record or "Start of Authority Record"

Specifies the DNS server providing authoritative information about an Internet domain, the email of the domain administrator, the domain serial number, and several timers relating to refreshing the zone.

#### **TXT Record**

The TXT Record allows an administrator to insert arbitrary text into a DNS record. For example, this record is used to implement the Sender Policy Framework and DomainKeys specifications.

## Add Static IP

Wednesday, July 14, 2021 10:54 PM

```
# nmcli con add con-name ens37 ifname ens37 type ethernet ip4 192.168.100.60/24 gw4 192.168.100.1  
# ls /etc/sysconfig/network-scripts/  
# nmcli dev status  
# nmcli con show ens37  
# nmcli con mod ens37 ipv4.dns 192.168.100.1  
# nmcli con show ens37  
# nmcli con modify ens37 ip4 192.168.100.1/24
```

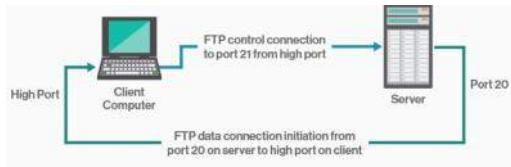
## Network Protocol Port

Tuesday, December 8, 2020 5:23 PM

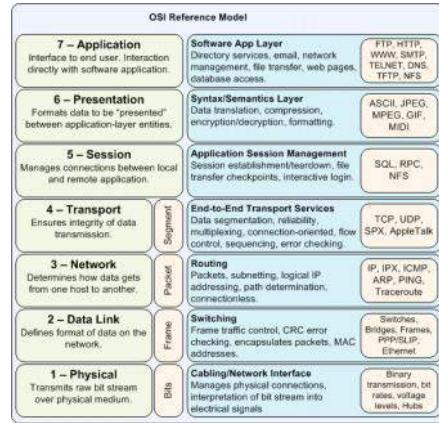
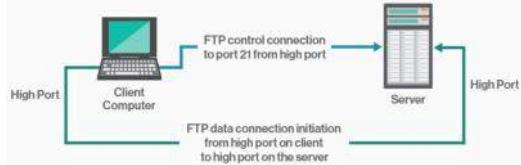
1. **FTP - 21 - TCP/IP**
2. **SFTP - 22 - SSH** Secure File Transfer Protocol. SFTP (SSH File Transfer Protocol) is a secure file transfer protocol. It runs over the SSH protocol.
3. **SNMP 161 - Simple Network Management Protocol (SNMP)** is an Internet Standard protocol for collecting and organizing information about managed devices on IP networks and for modifying that information to change device behavior.
4. **SMTP - 25 : The Simple Mail Transfer Protocol** is a communication protocol for electronic mail transmission. As an Internet standard, SMTP was first defined in 1982 by RFC 821, and updated in 2008 by RFC 5321 to Extended SMTP additions, which is the protocol variety in widespread use today.
5. **NFS - 2049 :**
6. **FTFP - 69 : Trivial File Transfer Protocol** is a simple lockstep File Transfer Protocol which allows a client to get a file from or put a file onto a remote host.
7. **DNS - 53 :**
8. **DHCP - 67, 68 :**
9. **NTP - 123 :**
10. **LDAP - 389, 636 :**
11. **HTTP - 80, 443 :**
12. **CURL : curl** is a command-line utility for transferring data from or to a server designed to work without user interaction. With curl, you can download or upload data using one of the supported protocols including HTTP, HTTPS, SCP, SFTP, and FTP.
13. **TELNET - 23 :**
14. **IMAP - 143 :**
15. **POP3 - 110 :**
16. **SSH - 22 :**
17. **MySQL - 3306**

**The Port no. list :**

FTP (For data transfer)	20	HTTP	80
FTP (For connection)	21	POP3	110
SSH	22	NTP	123
Telnet	23	LDAP	389
Send Mail or Postfix	25	Log Server	514
DNS	53	HTTPS	443
DHCP (For Server)	67	LDAPS (LDAP + SSL)	636
DHCP (For Client)	68	NFS	2049
FTFP (Trivial File transfer)	69	Squid	3128
Samba shared name verification	137	Samba Data Transfer	138
Samba Connection Establishment	138	Samba Authentication	445
MySQL	3306	iSCSI	3260



### Passive FTP



Layer	Function	Example
<b>Application (7)</b>	Services that are used with end user applications.	SMTP,
<b>Presentation (6)</b>	Formats the data so that it can be viewed by the user.	JPG, GIF, HTTPS, SSL, TLS
<b>Session (5)</b>	Establishes/ends connections between two hosts.	NetBIOS, RPTP
<b>Transport (4)</b>	Responsible for the transport protocol and error handling.	TCP, UDP
<b>Network (3)</b>	Reads the IP address from the packet.	Routers, Layer 3 Switches
<b>Data Link (2)</b>	Reads the MAC address from the data packet.	Switches
<b>Physical (1)</b>	Send data on to the physical wire.	Hubs, NICs, Cable

TCP/IP	OSI Model	Protocols
<b>Application Layer</b>	<b>Application Layer</b>	DNS, DHCP, FTP, HTTPS, IMAP, LDAP, NTP, POP3, RTP, RTSP, SSH, SIP, SMTP, SNMP, Telnet, TFTP
	<b>Presentation Layer</b>	JPEG, MIDI, MPEG, PICT, TIFF
	<b>Session Layer</b>	NetBIOS, NFS, PAP, SCP, SQL, ZIP
<b>Transport Layer</b>	<b>Transport Layer</b>	TCP, UDP
<b>Internet Layer</b>	<b>Network Layer</b>	ICMP, IGMP, IPsec, IPv4, IPv6, IPX, RIP
	<b>Data Link Layer</b>	ARP, ATM, CDP, FDDI, Frame Relay, HDLC, MPLS, PPP, STP, Token Ring
<b>Link Layer</b>	<b>Physical Layer</b>	Bluetooth, Ethernet, DSL, ISDN, 802.11 Wi-Fi

# NIC Teaming / BOND

Monday, September 13, 2021 12:29 PM

## 1. How do you perform NIC teaming? Network Interface Card

NICteaming/bonding is used mostly in scenarios where customer cannot afford to loose connectivity due to ethernet failover issues and also it has many other advantages like to **distribute bandwidth, fault tolerance** etc.

## 3. What are the benifits of NIC Teaming?

Load balancing Fault Tolerance Failover

### Q:16 How to check and verify the status the bond interface.

Ans: Using the command '**cat /proc/net/bonding/bond0**', we can check which mode is enabled and what lan cards are used in this bond. In this example we have one only one bond interface but we can have multiple bond interface like bond1,bond2 and so on.

#### CentOS 6

NIC driver exist - allow binding  
Physical NIC - At least 2 eth

Master + Slave file

1. As root, create a Bond0 Configuration File: # vi /etc/sysconfig/network-scripts/ifcfg-bond0
2. Add the following lines to the Bond0 Configuration File:

```
DEVICE=bond0
IPADDR=192.168.1.10
NETWORK=192.168.1.0
NETMASK=255.255.255.0
USERCTL=no
BOOTPROTO=none
ONBOOT=yes
BONDING_OPTS="mode=0 miimon=100"
Note: Replace IP address, Network and Netmask settings accordingly.
Note: Detailed description of the bonding options can be found in Red Hat's Deployment Guide.
```

3. Open the configuration file for eth0:

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth0
```

4. Edit eth0 configuration file adding the "MASTER" and "SLAVE" parameters:

```
DEVICE=eth0
USERCTL=no
ONBOOT=yes
MASTER=bond0
SLAVE=yes
BOOTPROTO=none
```

5. Repeat steps #3 and #4 for eth1.

6. Open the kernel modules configuration file:

```
RHEL5 # vi /etc/modprobe.conf
```

```
RHEL6 # vi /etc/modprobe.d/modprobe.conf
```

**Note:** modprobe.conf file does not exist on RHEL6. Following the step listed above, the file will be created.

7. Add the following line to modprobe.conf file:

```
alias bond0 bonding
options bond0 mode=balance-rr miimon=100
```

8. Load the bonding Module:

```
# modprobe bonding
```

9. Restart Network service:

```
# service network restart
```

10. Check if the bonding interface was created successfully looking at the output of the ifconfig command:

```
# ifconfig
```

The output should list bond0 up and running as master and eth0\eth1 up and running as slaves.

```
=====
```

**CentOS 8**

Install teamd Daemon - Responsible for creating a network team.

confirm the active network interfaces run:

```
$ nmcli device status
```

Gather UUID

```
$ nmcli connection show
```

Using their respective UUID's execute the commands below to delete the links:

```
$ nmcli connection delete e3cec54d-e791-4436-8c5f-4a48c134ad29  
$ nmcli connection delete dee76b4c-9alb-4f24-a9f0-2c9574747807
```

create a team interface called `team0`

```
$ nmcli connection add type team con-name team0 ifname team0 config '{"runner":  
{"name": "activebackup"}}'
```

To view the attributes assigned to the `team0` interface run the command:

```
$ nmcli connection show team0
```

```
$ nmcli connection show
```

Next, configure IP address for the `team0` interface as shown using the [nmcli command](#). Be sure to assign the IP's according to your network's subnet & IP addressing scheme.

```
$ nmcli con mod team0 ipv4.addresses 192.168.2.100/24  
$ nmcli con mod team0 ipv4.gateway 192.168.2.1  
$ nmcli con mod team0 ipv4.dns 8.8.8.8  
$ nmcli con mod team0 ipv4.method manual  
$ nmcli con mod team0 connection.autoconnect yes  
[tecmint@centos-8 ~]$ nmcli con mod nicbond0  ipv4.addresses 192.168.2.100/24  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$ nmcli con mod nicbond0  ipv4.gateway 192.168.2.1  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$ nmcli con mod nicbond0  ipv4.dns 8.8.8.8  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$ nmcli con mod nicbond0  ipv4.method manual  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$ nmcli con mod nicbond0  connection.autoconnect yes
```

Configure Team Network Interface

Thereafter, create slave links and associate the slaves to the team link:

```
$ nmcli con add type team-slave con-name team0-slave0 ifname enp0s3 master team0  
$ nmcli con add type team-slave con-name team0-slave1 ifname enp0s8 master team0  
[tecmint@centos-8 ~]$ nmcli con add type team-slave con-name team0-slave0 ifname enp0s3  
master team0  
Connection 'team0-slave0' (a6e5d9b6-3038-42d3-9377-6a503c9a9a16) successfully added.  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$  
[tecmint@centos-8 ~]$ nmcli con add type team-slave con-name team0-slave1 ifname enp0s8  
master team0  
Connection 'team0-slave1' (3e1669cb-cd8e-4a78-b4a8-8cd42561c6aa) successfully added.  
[tecmint@centos-8 ~]$
```

Configure Slave Network Interface

Check the status of the links again, and you'll notice that the slave links are now active.

```
$ nmcli connection show  
[tecmint@centos-8 ~]$ nmcli connection show  
NAME          UUID                TYPE      DEVICE  
team0         f64e3338-a33f-42a7-929a-6a9e619dcdb1  team      team0  
team0-slave0  a6e5d9b6-3038-42d3-9377-6a503c9a9a16  ethernet  enp0s3  
team0-slave1  3e1669cb-cd8e-4a78-b4a8-8cd42561c6aa  ethernet  enp0s8  
[tecmint@centos-8 ~]$
```

Check Team Network Interfaces

Next, deactivate and activate the team link. This activates the connection between the slave links and the team link.

```
$ nmcli connection down team0 && nmcli connection up team0
```

```
[tecmint@centos-8 ~]$ nmcli connection down team0 && nmcli connection up team0
Connection 'team0' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
Connection successfully activated (master waiting for slaves) (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/8)
[tecmint@centos-8 ~]$
```

Active Team Network Interfaces

Next, verify the state of the team link connection as shown.

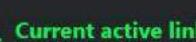
```
$ ip addr show dev team0
[tecmint@centos-8 ~]$ ip addr show dev team0
6: team0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    qlen 1000
        link/ether 08:00:27:c0:b6:8c brd ff:ff:ff:ff:ff:ff
        inet 192.168.2.100/24 brd 192.168.2.255 scope global noprefixroute team0
            valid_lft forever preferred_lft forever
            inet6 fe80::54d5:d941:923c:813/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
[tecmint@centos-8 ~]$
```

Verify Team Network Status

We can see that the link is up with the correct IP addressing that we configured earlier.

To retrieve additional details about the team link, run the command:

```
$ sudo teamdctl team0 state
[tecmint@centos-8 ~]$ sudo teamdctl team0 state
[sudo] password for tecmint:
setup:
runner: activebackup
ports:
  enp0s3
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
        down count: 0
  enp0s8
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
        down count: 0
runner:
  active port: enp0s3
[tecmint@centos-8 ~]$
```



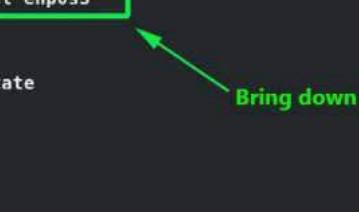
Check Team Network Info

From the output, we can see that both links (`enp0s3` and `enp0s8`) are up and that the active link is `enp0s8`.

## Step 3: Testing Network Teaming Redundancy

To test our active-backup teaming mode, we will disconnect the currently active link – `enp0s3` – and check whether the other link kicks in.

```
$ nmcli device disconnect enp0s3
$ sudo teamdctl team0 state
[tecmint@centos-8 ~]$ nmcli device disconnect enp0s3
Device 'enp0s3' successfully disconnected.
[tecmint@centos-8 ~]$ [tecmint@centos-8 ~]$ sudo teamdctl team0 state
setup:
runner: activebackup
ports:
  enp0s8
    link watches:
      link summary: up
      instance[link_watch_0]:
        name: ethtool
        link: up
        down count: 0
runner:
  active port: enp0s8
[tecmint@centos-8 ~]$
```



Testing Network Teaming

When you check the status of the teaming interface, you'll find that the link `enp0s8` has kicked in and serving connections to the server. This confirms that our setup is working!

## Step 4: Deleting a Network Teaming Interface

If you wish to delete the teaming interface/link and revert to default network settings, first bring down the teaming link:

```
$ nmcli connection down team0
```

Next, delete the slaves.

```
$ nmcli connection delete team0-slave0 team0-slave1
```

Finally, delete the teaming interface.

```
$ nmcli connection delete team0
```

```
[tecmint@centos-8 ~]$ nmcli connection down team0
Connection 'team0' successfully deactivated (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/8)
[tecmint@centos-8 ~]$
[tecmint@centos-8 ~]$
[tecmint@centos-8 ~]$ nmcli connection delete team0-slave0 team0-slave1
Connection 'team0-slave0' (af6e5d9b6-3038-42d3-9377-6a503c9a9a16) successfully deleted.
Connection 'team0-slave1' (3e1669cb-cd8e-4a78-b4a8-8cd42561c6aa) successfully deleted.
[tecmint@centos-8 ~]$
[tecmint@centos-8 ~]$
[tecmint@centos-8 ~]$ nmcli connection delete team0
Connection 'team0' (f64e3338-a33f-42a7-929a-6a9e619dcdb1) successfully deleted.
[tecmint@centos-8 ~]$
```

Delete Team Network Interfaces

AD

At this point, all the interfaces are down and your server is not reachable. To activate your network interfaces and regain connectivity, run the commands:

```
$ sudo ifconfig enp0s3 up
```

```
$ sudo ifconfig enp0s8 up
```

```
$ sudo systemctl restart NetworkManager
```

# Subnet

Wednesday, September 22, 2021

10:58 PM

## Subnetting

Prefix Length Slash Notation (CIDR)	Addresses (Total IPs)	Max Available Hosts (Usable IPs)	Subnet Length	Subnet Mask
/32	1	1	0	255.255.255.255
/31	2	0	1	255.255.255.254
/30	4	2	2	255.255.255.252
/29	8	6	3	255.255.255.248
/28	16	14	4	255.255.255.240
/27	32	30	5	255.255.255.224
/26	64	62	6	255.255.255.192
/25	128	126	7	255.255.255.128
/24	256	254	8	255.255.255.0
/23	512	510	9	255.255.254.0
/22	1024	1022	10	255.255.252.0
/21	2048	2046	11	255.255.248.0
/20	4096	4094	12	255.255.240.0
/19	8192	8190	13	255.255.224.0
/18	16384	16382	14	255.255.192.0
/17	32768	32766	15	255.255.128.0
/16	65536	65534	16	255.255.0.0
/15	131072	131070	17	255.254.0.0
/14	262144	262142	18	255.252.0.0
/13	524288	524286	19	255.248.0.0
/12	1048576	1048574	20	255.240.0.0
/11	2097152	2097150	21	255.224.0.0
/10	4194304	4194302	22	255.192.0.0
/9	8388608	8388606	23	255.128.0.0
/8	16777216	16777214	24	255.0.0.0
/7	33554432	33554430	25	254.0.0.0
/6	67108864	67108862	26	252.0.0.0
/5	134217728	134217726	27	248.0.0.0
/4	268435456	268435454	28	240.0.0.0
/3	536870912	536870910	29	224.0.0.0
/2	1073741824	1073741822	30	192.0.0.0
/1	2147483648	2147483646	31	128.0.0.0
/0	4294967296	4294967294	32	0.0.0.0

## Cheat Sheet Series

comparitech

Classful IPv4 Addresses	
Class A	0.0.0.0 – 127.255.255.255
Class B	128.0.0.0 – 191.255.255.255
Class C	192.0.0.0 – 223.255.255.255
Class D	224.0.0.0 – 239.255.255.255
Class E	240.0.0.0 – 255.255.255.255

Private IPv4 Addresses	
10.0.0.0 – 10.255.255.255	
172.16.0.0 – 172.31.255.255	
192.168.0.0 – 192.168.255.255	

Special IPv4 Addresses	
Local Host	127.0.0.0 – 127.255.255.255
APIPA	169.254.0.0 – 169.254.255.255

Bogon IPv4 Addresses	
This network	0.0.0.0/8
Private IPv4 Block	10.0.0.0/8
Carrier-grade NAT	100.64.0.0/10
Loopback	127.0.0.0/8
Name collision occurrence	127.0.53.53
Link local	169.254.0.0/16
Private IPv4 Block	172.16.0.0/12
IETF Assignments	192.0.0.0/24
TEST-NET-1	192.0.2.0/24
Private IPv4 Block	192.168.0.0/16
Benchmark testing	198.18.0.0/15
TEST-NET-2	198.51.100.0/24
TEST-NET-3	203.0.113.0/24
Multicast	224.0.0.0/4
Reserved	240.0.0.0/4
Limited broadcast	255.255.255.255/32

Creating a subnet by dividing the host identifier			
Before Subnetting	Network Identifier	Host Identifier	
			 
After Subnetting	Network Identifier	Subnet Identifier	Host Identifier

# Iptables

Wednesday, December 2, 2020 2:58 PM

## IPtables address under Linux

```
iptables -A INPUT -s IP-ADDRESS -j DROP
```

Replace IP-ADDRESS with your actual IP address. For example, if you wish to block an ip address 65.55.44.100 for

whatever reason then type the command as follows:

```
# iptables -A INPUT -s 65.55.44.100 -j DROP
```

If you have IP tables firewall script, add the above rule to your script.

If you just want to block access to one port from an ip 65.55.44.100 to port 25 then type command:

```
# iptables -A INPUT -s 65.55.44.100 -p tcp --destination-port 25 -j DROP
```

The above rule will drop all packets coming from IP 65.55.44.100 to port mail server port 25.

## **CentOS / RHEL / Fedora Block An IP And Save It To Config File**

Type the following two command:

```
# iptables -A INPUT -s 65.55.44.100 -j DROP  
# service iptables save
```

## **How Do I Unblock An IP Address?**

Use the following syntax (the -d options deletes the rule from table):

```
# iptables -D INPUT -s xx.xxx.xx.xx -j DROP  
# iptables -D INPUT -s 65.55.44.100 -j DROP  
# service iptables save
```

## **In Details.**

### **3. iptables - Implementing Packet Filtering**

#### 3.1 Understanding Packaeet Filtering

- Firewalling all handed by linux kernel
  - In kernel we have Net filter
  - Net filter is part of kernel taking care of firewall
  - One generic tool is doing that is iptables
  - Available on all linux distribution
  - not easy interface to create firewall rules
  - so have different name on different OS
- susefirewall - Suse  
firewalld - Redhat  
ufw - Ubuntu

### **3.2 Understanding iptables working**

- Core of IP table is Table
- Default table is Mangle table
- Other table also available such as NAT table

Inside table there is chain

- Forward chain (Forwarding packets - Router)
- Input Chain (it's for incoming packets)
- Output Chain (It's for out going packets)

Inside NAT Table

- Pre-Routing
- Post routing

On chain level you will define the rules

Most important chain is INPUT chain that is filtering out traffic that is getting into the server  
OUTPUT chain will decide only allowed packets will go out of the servers

```
# iptables -A INPUT -i eth0 -s 10.0.0.0/24 -p tcp --dport 80 -j ACCEPT
```

-A : Attending

INPUT: in the input chain

-i eth0: Filtering conditions

-p tcp: port

--dport: Destination port

-j : Jump the target

ACCEPT: Accept the packets

DROP:

REJECT:

LOG:

### **3.3 setting up a basic iptables configuration**

Policy:

You can set them using "iptables"

```
# iptables -P INPUT DROP
```

```
# iptables -L -v (default configuration output)
```

**if you dont want to allow anyone one server (disabling everything)**

```
# iptables -P INPUT DROP
```

```
# iptables -P OUTPUT DROP
```

```
# iptables -P FORWARD DROP
```

**Incoming traffic from loopback local accepted (Starting point)**

```
# iptables -A INPUT -i lo -j ACCEPT
```

```
# iptables -A OUTPUT -o lo -j ACCEPT
```

**Now Start enabling the iptables rules.**

**HTTP, HTTPS, SSH, UDP** port enabling coming in

```
# iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

```
# iptables -A INPUT -p tcp --dport 53 -j ACCEPT
```

```
# iptables -A INPUT -p udp --dport 53 -j ACCEPT
```

```
# iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

```
# iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
```

```
# iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

```
# iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
# iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT
```

```
# iptables -A OUTPUT -p tcp --dport 80 -j ACCEPT  
# iptables -A OUTPUT -p tcp --dport 443 -j ACCEPT  
# iptables -I OUTPUT 2 -p tcp --dport 53 -j ACCEPT  
# iptables -I OUTPUT 2 -p udp --dport 53 -j ACCEPT  
# iptables -A INPUT -p icmp -j ACCEPT  
# iptables -A OUTPUT -p icmp -j ACCEPT
```

### 3.4 making iptables configuration persistent

Save the IPTABLE permanent

```
# iptables-save  
# iptables-save > /etc/sysconfig/iptbales (Rules will load auto on reboot)  
systemctl enable --now iptables (Activate the service)
```

### 3.5 configuring iptables NAT

- NAT is part of iptable - NAT table
  - you can configure linux as NAT router
  - linux server will apply NAT post routing (server will make decision on routing)

### 3.6 Using Logging for iptables troubleshooting

### 3.7 configuration port forwarding in iptables

# Firewall Security 1

Wednesday, June 30, 2021 9:22 PM

## Configure Firewall Settings

- Install firewalld: `yum install firewalld`
- Start and enable firewalld:  
`systemctl start firewalld &&`  
`systemctl enable firewalld`
- View `firewall-cmd` options:  
`firewall-cmd -h | man firewall-cmd`
- List zones: `firewall-cmd --get-zones`  
`(--get-default-zone)`
- List everything added for or enabled in a zone:  
`firewall-cmd --list-all --zone=public`
- Add a service for a zone:  
`firewall-cmd --add-service=service`  
`(--permanent)`
- Add a port for a zone:  
`firewall-cmd --add-port=port/protocol`  
`(--permanent)`
- Reload firewall rules: `firewall-cmd --reload`

## Configure Key Based Authentication for SSH

- Generate public and private key pair: `ssh-keygen`
- Copy a public key to a remote server:  
`ssh-copy-id username@remote_host`
- Default public/private key location:  
`/home/username/.ssh/`

`firewall-cmd --state`

`firewall-cmd --get-default-zone`

`firewall-cmd --list-all`

`firewall-cmd --get-zones`

`firewall-cmd --zone=public --add-port=5000/tcp`

`firewall-cmd --zone=public --list-ports`

`firewall-cmd --zone=public --add-port=4990-4999/udp`

Sudo `firewall-cmd --zone=public --permanent --add-port=888/tcp`

```
Sudo firewall-cmd --zone=public --permanent --add-port=4990-4999/udp
```

```
Sudo firewall-cmd --zone=public --permanent --list-ports
```

```
firewall-cmd --reload
```

```
firewall-cmd --get-services
```

# SELinux Security 2

Wednesday, June 30, 2021 9:59 PM

## Working with SELinux

- View SELinux modes: `getenforce`
- Set mode to permissive or enforcing:  
`setenforce 0 | 1`
- List booleans: `getsebool -a`
- Turn booleans on or off:  
`setsebool boolean on | off`  
(`-P` for permanent)
- List SELinux contexts: `semanage fcontext -l`
- View context on files and process:  
`ls -Z | ps -axZ`
- Change SELinux context:  
`semanage fcontext -a -t context_type '/directory(/.*)?'`
- Restore default contexts:  
`restorecon -R /directory`
- View SELinux policy violations:  
`sealert -a /var/log/audit/audit.log`

# Set Timezone / NTP

Wednesday, July 14, 2021 10:16 PM

```
# timedatectl list-timezones  
# timedatectl set-timezone "Asia/Kuala_Lumpur"  
# timedatectl set-ntp yes
```

```
# dnf install chrony  
# systemctl status chronyd
```

```
# cat /etc/chrony.conf
```

```
# timedatectl  
Local time: Wed 2021-07-14 22:52:30 +08  
Universal time: Wed 2021-07-14 14:52:30 UTC  
RTC time: Wed 2021-07-14 14:52:29  
Time zone: Asia/Kuala_Lumpur (+08, +0800)  
System clock synchronized: yes  
NTP service: active  
RTC in local TZ: no
```

## Selinux add

Sunday, January 2, 2022 8:48 PM

1. Start with checking the port allocation and confirming the port you want to allow access to isn't already being used,  
`sudo semanage port -l | grep http_port_t`
2. Allow access to port  
`sudo semanage port -a -t http_port_t -p tcp 8090`
3. Check firewall ports passthrough  
`sudo firewall-cmd --list-all`
4. Add port (and make it permanent)  
`sudo firewall-cmd --zone=public --add-port=9443/tcp --permanent`
5. Reload firewall for the changes to take effect  
`sudo firewall-cmd --reload`

# NFS

Friday, 3 February 2023 1:28 PM

7. After you have verified that the fstab entry is working, export the new filesystem by using the NFS filesystem. Add the following line to /etc(exports:

```
# /mnt/xfs *(rw,async)
```

Use your favorite editor or enter the echo command like in previous steps:

```
# echo i/mnt/xfs *(rw,async)i >> /etc(exports
```

```
[root@nfsserver ~]# vi /etc(exports  
/nfsshare 192.168.0.101(rw,sync,no_root_squash)
```

```
[root@nfsclient ~]# mount -t nfs 192.168.0.100:/nfsshare /mnt/nfsshare
```

```
[root@nfsclient ~]# showmount -e 192.168.0.100  
Export list for 192.168.0.100:  
/nfsshare 192.168.0.101
```

Some more important commands for NFS.

1. showmount -e : Shows the available shares on your local machine
2. showmount -e <server-ip or hostname>: Lists the available shares at the remote server
3. showmount -d : Lists all the sub directories
4. exportfs -v : Displays a list of shares files and options on a server
5. exportfs -a : Exports all shares listed in /etc(exports, or given name
6. exportfs -u : Unexports all shares listed in /etc(exports, or given name
7. exportfs -r : Refresh the server's list after modifying /etc(exports

## What are the difference between NFS 3 and NFS 4?

In NFS 3 there is no security to protect the data, but in NFS 4 there is a kerberos security to protect the data. In NFS 3 there is no ACL permissions on the shared directory, but in NFS 4 there is an ACL permissions on the shared directory.

# ICMP block

Sunday, 5 February 2023 4:28 PM

Temp:

```
# echo "1" > /proc/sys/ne/ipv4/icmp_echo_ignore_all
```

Permanent

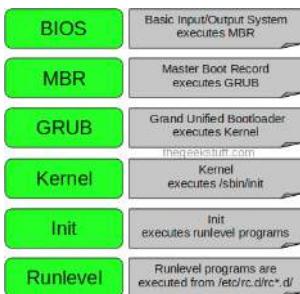
Edit sysctl.conf

```
net.ipv4.icmp_echo_ignore_all = 1
```

```
# sysctl -p
```

# Unix Interview

Friday, June 19, 2020 3:50 PM



## 1. BIOS

- BIOS stands for Basic Input/Output System
- Performs some system integrity checks
- Searches, loads, and executes the boot loader program.
- It looks for boot loader in floppy, cd-rom, or hard drive. You can press a key (typically F12 of F2, but it depends on your system) during the BIOS startup to change the boot sequence.
- Once the boot loader program is detected and loaded into the memory, BIOS gives the control to it.
- So, in simple terms BIOS loads and executes the MBR boot loader.

## 2. MBR

- MBR stands for Master Boot Record.
- It is located in the 1st sector of the bootable disk. Typically /dev/hda, or /dev/sda
- MBR is less than 512 bytes in size. This has three components 1) primary boot loader info in 1st 446 bytes 2) partition table info in next 64 bytes 3) mbr validation check in last 2 bytes.
- It contains information about GRUB (or LILO in old systems).
- So, in simple terms MBR loads and executes the GRUB boot loader.

## 3. GRUB

- GRUB stands for Grand Unified Bootloader.
- If you have multiple kernel images installed on your system, you can choose which one to be executed. ▪ GRUB displays a splash screen, waits for few seconds, if you don't enter anything, it loads the default kernel image as specified in the grub configuration file.
- GRUB has the knowledge of the filesystem (the older Linux loader LILO didn't understand filesystem).
- Grub configuration file is /boot/grub/grub.conf (/etc/grub.conf is a link to this). The following is sample grub.conf of CentOS.
- As you notice from the above info, it contains kernel and initrd image.
- So, in simple terms GRUB just loads and executes Kernel and initrd images.

## 4. Kernel

- Mounts the root file system as specified in the "root=" in grub.conf
- Kernel executes the /sbin/init program
- Since init was the 1st program to be executed by Linux Kernel, it has the process id (PID) of 1. Do a 'ps -ef | grep init' and check the pid.
  
- **initrd** stands for Initial RAM Disk.
- initrd is used by kernel as temporary root file system until kernel is booted and the real root file system is mounted. It also contains necessary drivers compiled inside, which helps it to access the hard drive partitions, and other hardware.

## 5. Init

- Looks at the /etc/inittab file to decide the Linux run level.
- Following are the available run levels
  - 0 – halt
  - 1 – Single user mode
  - 2 – Multiuser, without NFS
  - 3 – Full multiuser mode
  - 4 – unused
  - 5 – X11
  - 6 – reboot
- Init identifies the default initlevel from /etc/inittab and uses that to load all appropriate programs.
- Execute 'grep initdefault /etc/inittab' on your system to identify the default run level
- If you want to get into trouble, you can set the default run level to 0 or 6. Since you know what 0 and 6 means, probably you might not do that.
- Typically you would set the default run level to either 3 or 5.

## . Runlevel programs

- When the Linux system is booting up, you might see various services getting started. For example, it might say "starting sendmail .... OK". Those are the runlevel programs, executed from the run level directory as defined by your run level. ▪ Depending on your default init level setting, the system will execute the programs from one of the following directories.
  - Run level 0 – /etc/rc.d/rc0.d/
  - Run level 1 – /etc/rc.d/rc1.d/
  - Run level 2 – /etc/rc.d/rc2.d/
  - Run level 3 – /etc/rc.d/rc3.d/
  - Run level 4 – /etc/rc.d/rc4.d/
  - Run level 5 – /etc/rc.d/rc5.d/
  - Run level 6 – /etc/rc.d/rc6.d/
- Please note that there are also symbolic links available for these directory under /etc directly. So,

## Q1. Purpose of having different network ports

- Each service or each server that you run on next machine has to communicate over an IP and through port. Example DNS use port 53 NFS use 2049 on same server IP.

## Q2. What are the fields in /etc/passwd?

- User info saved in /etc/passwd file
- Group info saved in /etc/group file
- Password info saved in /etc/shadow file

First field always the user name.

Second field with x means this user has a password

Third Field **User ID**

Forth field **Group ID**

Fifth field User description

Sixth field Home directory

Seven Shell the user will be using

All the encryption goes to the shadow file.

## Q3. Explain cron job syntax

- To schedule Cron job
- # crontab -e

## Q4. Linux Boot Process

- The boot sequence changes in CentOS/Redhat 7 and above
- Systemd is the new service manager in CentOS/RHEL7 that manages the boot sequence.
- It is backward compatible with SysV init script used by previous version of redhat Linux including RHEL 6.

BIOS - Basic input and output settings (firmware interface) not related to OS

POST - Power-on-Self-test started (Check Hardware status)

MBR - Master Boot Record

Information saved in the first sector of hard disk that indicates where the GRUB2 is located so it can be loaded in computer RAM.

GRUB2 - Grand unified boot loader v2

Loads linux kernel  
/boot/grub2/grub.cfg

Kernel - Core of the OS

The Linux Kernel is a low-level systems software whose main role is to manage hardware resources for the user.  
Loads required drivers from initrd.img  
Starts the first OS process (systemd)

Systemd - System Daemon (PID # 1)

It then starts all the required processes  
Reads - /etc/systemd/system/default.target to bring the system to run level  
Total of 7 run-levels

## Q5. Explain the difference between RAID0, RAID1 and RAID5?

**RAID 0** and **RAID 1** are two types of configurations or levels that can be set up with an array of independent disks. **RAID 0** offers striping, which translates to better performance, but no-fault tolerance or data redundancy. **RAID 1**, on the other hand, offers mirroring, so the same data is available in two disks.

>

/etc/rc0.d is linked to /etc/rc.d/rc0.d.

- Under the /etc/rc.d/rc\*.d/ directories, you would see programs that start with S and K.
- Programs starts with S are used during startup. S for startup.
- Programs starts with K are used during shutdown. K for kill.

#### 4) What is Linux Kernel?

The Linux Kernel is a low-level systems software whose main role is to manage hardware resources for the user. It is also used to provide an interface for user-level interaction.

#### 5) What is LILO?

LILO is a boot loader for Linux. It is used mainly to load the Linux operating system into main memory so that it can begin its operations.

#### 1) What is Linux?

Linux is an operating system based on UNIX, and was first introduced by Linus Torvalds. It is based on the Linux Kernel, and can run on different hardware platforms manufactured by Intel, MIPS, HP, IBM, SPARC and Motorola. Another popular element in Linux is its mascot, a penguin figure named Tux.

#### 2) What is the difference between UNIX and LINUX?

Unix originally began as a propriety operating system from Bell Laboratories, which later on spawned into different commercial versions. On the other hand, Linux is free, open source and intended as a non-proprietary operating system for the masses.

#### 3) What is BASH?

BASH is short for Bourne Again SHELL. It was written by Steve Bourne as a replacement to the original Bourne Shell (represented by /bin/sh). It combines all the features from the original version of Bourne Shell, plus additional functions to make it easier and more convenient to use. It has since been adapted as the default shell for most systems running Linux.

#### 6) What is a swap space?

A swap space is a certain amount of space used by Linux to temporarily hold some programs that are running concurrently. This happens when RAM does not have enough memory to hold all programs that are executing.

#### 8 ) What are the basic components of Linux?

Just like any other typical operating system, Linux has all of these components: kernel, shells and GUIs, system utilities, and application programs. What makes Linux advantageous over other operating systems is that every aspect comes with additional features and all codes for these are downloadable for free.

#### 9) Does it help for a Linux system to have multiple desktop environments installed?

In general, one desktop environment, like KDE or Gnome, is good enough to operate without issues. It's all a matter of preference for the user, although the system allows switching from one environment to another. Some programs will work on one environment and not work on the other, so it could also be considered a factor in selecting which environment to use.

#### 1. How do you perform NIC teaming?

NIC Teaming/bonding is used mostly in scenarios where you cannot afford to lose connectivity due to ethernet failover issues and also it has many other advantages like to distribute bandwidth, fault tolerance etc.

#### 2. What is the difference between TCP and UDP protocol?

TCP is a connection-oriented protocol and contains the information of sender as well as receiver. Eg: HTTP, FTP, Telnet TCP is slower than UDP due to its error checking mechanism. UDP protocols are connectionless; packets have no information on where they are going. These type of ports are generally used for broadcasting. For eg: DNS, DHCP UDP are faster.

#### 3. What are the benefits of NIC Teaming?

Load balancing, Fault Tolerance, Failover

#### 4. Mention all the network configuration files you would check to configure your ethernet card Show/Hide Answer

Ans: /etc/sysconfig/network-scripts/ifcfg-eth\* /etc/sysconfig/network/etc/resolv.conf /etc/nsswitch.conf

#### 5. What is the use of /etc/resolv.conf? Show/Hide Answer

Ans: It contains the details of nameserver, i.e., details of your DNS server which helps us connect to the Internet.

#### 6. What is the use of /etc/hosts file? Show/Hide Answer

Ans: To map any hostname to its relevant IP

#### 7. What is the command to check all the open ports of your machine? Show/Hide Answer

Ans: nmap localhost

#### 8. What is the command to check all the listening ports and services of your machine?

Ans: netstat -ntlp

#### 10. What are the 6 run levels of Linux? And how can you configure your script to run only when the system boots into GUI and not to any other runlevel? Show/Hide Answer

Ans: 0 power off, 1 single user, 2 multi user, 3 multiuser with network, 4 development purpose, 5 GUI, 6 Restart, ch kconfig --level 5 service\_name on chkconfig --level 1234 service\_name off

#### 11. What is a 3-way handshake protocol? Give an example of it Show/Hide Answer

Ans: SYN - system 1 sends SYN signal to remote system. SYN-ACK - remote system receives the SYN signal and sends ACK signal. ACK - system again receives ACK signal from remote system and connection is established. For example: When you ping to a machine, you are sending a SYN signal which is ACK by the remote machine, then it sends a SYN-ACK signal back to the host machine. Then the host machine receives SYN-ACK and sends the ACK signal back to confirm the same.

#### 12. What are the possible ways to check if your system is listening to port 67? Show/Hide Answer

Ans: # nmap localhost | grep 67 # netstat -ntlp | grep 67

#### Q:1 Why LVM is required ?

Ans: LVM stands for Logical Volume Manager, to resize filesystem's size online we require LVM partition in Linux. Size of LVM partition can be extended and reduced using the lvextend & lvreduce commands respectively.

#### Q:2 How To check Memory stats and CPU stats ?

Ans: Using 'free' & 'vmstat' command we can display the physical and virtual memory statistics respectively. With the help of 'sar' command we see the CPU utilization & other stats.

#### Q:3 What does Sar provides and at which location Sar logs are stored ?

Ans: Sar Collect, report, or save system activity information. The default version of the sar command (CPU utilization report) might be one of the first facilities the user runs to begin system activity investigation, because it monitors major system resources. If CPU utilization is near 100 percent (user + nice + system), the workload sampled is CPU-bound.

By default log files of Sar command is located at /var/log/sa/sadd file, where the dd parameter indicates the current day.

#### **Q:4 How to increase the size of LVM partition ?**

Ans: Below are the Logical Steps : – Use the lvextend command (lvextend -L +100M /dev/<Name of the LVM Partition> , in this example we are extending the size by 100MB. – resize2fs /dev/<Name of the LVM Partition> – check the size of partition using 'df -h' command

#### **Q:5 How to reduce or shrink the size of LVM partition ?**

Ans: Below are the logical Steps to reduce size of LVM partition : -Unmount the filesystem using umount command, -use resize2fs command , e.g resize2fs /dev/mapper/myvg-my lv 10G -Now use the lvreduce command , e.g lvreduce -L 10G /dev/mapper/myvg-my lv  
Above Command will shrink the size & will make the filesystem size 10GB.

#### **Q:6 How to create partition from the raw disk ?**

Ans: Using fdisk utility we can create partitions from the raw disk.Below are the steps to create partition from the raw disk : – fdisk /dev/hd\* (IDE) or /dev/sd\* (SCSI) – Type n to create a new partition – After creating partition , type w command to write the changes to the partition table.

#### **Q:7 Where the kernel modules are located ?**

Ans: The '/lib/modules/kernel-version/' directory stores all kernel modules or compiled drivers in Linux operating system. Also with 'lsmod' command we can see all the installed kernel modules.

#### **Q:8 What is umask ?**

Ans: umask stands for 'User file creation mask', which determines the settings of a mask that controls which file permissions are set for files and directories when they are created.

#### **Q:9 How to set the umask permanently for a user ?**

Ans: To set this value permanently for a user, it has to be put in the appropriate profile file which depends on the default shell of the user.

#### **Q:10 How to change the default run level in linux ?**

Ans: To change the run level we have to edit the file "/etc/inittab" and change initdefault entry ( id:5:initdefault:). Using 'init' command we change the run level temporary like 'init 3' , this command will move the system in runlevel 3.

#### **Q:11 How to share a directory using nfs ?**

Ans: To share a directory using nfs , first edit the configuration file '/etc/exportfs' , add a entry like '</directory-name> <ip or Network>(Options)' and then restart the nfs service.

#### **Q:12 How to check and mount nfs share ?**

Ans: Using 'showmount' command we can see what directories are shared via nfs e.g 'showmount -e <ip address of nfs server>'.Using mount command we can mount the nfs share on linux machine.

#### **Q:13 What are the default ports used for SMTP,DNS,FTP,DHCP,SSH and squid ?**

Ans: Service Port SMTP 25 DNS 53 FTP 20 (data transfer) , 21 ( Connection established) DHCP 67/UDP(dhcp server) , 68/UDP(dhcp client) SSH 22 Squid 3128

#### **Q:14 What is Network Bonding ?**

Ans: Network bonding is the aggregation of multiple Lan cards into a single bonded interface to provide fault tolerance and high performance.  
Network bonding is also known as NIC Teaming.

#### **Q:15 What are the different modes of Network bonding in Linux ?**

Ans: Below are list of modes used in Network Bonding :  
balance-rr or 0 – round-robin mode for fault tolerance and load balancing. active-backup or 1 – Sets active-backup mode for fault tolerance. balance-xor or 2 – Sets an XOR (exclusive-or) mode for fault tolerance and load balancing. broadcast or 3 – Sets a broadcast mode for fault tolerance. All transmissions are sent on all slave interfaces. 802.3ad or 4 – Sets an IEEE 802.3ad dynamic link aggregation mode. Creates aggregation groups that share the same speed & duplex settings. balance-tlb or 5 – Sets a Transmit Load Balancing (TLB) mode for fault tolerance & load balancing. balance-alb or 6 – Sets an Active Load Balancing (ALB) mode for fault tolerance & load balancing.

#### **Q:16 How to check and verify the status the bond interface.**

Ans: Using the command '`cat /proc/net/bonding/bond0`' , we can check which mode is enabled and what lan cards are used in this bond. In this example we have one only one bond interface but we can have multiple bond interface like bond1,bond2 and so on.

#### **Q:17 How to check default route and routing table ?**

Ans: Using the Commands 'netstat -nr' and 'route -n' we can see the default route and routing tables.

#### **Q:18 How to check which ports are listening in my Linux Server ?**

Ans: Use the Command 'netstat -listen' and 'lsof -i'

#### **Q:19 List the services that are enabled at a particular run level in linux server ?**

Ans: With the help of command 'chkconfig --list | grep 5:on' we can list all the service that are enabled in run level5. For other run levels just replace 5 with the respective run level.

#### **Q:20 How to enable a service at a particular run level ?**

Ans: We can enable a service using the Command 'chkconfig <Service-Name> on --level 3'

#### **Q:21 How to upgrade Kernel in Linux ?**

Ans: We should never upgrade Linux Kernel , always install the new New kernel using rpm command because upgrading a kernel can make your linux box in a unbootable state.

#### **Q:22 How To scan newly assigned luns on linux box without rebooting ?**

Ans: There are two ways to scan newly assigned luns : Method:1 if sg3 rpm is installed , then run the command 'rescan -scsi-bus.sh' Method:2 Run the Command , `echo " --- > /sys/class/scsi_host/hostX/scan`

#### **Q:23 How to find WWN numbers of HBA cards in Linux Server ?**

Ans: We can find the WWN numbers of HBA cards using the command 'systool -c fc\_host -v | grep port\_name'

#### **Q:24 How to add & change the Kernel parameters ?**

Ans: To Set the kernel parameters in linux , first edit the file '/etc/sysctl.conf' after making the changes save the file and run the command 'sysctl -p' , this command will make the changes permanently without rebooting the machine.

#### **Q:25 What is Puppet Server ?**

Ans: Puppet is an open-source & enterprise software for configuration management toll in UNIX like operating system. Puppet is a IT automation software used to push configuration to its clients (puppet agents) using code. Puppet code can do a variety of tasks from installing new software, to check file permissions, or updating user accounts & lots of other tasks.

#### **Q:26 What are manifests in Puppet ?**

Ans: Manifests in Puppet are the files in which the client configuration is specified.

#### **Q:27 Which Command is used to sign requested certificates in Puppet Server ?**

Ans: 'puppetca --sign hostname-of-agent' in (2.X) & 'puppet ca sign hostname-of-agent' in (3.X)

Q:28 At which location Puppet Master Stores Certificates ?

Ans: /var/lib/puppet/ssl/ca/signed

Q:29 How to find all the regular files in a directory ?

Ans: using the command 'find /<directory> -type f'.

**Q:30 What is load average in a linux ?**

Ans: Load Average is defined as the average sum of the number of process waiting in the run queue and number of process currently executing over the period of 1,5 and 15 minutes. Using the 'top' and 'uptime' command we find the load average of a linux sever.

10) What is the basic difference between BASH and DOS?

The key differences between the BASH and DOS console lies in 3 areas: – BASH commands are case sensitive while DOS commands are not; – under BASH, / character is a directory separator and \ acts as an escape character. Under DOS, / serves as a command argument delimiter and \ is the directory separator – DOS follows a convention in naming files, which is 8 character file name followed by a dot and 3 character for the extension. BASH follows no such convention.

11) What is the importance of the GNU project?

This so-called Free software movement allows several advantages, such as the freedom to run programs for any purpose and freedom to study and modify a program to your needs. It also allows you to redistribute copies of a software to other people, as well as freedom to improve software and have it released to the public.

12) Describe the root account.

The root account is like a systems administrator account, and allows you full control of the system. Here you can create and maintain user accounts, assigning different permissions for each account. It is the default account every time you install Linux.

13) What is CLI?

CLI is short for Command Line Interface. This interface allows user to type declarative commands to instruct the computer to perform operations. CLI offers an advantage in that there is greater flexibility. However, other users who are already accustomed with using GUI find it difficult to remember commands including attributes that come with it.

14) What is GUI?

GUI, or Graphical User Interface, makes use of images and icons that users click and manipulate as a way of communicating with the computer. Instead of having to remember and type commands, the use of graphical elements makes it easier to interact with the system, as well as adding more attraction through images, icons and colors.

15) How do you open a command prompt when issuing a command?

To open the default shell (which is where the command prompt can be found), press Ctrl+Alt+F1. This will provide a command line interface (CLI) from which you can run commands as needed.

16) How can you find out how much memory Linux is using?

From a command shell, use the "concatenate" command: cat /proc/meminfo for memory usage information. You should see a line starting something like: Mem: 64655360, etc. This is the total memory Linux thinks it has available to use.

17) What is typical size for a swap partition under a Linux system?

The preferred size for a swap partition is twice the amount of physical memory available on the system. If this is not possible, then the minimum size should be the same as the amount of memory installed.

18) What are symbolic links?

Symbolic links act similarly to shortcuts in Windows. Such links point to programs, files or directories. It also allows you instant access to it without having to go directly to the entire pathname.

19) Does the Ctrl+Alt+Del key combination work on Linux?

Yes, it does. Just like Windows, you can use this key combination to perform a system restart. One difference is that you won't be getting any confirmation message and therefore, reboot is immediate.

20) How do you refer to the parallel port where devices such as printers are connected?

Whereas under Windows you refer to the parallel port as the LPT port, under Linux you refer to it as /dev/lp. LPT1, LPT2 and LPT3 would therefore be referred to as /dev/lp0, /dev/lp1, or /dev/lp2 under Linux.

21) Are drives such as harddrive and floppy drives represented with drive letters?

No. In Linux, each drive and device has different designations. For example, floppy drives are referred to as /dev/fd0 and /dev/fd1. IDE/EIDE hard drives are referred to as /dev/hda, /dev/hdb, /dev/hdc, and so forth.

22) How do you change permissions under Linux?

Assuming you are the system administrator or the owner of a file or directory, you can grant permission using the chmod command. Use + symbol to add permission or - symbol to deny permission, along with any of the following letters: u (user), g (group), o (others), a (all), r (read), w (write) and x (execute). For example the command chmod go+rwx FILE1.TXT grants read and write access to the file FILE1.TXT, which is assigned to groups and others.

23) In Linux, what names are assigned to the different serial ports?

Serial ports are identified as /dev/ttyS0 to /dev/ttyS7. These are the equivalent names of COM1 to COM8 in Windows.

24) How do you access partitions under Linux?

Linux assigns numbers at the end of the drive identifier. For example, if the first IDE hard drive had three primary partitions, they would be named/numbered, /dev/hda1, /dev/hda2 and /dev/hda3.

25) What are hard links?

Hard links point directly to the physical file on disk, and not on the path name. This means that if you rename or move the original file, the link will not break, since the link is for the file itself, not the path where the file is located.

26) What is the maximum length for a filename under Linux?

Any filename can have a maximum of 255 characters. This limit does not include the path name, so therefore the entire pathname and filename could well exceed 255 characters.

27) What are filenames that are preceded by a dot?

In general, filenames that are preceded by a dot are hidden files. These files can be configuration files that hold important data or setup info. Setting these files as hidden makes it less likely to be accidentally deleted.

28) Explain virtual desktop.

This serves as an alternative to minimizing and maximizing different windows on the current desktop. Using virtual desktops, each desktop is a clean slate where you can open one or more programs. Rather than minimizing/restoring all those programs as needed, you can simply shuffle between virtual desktops with programs intact in each one.

29) How do you share a program across different virtual desktops under Linux?

To share a program across different virtual desktops, in the upper left-hand corner of a program window look for an icon that looks like a pushpin. Pressing this button will "pin" that application in place, making it appear in all virtual desktops, in the same position on screen.

30) What does a nameless (empty) directory represent?

This empty directory name serves as the nameless base of the Linux file system. This serves as an attachment for all other directories, files, drives and devices.

31) What is the pwd command?

The pwd command is short for print working directory command. Its counterpart in DOS is the cd command, and is used to display the current location in the directory tree.

32) What are daemons?

Daemons are services that provide several functions that may not be available under the base operating system. Its main task is to listen for service request and at the same time to act on these requests. After the service is done, it is then disconnected and waits for further requests.

33) How do you switch from one desktop environment to another, such as switching from KDE to Gnome?

Assuming you have these two environments installed, just log out from the graphical interface. Then at the Log in screen, type your login ID and password and choose which session type you wish to load. This choice will remain your default until you change it to something else.

34) What are the kinds of permissions under Linux?

There are 3 kinds of permissions under Linux: – Read: users may read the files or list the directory – Write: users may write to the file or new files to the directory – Execute: users may run the file or lookup a specific file within a directory

35) How does case sensitivity affect the way you use commands?

When we talk about case sensitivity, commands are considered identical only if every character is encoded as is, including lowercase and uppercase letters. This means that CD, cd and Cd are three different commands. Entering a command using uppercase letters, where it should be in lowercase, will produce different outputs.

36) What are environmental variables?

Environmental variables are global settings that control the shell's function as well as that of other Linux programs. Another common term for environmental variables is global shell variables.

37) What are the different modes when using vi editor?

There are 3 modes under vi: – Command mode – this is the mode where you start in – Edit mode – this is the mode that allows you to do text editing – Ex mode – this is the mode wherein you interact with vi with instructions to process a file

38) Is it possible to use shortcut for a long pathname?

Yes, there is. A feature known as filename expansion allows you to do this using the TAB key. For example, if you have a path named /home/iceman/assignments directory, you would type as follows: /ho[tab]/ice[tab]/assi[tab]. This, however, assumes that the path is unique, and that the shell you're using supports this feature.

39) What is redirection?

Redirection is the process of directing data from one output to another. It can also be used to direct an output as an input to another process.

40) What is grep command?

grep a search command that makes use of pattern-based searching. It makes use of options and parameters that are specified along the command line and applies this pattern into searching the required file output.

41) What could possibly be the problem when a command that was issued gave a different result from the last time it was used?

One highly possible reason for getting different results from what seems to be the same command has something to do with case sensitivity issues. Since Linux is case sensitive, a command that was previously used might have been entered in a different format from the present one. For example, to lists all files in the directory, you should type the command ls, and not LS. Typing LS would either result in an error message if there is no program by that exact name exist, or may produce a different output if there is a program named LS that performs another function.

42) What are the contents in /usr/local?

It contains locally installed files. This directory actually matters in environments where files are stored on the network. Specifically, locally-installed files go to /usr/local/bin, /usr/local/lib, etc.). Another application of this directory is that it is used for software packages installed from source, or software not officially shipped with the distribution.

43) How do you terminate an ongoing process?

Every process in the system is identified by a unique process id or pid. Use the kill command followed by the pid in order to terminate that process. To terminate all process at once, use kill 0.

44) How do you insert comments in the command line prompt?

Comments are created by typing the # symbol before the actual comment text. This tells the shell to completely ignore what follows. For example: "# This is just a comment that the shell will ignore."

45) What is command grouping and how does it work?

You can use parentheses to group commands. For example, if you want to send the current date and time along with the contents of a file named OUTPUT to a second file named MYDATES, you can apply command grouping as follows: (date cat OUTPUT) > MYDATES

# RHEL 7 Boot Process

Friday, June 19, 2020 3:49 PM

## Introduction of systemd

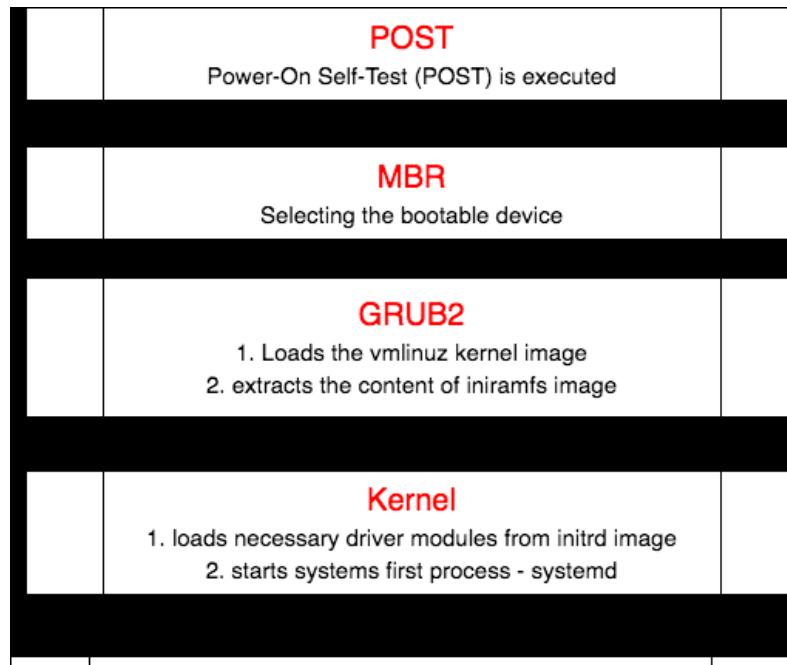
systemd is the new system and service manager in CentOS/RHEL 7.

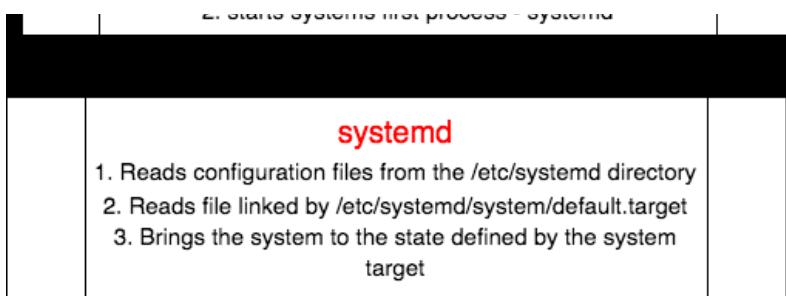
It is backward compatible with SysV init scripts used by previous versions of RedHat Linux including RHEL 6.

It replaces Upstart as the default initialization system.

## The following steps summarize how the boot procedure happens in RHEL/CentOS 7.

1. The computer's BIOS performs POST. (Firmware interface - check the system)
2. BIOS reads the MBR for the bootloader. (Info saved in first sector of hard disk that indicates where the GRUB2 is located so it can be loaded in computer RAM)
3. GRUB 2 bootloader loads the vmlinuz kernel image. (Grand Unified Boot Loader v2, Loads Linux kernel. /boot/grub2/grub.cfg.)
4. GRUB 2 extracts the contents of the initramfs image.
5. The kernel loads driver modules from initramfs. (Core of Operating System. Loads required drivers from initrd.img Starts the first OS process (systemd)).
6. Kernel starts the system's first process, systemd.
7. The systemd process takes over. It: System Daemon (PID # 1)  
It start all the required processes. Read /etc/systemd/system/default.target to bring the system to run-level.  
Total of 7 run-levels (0 thru 6)
  - Reads configuration files from the /etc/systemd directory
  - Reads file linked by /etc/systemd/system/default.target
  - Brings the system to the



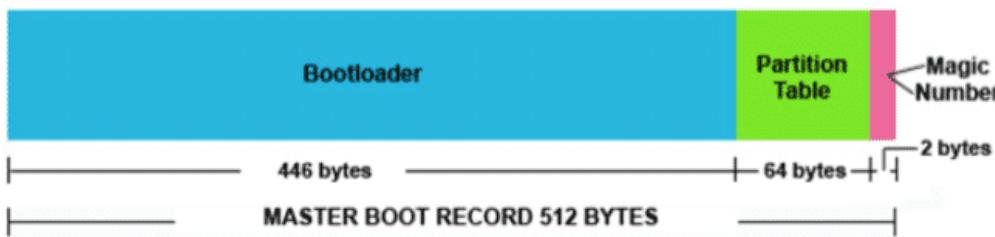


### **1. POST (Power on Self Test)**

From the system firmware, which can be the modern Universal Extended Firmware Interface (UEFI) or the classical Basic Input Output System (BIOS), the Power-On Self-Test (POST) is executed, and the hardware that is required to start the system is initialized.

### **2. Selecting the bootable device (With MBR)**

– Master Boot Record (MBR) is the first 512 bytes of the boot drive that is read into memory by the BIOS. – The next 64 bytes contain the partition table for the disk. The last two bytes are the “Magic Number” which is used for error detection.



### **3. Loading the boot loader (GRUB2)**

– The default bootloader program used on RHEL 7 is GRUB 2. GRUB stands for GRand Unified Bootloader. GRUB 2 replaces the older GRUB bootloader also called as legacy GRUB. – The GRUB 2 configuration file is located at /boot/grub2/grub.cfg (Do not edit this file directly).

```

### BEGIN /etc/grub.d/10_linux ####
menuentry 'CentOS Linux (3.10.0-693.21.1.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.8.1.el7.x86_64-advanced-0f790447-ebef-4ca0-b229-d0aa1985d57f' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ $feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint='hd0,msdos1' 0f790447-ebef-4ca0-b229-d0aa1985d57f
    else
        search --no-floppy --fs-uuid --set=root 0f790447-ebef-4ca0-b229-d0aa1985d57f
    fi
    linux16 /boot/vmlinuz-3.10.0-693.21.1.el7.x86_64 root=UUID=0f790447-ebef-4ca0-b229-d0aa1985d57f ro console=ttyS0,115200 console=tty0 console=ttyS0,115200n8 vconsole.font=latarcyrheb-sun16 crashkernel=auto vconsole.keymap=us LANG=en_US.UTF-8
        initrd16 /boot/initramfs-3.10.0-693.21.1.el7.x86_64.img
}

```

- GRUB 2 menu-configuration settings are taken from /etc/default/grub when generating grub.cfg.– Sample /etc/default/grub file :

```
# cat /etc/default/grub
GRUB_TIMEOUT=5 GRUB_DEFAULT=saved GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console" GRUB_CMDLINE_LINUX="rd.lvm.lv=rhel/swap crashkernel=auto
rd.lvm.lv=rhel/root rhgb quiet net.ifnames=0" GRUB_DISABLE_RECOVERY="true"
```

- If changes are made to any of these parameters, you need to run grub2-mkconfig to re-generate the /boot/grub2/grub.cfg file. # grub2-mkconfig –o /boot/grub2/grub.cfg
- GRUB2 searches the compressed kernel image file also called as vmlinuz in the /boot directory.– GRUB2 loads the vmlinuz kernel image file into memory and extracts the contents of the initramfs image file into a temporary, memory-based file system (tmpfs). – The initial RAM disk (initrd) is an initial root file system that is mounted before the real root file system.

### **initramfs**

- The job of the initial RAM file system is to preload the block device modules, such as for IDE, SCSI, or RAID, so that the root file system, on which those modules normally reside, can then be accessed and mounted.– The initramfs is bound to the kernel and the kernel mounts this initramfs as part of a two-stage boot process.– The dracut utility creates initramfs whenever a new kernel is installed.– Use the lsinitrd command to view the contents of the image created by dracut: # lsinitrd | less

### **4. Loading the kernel**

- The kernel starts the systemd process with a process ID of 1 (PID 1).
- It also loads the necessary driver modules from initrd image.– The boot loader (GRUB2) may present a boot menu to the user, or can be configured to automatically start a default operating system.– To load Linux, the kernel is loaded together with the initramfs. The initramfs contains kernel modules for all hardware that is required to boot, as well as the initial scripts required to proceed to the next stage of booting.– On RHEL 7, the initramfs contains a complete operational system (which may be used for troubleshooting purposes).

### **5. Starting systemd**

- The kernel starts the systemd process with a process ID of 1 (PID 1). root 1 0 0 02:10 ? 00:00:02 /usr/lib/systemd/systemd --switched-root --system --deserialize 23
- systemd is the first process that starts after the system boots, and is the final process that is running when the system shuts down.– It controls the final stages of booting and prepares the system for use. It also speeds up booting by loading services concurrently.– systemd reads the file linked by

`/etc/systemd/system/default.target` (for example, `/usr/lib/systemd/system/multi-user.target`) to determine the default system target (equivalent to run level). The system target file defines the services that systemd starts. – systemd allows you to manage various types of units on a system, including services (`name.service`) and targets (`name.target`), devices (`name.device`), file system mount points (`name.mount`), and sockets (`name.socket`).

**systemd brings the system to the state defined by the system target, performing system initialization tasks such as:**

1. Setting the host name
2. Initializing the network
3. Initializing SELinux based on its configuration
4. Printing a welcome banner
5. Initializing the system hardware based on kernel boot arguments
6. Mounting the file systems, including virtual file systems such as the `/proc` file system
7. Cleaning up directories in `/var`
8. Starting swapping

### **View default/current target unit**

Use the following command to view which target unit is used by default:

```
# systemctl get-default  
graphical.target
```

The `graphical.target` target unit indicates that the system is running in a graphical, multi-user state. This is similar to run level 5 in a SysV init system. You can verify this using the old command `runlevel :# runlevel`

```
N 5
```

The default target unit is represented by the `/etc/systemd/system/default.target` file. This file is a symbolic link to the current default target unit. For example :

```
# ls -lrt /etc/systemd/system/default.target  
lrwxrwxrwx. 1 root root 36 Sep 23 20:01 /etc/systemd/system/default.target ->  
/lib/systemd/system/graphical.target
```

### **Change default target unit**

Use the following command to change the default target unit (for example, to change the default to the `multi-user.target` unit):

```
# systemctl set-default multi-user.target  
Removed symlink /etc/systemd/system/default.target.  
Created symlink from /etc/systemd/system/default.target to /usr/lib/systemd/system/multi-user.target.
```

**Notice that the `default.target` symbolic link has changed, and is now pointing to the `multi-user.target` unit:**

```
# ls -lrt /etc/systemd/system/default.target  
lrwxrwxrwx. 1 root root 41 Sep 24 11:58 /etc/systemd/system/default.target ->  
/usr/lib/systemd/system/multi-user.target
```

Below table summarizes where a specific phase is configured and what you can do to troubleshoot if

things go wrong.

Boot Phase	Configuration
POST	Hardware Configuration (F2, ESC, F10 or another key)
Select bootable Device	BIOS/UEFI configuration or hardware boot menu
Loading the boot loader	grub2-install and edits to /etc/default/grub
Loading the kernel	Edits to the GRUB configuration and /etc/dracut.conf
starting /sbin/init	Compiled into initramfs
Processing initrd.target	Compiled into initramfs
Switch to the root filesystem	/etc/fstab
Running the default target	/etc/systemd/system/default.target

# From Tareek

Tuesday, December 15, 2020 1:39 PM

## 1- What are Vi, Ex, and Vim?

Ex stands for EXTENDED. It is a line editor. The need for the ex-command line came from the early days where computers were using printing terminals instead of screens. People were working with line numbers and editing individual lines as needed.

Vim stands for VI IMPROVED. It is an implementation of the Vi standard with many additions. It is the most commonly used implementation of the standard. Most Linux distributions come with Vim already installed.

Vi stands for Visual. It is a text editor that is an early attempt to a visual text editor.

## 2.What is the difference between TCP and UDP protocol?

## 3. What are the benefits of NIC Teaming?

## 4 How can you make a service run automatically after boot?

systemctl enable

## 5 How to lock an user account?

usermod -L

## 6 How to check if an user account has been locked?

Run the command "passwd -S <UserName>", this would show if the password has been locked or not. Otherwise, grep for the username from /etc/shadow file and you could see "!" mark prefixed to the encrypted password field.

## 7 What command to find memory and swap usage?

free

## 8 How to list and mount devices in Linux?

```
df -AT  
findmnt  
cat /proc/self/mounts
```

## 9 How to backup or archive files in Linux

```
tar -cvf tarball_name.tar /path/to/directory
```

## 10 How to set ownership for files/directories?

```
chown chown user:filename
```

## 11 Synchronize Time on Installed Linux

Run the ntpdate -u <ntpserver>

## 4. How do you find which processes are using a particular file?

By using lsof command in UNIX. It will list down PID of all the processes which are using a particular file.

## 5. How do you find which remote hosts are connecting to your host on a particular port say 10123?

By using netstat command

For example: execute netstat -a | grep "port" and it will list the entire hosts which are connected to this host on port 10123.

## 6. If one process is inserting data into your MySQL database? How will you check how many rows inserted into every second?

By using "watch" command in UNIX

## 7. There is a file Unix\_Test.txt which contains words "Unix". How will you replace all Unix to UNIX?

By using SED command in UNIX

For example: you can execute sed s/Unix/UNIX/g fileName.

## 8. How to check if the last command was successful in Unix?

To check the status of last executed command in UNIX, you can check the value of an inbuilt bash variable [\$.]. See the below example:  
\$> echo \$?

## 9. How to check all the running processes in Unix?

```
ps -ef  
ps aux
```

## 10. What's a PTR in DNS?

Pointer (PTR) record is used for reverse DNS (Domain Name System) lookup

## 39. Please explain how to enable curl on Ubuntu LAMP stack and root logging in Ubuntu?

Answer: To enable curl on Ubuntu LAMP stack:

Install libcurl

Use the command: sudo/etc/init.d/apache2 restart OR sudo service apache2 restart

To enable root logging in Ubuntu, use the command:

## 40. what is the size of a Swap Partition in LINUX

Double of Ram

## 35. What are the full forms of FCM and GCM in Firebase?

FCM and GCM stands for **Firebase Cloud Messaging** and **Google Cloud Messaging** respectively.

## 36. Few Difference between cPanel & Plesk

As we said before, Plesk runs on both Linux and Windows Server, while cPanel is a Linux-only deal.

## 37.How will you check out how much memory Linux is using

```
cat /proc/meminfo
```

## 38. What do you understand by daemons?

Daemons are a way of extending the functionality of the base operating system. In other words, daemons are services that

## 11. Which language is used in Git?

Git is written in C. It is very fast and reduces runtimes overhead.

## 12. What is SubGit?

SubGit is an open-source, version control tool for migrating Subversions (SVN) to Git. It allows creating a writable Git mirror of a Subversion repository, which can then be used to push to Git.

## 13. What is Jenkins?

Jenkins is an open-source continuous integration server that facilitates achieving a Continuous Integration process in an automated manner. It is also capable to highlight any errors in the project in its early stages.

## 14. What is a Virtual Private Cloud?

Virtual Private Cloud offers the flexibility to scale and control the workloads. It provides global access to manage workloads when you connect your on-premises or remote resources to Google Cloud Platform (GCP).

## 15. What is Chef?

Chef comprises of –

Chef Server – Also regarded as the central store for configuration data of infrastructure, and allows to dynamically drive node configuration based on data.

Chef Node – These are referred to as clients as they run the Chef-client software.

Chef Workstation – It is the place where users can interact with the Chef, develop and test cookbooks and recipes, and modify other configuration data.

## 16. Tell me what is kubectl.

It is a command-line interface to run commands against Kubernetes clusters, deploy applications, manage cluster resources, and view logs.

## 17. How will you restrict communication between Kubernetes Pods?

Communication between Kubernetes Pods is depending on the Container Network Interface (CNI) network plugin we are using. If it supports the Kubernetes network policy API, Kubernetes allows specifying network policies that restrict network access. Communication can be restricted based on IP addresses, ports, and/or selectors, a Kubernetes-specific feature for connecting and associating rules or components.

## 18. Why should you prefer Containerization to Virtualization?

Here why we should prefer Containerization to Virtualization –

Containers ensure real-time provisioning and scalability, whereas Virtual Machines (VMs) provide slow provisioning

Containers are lightweight than VMs

Containers display superior performance than VMs

Containers ensure better resource utilization than VMs

## 19. What is the Docker hub?

Docker Hub is a cloud-based repository of Docker. It allows users to create, test, store, and distribute container images.

It helps to –

Access public, open-source image repositories,

Use space to create their own private repositories

Build automated build functions, webhooks, and workgroups

Store manually pushed images and links to Docker cloud

## 20. What does the following command do with respect to the Amazon EC2 security groups

```
ec2-create-group CreateSecurityGroup
```

## 21. What is AMI?

AMI stands for Amazon Machine Image.

## 22. How many buckets can you create in AWS by default?

By default, you can create up to 100 buckets in each of your AWS accounts.

## 23. In VPC with private and public subnets, database servers should ideally be launched into which subnet?

With private and public subnets in VPC, database servers should ideally launch into private subnets.

## 24. What is a redshift?

Redshift is a big data warehouse product. It is fast and powerful, fully managed data warehouse service in the cloud.

## 25. What is AWS Lambda?

Lambda is an Amazon compute service which allows you to run code in the AWS Cloud without managing servers.

## 26. What are the different types of Load Balancer in AWS services?

Two types of Load balancer are:

Application Load Balancer

Classic Load Balancer

## 27. Name some of the DB engines which can be used in AWS RDS

MS-SQL DB

MariaDB

MySQL DB

OracleDB

PostgreSQL

## 28. What are the two most common ports used on XenApp ICA sessions? What are each used for?

A8 - Ports 1494 (ICA/HDX) and 2598(Session Reliability)

## 29. What free tool from Citrix will allow you to analyze your log files, profile your Citrix environment, scan for known

issues and attach a log file to a Citrix support ticket.

A15 - Citrix Insight Services or TaaS

## 30. This command temporarily stores all the modified tracked files.

```
git stash save
```

## 31. What are some differences between BSON documents used in MongoDB and JSON documents in general?

JSON (JavaScript Object Notation)—like XML, for example—is a human-readable standard used for data exchange. JSON has become the most widely used standard for data exchange on the web. JSON supports data types like booleans, numbers, strings, and arrays.

BSON, however, is the binary encoding that MongoDB uses to store its documents. It is similar to JSON, but it extends JSON to support more data types, like Date. BSON documents, unlike JSON documents, are ordered. BSON usually takes

**37. How will you check out how much memory Linux is using**  
cat /proc/meminfo

**38. What do you understand by daemons?**

Daemons are a way of extending the functionality of the base operating system. In other words, daemons are services that offer several functions that might not be available in the operating system. The main task of a daemon is to actively listen for a service request and to act upon them at the very same time. Once it completes the service, a daemon gets disconnected and waits for further requests.

has become the most widely used standard for data exchange on the web. JSON supports data types like booleans, numbers, strings, and arrays.

BSON, however, is the binary encoding that MongoDB uses to store its documents. It is similar to JSON, but it extends JSON to support more data types, like Date. BSON documents, unlike JSON documents, are ordered. BSON usually takes less space than JSON and is faster to traverse. BSON, since it is binary, is also quicker to encode and decode.

**32. What are the features of Firebase?**

Hosting  
Authentication  
Real-time Database

**33. What is Firebase?**

Firebase is a platform which is used for building Web, IOS and Android applications.

It offers:

Real time database  
Different APIs  
Multiple authentication types  
and Hosting platform

**34. What are the different types of event types for reading data in Firebase?**

value  
child\_added  
child\_changed  
child\_removed

# Samba

Saturday, August 28, 2021 1:03 PM

Samba is a free and open-source re-implementation of the [SMB/CIFS network file sharing protocol](#) that allows end users to access files, printers, and other shared resources.

In this tutorial, we will show how to install Samba on CentOS 7 and configure it as a standalone server to provide file sharing across different operating systems over a network.

We'll create the following Samba shares and users.

Users:

- **sadmin** - An administrative user with read and write access to all shares.
- **josh** - A regular user with its own private file share.

Shares:

- **users** - This share will be accessible with read/write permissions by all users.
- **josh** - This share will be accessible with read/write permissions only by users josh and sadmin.

The file shares will be accessible from all devices on your network. Later in the tutorial, we will also provide detailed instructions on how to connect to the Samba server from Linux, Windows and macOS clients.

## Prerequisites

Before you begin, make sure you are logged in to your CentOS 7 system as a [user with sudo privileges](#).

## Installing Samba on CentOS

Samba is available from the standard CentOS repositories. To install it on your CentOS system run the following command:

```
sudo yum install samba samba-client
```

Once the installation is completed, start the Samba services and enable them to start automatically on system boot:

```
sudo systemctl start smb.servicesudo systemctl start nmb.service
```

```
sudo systemctl enable smb.servicesudo systemctl enable nmb.service
```

The smbd service provides file sharing and printing services and listens on TCP ports 139 and 445. The nmbd service provides NetBIOS over IP naming services to clients and listens on UDP port 137.

## Configuring Firewall

Now that Samba is installed and running on your CentOS machine, you'll need to [configure your firewall](#) and open the necessary ports. To do so, run the following commands:

```
firewall-cmd --permanent --zone=public --add-service=sambafirewall-cmd --zone=public --add-service=samba
```

## Creating Samba Users and Directory Structure

For easier maintainability and flexibility instead of using the standard home directories (/home/user) all Samba directories and data will be located in the /samba directory.

Start by creating the /samba directory:

```
sudo mkdir /samba
```

[Create a new group](#) named sambashare. Later we will add all Samba users to this group.

```
sudo groupadd sambashare
```

Set the /samba directory [group ownership](#) to sambashare:

```
sudo chgrp sambashare /samba
```

Samba uses Linux users and group permission system but it has its own authentication mechanism separate from the standard Linux authentication. We will create the users using the standard Linux useradd tool and then set the user password with the smbpasswd utility.

As we mentioned in the introduction, we'll [create a regular user](#) that will have access to its private file share and one administrative account with read and write access to all shares on the Samba server.

## Creating Samba Users

To create a new user named josh, use the following command:

```
sudo useradd -M -d /samba/josh -s /usr/sbin/nologin -G sambashare josh
```

The useradd options have the following meanings:

- -M -do not create the user's home directory. We'll manually create this directory.
- -d /samba/josh - set the user's home directory to /samba/josh.
- -s /usr/sbin/nologin - disable shell access for this user.
- -G sambashare - add the user to the sambashare group.

[Create the user's home directory](#) and set the directory ownership to user josh and group sambashare:

```
sudo mkdir /samba/joshsudo chown josh:sambashare /samba/josh
```

The following command will add the setgid bit to the /samba/josh directory so the newly created files in this directory will inherit the group of the parent directory. This way, no matter which user creates a new file, the file will have group-owner of sambashare. For example, if you don't set the [directory's permissions](#) to 2770 and the sadmin user creates a new file the user josh will not be able to read/write to this file.

```
sudo chmod 2770 /samba/josh
```

Add the josh user account to the Samba database by setting the user password:

```
sudo smbpasswd -a josh
```

You will be prompted to enter and confirm the user password.

New SMB password:

Retype new SMB password:

Added user josh.

[Copy](#)

Once the password is set, enable the Samba account by typing:

```
sudo smbpasswd -e josh
```

Enabled user josh.

[Copy](#)

To create another user repeat the same process as when creating the user josh.

Next, let's create a user and group sadmin. All members of this group will have administrative permissions. Later if you want to grant administrative permissions to another user simply [add that user to the sadmin group](#).

Create the administrative user by typing:

```
sudo useradd -M -d /samba/users -s /usr/sbin/nologin -G sambashare sadmin
```

The command above will also create a group sadmin and add the user to both sadmin and sambashare groups.

Set a password and enable the user:

```
sudo smbpasswd -a sadmin
```

```
sudo smbpasswd -e sadmin
```

Next, create the Users share directory:

```
sudo mkdir /samba/users
```

[Set the directory ownership](#) to user sadmin and group sambashare:

```
sudo chown sadmin:sambashare /samba/users
```

This directory will be accessible by all authenticated users. The following command configures write/read access to members of the sambashare group in the /samba/users directory:

```
sudo chmod 2770 /samba/users
```

## Configuring Samba Shares

Open the Samba configuration file and append the sections:

```
sudo nano /etc/samba/smb.conf
```

/etc/samba/smb.conf

[users]path=/samba/users

  browseable = yes

  read only = no

  force create mode = 0660

  force directory mode = 2770

  valid users = @sambashare @sadmin[josh]path=/samba/josh

  browseable = no

  read only = no

  force create mode = 0660

  force directory mode = 2770

  valid users = josh @sadmin

[Copy](#)

The options have the following meanings:

- [users] and [josh] - The names of the shares that you will use when logging in.
- path - The path to the share.
- browseable - Whether the share should be listed in the available shares list. By setting to no other users will not be able to see the share.
- read only - Whether the users specified in the valid users list are able to write to this share.
- force create mode - Sets the permissions for the newly created files in this share.
- force directory mode - Sets the permissions for the newly created directories in this share.
- valid users - A list of users and groups that are allowed to access the share. Groups are prefixed with the @ symbol.

For more information about available options see the [Samba configuration file](#) documentation page.

Once done, restart the Samba services with:

```
sudo systemctl restart smb.service
```

```
sudo systemctl restart nmb.service
```

In the following sections, we will show you how to connect to a Samba share from Linux, macOS and Windows clients.

## Connecting to a Samba Share from Linux

Linux users can access the samba share from the command line, using the file manager or mount the Samba share.

Using the smbclient client

smbclient is a tool that allows you to access Samba from the command line. The smbclient package is not pre-installed on most Linux distros so you will need to install it with your distribution package manager.

To install smbclient on Ubuntu and Debian run:

```
sudo apt install smbclient
```

To install smbclient on CentOS and Fedora run:

```
sudo yum install samba-client
```

The syntax to access a Samba share is as follows:

```
mbclient //samba_hostname_or_server_ip/share_name -U username
```

For example to connect to a share named josh on a Samba server with IP address 192.168.121.118 as user josh you would run:

```
smbclient //192.168.121.118/josh -U josh
```

You will be prompted to enter the user password.

Enter WORKGROUP\josh's password:

Copy

Once you enter the password you will be logged into the Samba command line interface.

Try "help" to get a list of possible commands.

```
smb: \>
```

Copy

## Mounting the Samba share

To [mount](#) a Samba share on Linux first you need to install the cifs-utils package.

On Ubuntu and Debian run:

```
sudo apt install cifs-utils
```

On CentOS and Fedora run:

```
sudo yum install cifs-utils
```

Next, create a mount point:

```
sudo mkdir /mnt/smbmount
```

Mount the share using the following command:

```
sudo mount -t cifs -o username=username //samba_hostname_or_server_ip/sharename /mnt/smbmount
```

For example to mount a share named josh on a Samba server with IP address 192.168.121.118 as user josh to the /mnt/smbmount mount point you would run:

```
sudo mount -t cifs -o username=josh //192.168.121.118/josh /mnt/smbmount
```

You will be prompted to enter the user password.

Password for josh@//192.168.121.118/josh: \*\*\*\*\*

Copy

## Using GUI

Files, the default file manager in Gnome has a built-in option to access Samba shares.

1. Open Files and click on "Other Locations" in the sidebar.
2. In "Connect to Server", enter the address of the Samba share in the following format smb://samba\_hostname\_or\_server\_ip/sharename.
3. Click "Connect" and the following screen will appear:

- 
4. Select “Registered User”, enter the Samba username and password and click “Connect”.
  5. The files on the Samba server will be shown.



## Connecting to a Samba Share from macOS

In macOS, you can access the Samba Shares either from the command line or using the default macOS file manager Finder. The following steps show how to access the share using Finder.

1. Open “Finder”, select “Go” and click “Connect To”.
2. In “Connect To”, enter the address of the Samba share in the following format `smb://samba_hostname_or_server_ip/sharename`.

- 
3. Click "Connect" and the following screen will appear:

- 
4. Select "Registered User", enter the Samba username and password and click "Connect".
  5. The files on the Samba server will be shown.

X

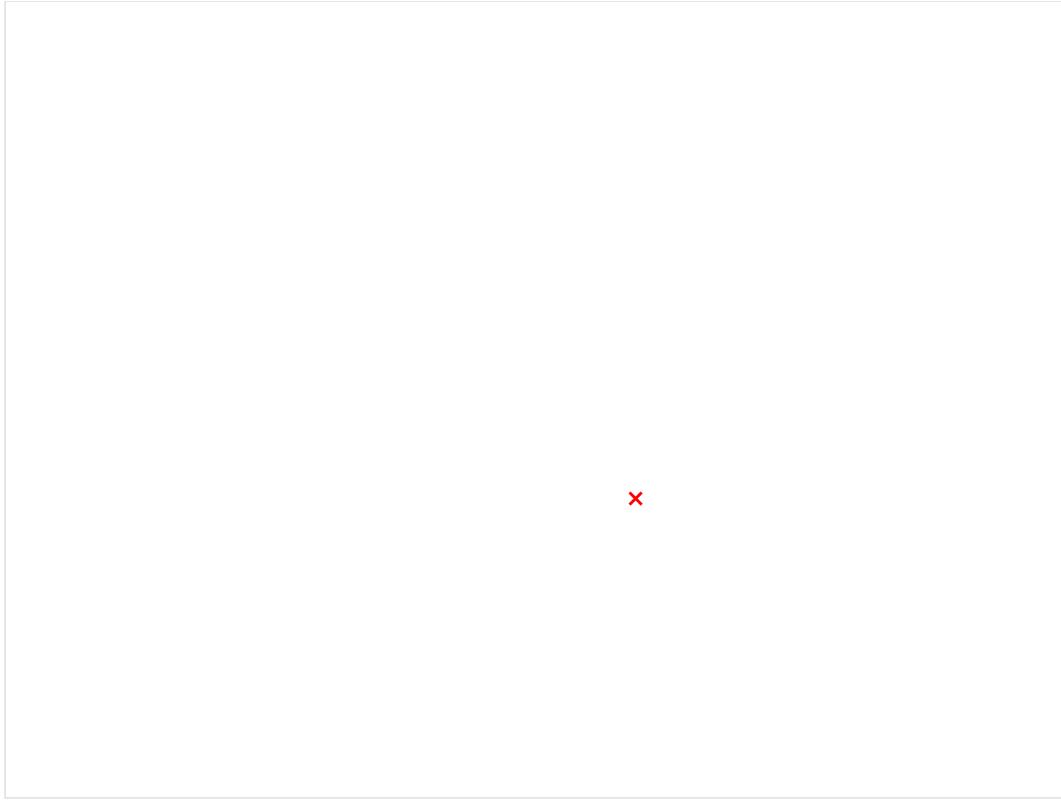
## Connecting to a Samba Share from Windows

Windows users also have an option to connect to the Samba share from both command line and GUI. The steps below show how to access the share using the Windows File Explorer.

1. Open up File Explorer and in the left pane right-click on “This PC”.
2. Select “Choose a custom network location” and then click “Next”.
3. In “Internet or network address”, enter the address of the Samba share in the following format [\\samba\\_hostname\\_or\\_server\\_ip\\sharename](\\samba_hostname_or_server_ip\\sharename).

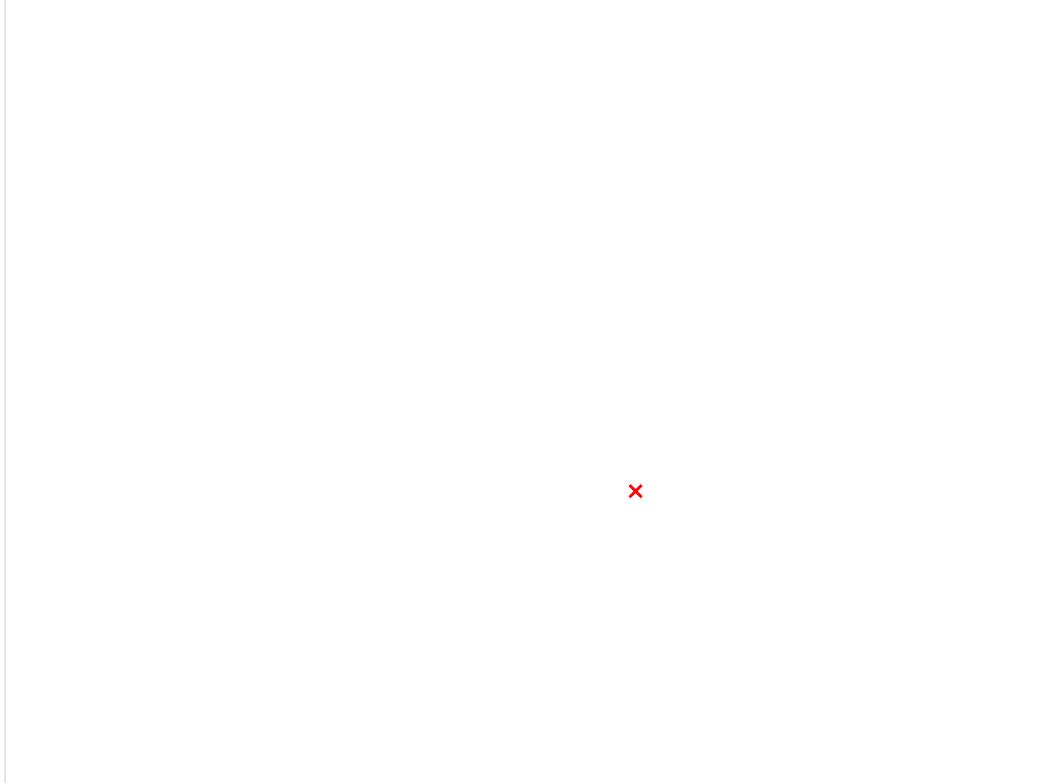
X

4. Click “Next” and you will be prompted to enter the login credentials as shown below:



X

5. In the next window, you can type a custom name for the network location. The default one will be picked up by the Samba server.



X

6. Click “Next” to move to the last screen of the connection setup wizard.
7. Click “Finish” and the files on the Samba server will be shown.



## Conclusion

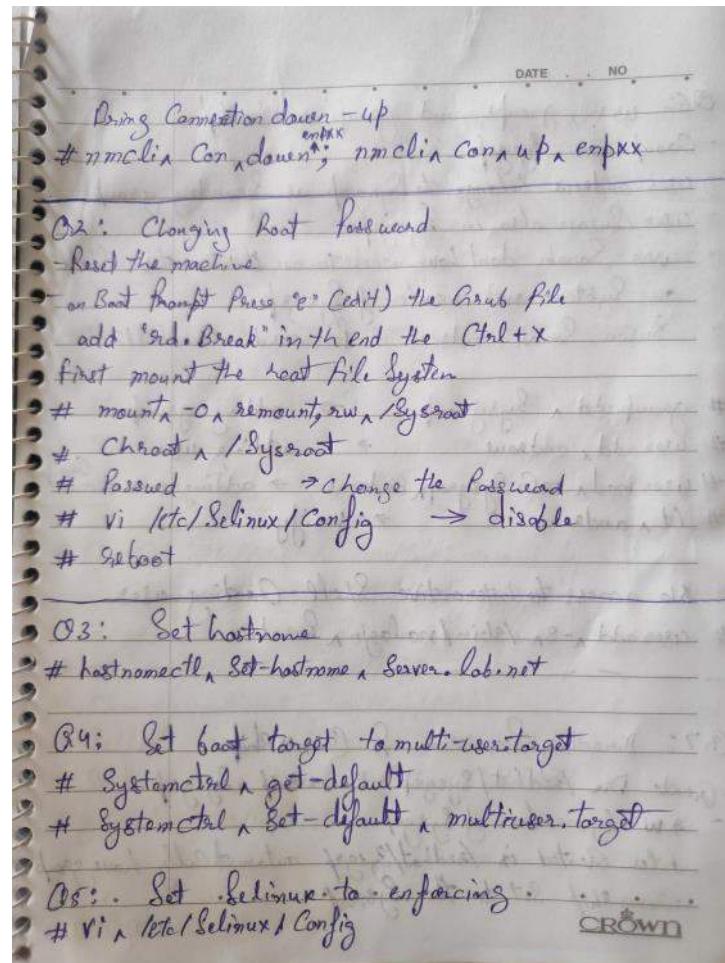
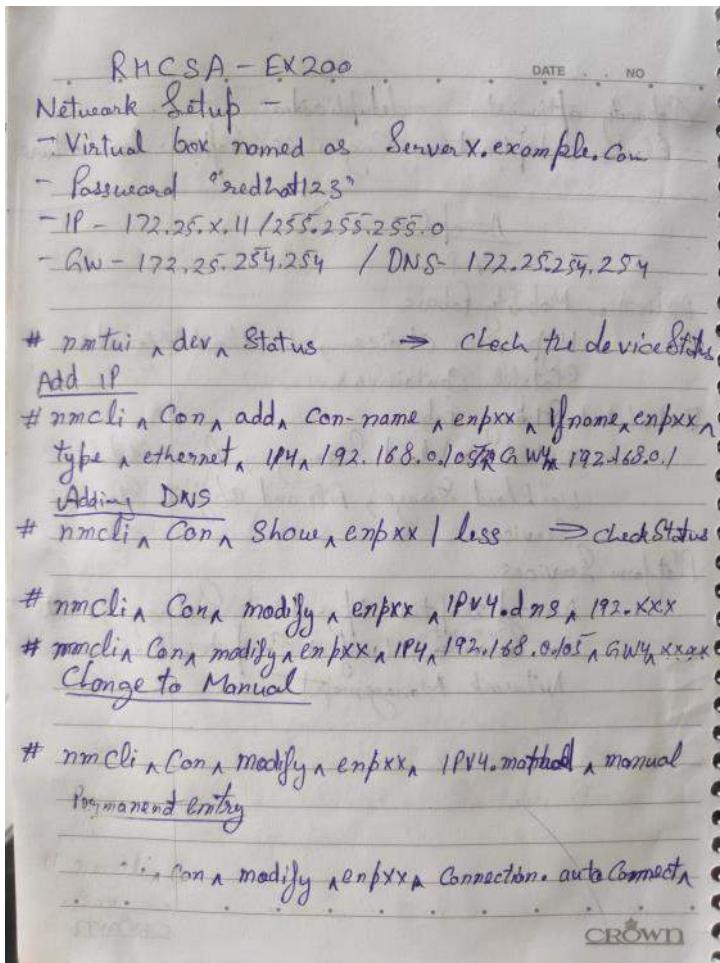
In this tutorial, you have learned how to install a Samba server on CentOS 7 and create different types of shared and users. We have also shown you how to connect to the Samba server from Linux, macOS and Windows devices.

From <<https://linuxize.com/post/how-to-install-and-configure-samba-on-centos-7/>>

# RHEL Exam

Wednesday, December 2, 2020 3:18 PM

IP, Change root Password, Set hostname, Boot target, Selinux Enforcing



User Group, Dir Permission Ownership, IP Forwarding,

Q6: user, groups, and groups membership.

- Group name sysgrp
- user andrew belongs to sysgrp as Secondary group.  
user Susan also in sysgrp  
user Sarah - don't have access to an interactive shell  
... system not member of sysgrp

Susan, Sarah, andrew - password = redhat123

# groupadd sysgrp → adding group

# useradd andrew → adding user

# usermod -aG sysgrp andrew → adding user as Secondary group

# id andrew → verify

No access to interactive Shell - Creating user

# useradd -s /sbin/nologin Sarah

Q7: Directory Permission & Ownership.

- Create Dir /redhat/sysgrp - Ownership sysgrp

- r.w by members of sysgrp  
Created in /redhat/sysgrp automatically have group  
chip Set to the sysgrp

CROWN

# mkdir -P /redhat/sysgrp

# ls -ltd /redhat/sysgrp

# Chgrp +sysgrp /redhat/sysgrp

# chmod 770 /redhat/sysgrp

# chmod 2770 /redhat/sysgrp

Q8 - IP Forwarding on your machine

# vi /etc/sysctl.conf

net.ipv4.ip\_forward = 1 → add this line

# Sysctl -P → enable

# Sysctl -a | grep -i net.ipv4 → check status

Q9: Copy /etc/fstab to /var/tmp: owned by root user  
Group root, not executable by anyone.

andrew can rw, Susan neither rw, All others r

# Cp -P /etc/fstab /var/tmp/

# Chown root:root /var/tmp/fstab

# Setfacl -m u:andrew:rwx -m /var/tmp/fstab

# Setfacl -m u:susan:--- /var/tmp/fstab

# getfacl /var/tmp/fstab

CROWN

Add user with ID, Create archive, Swap, Create Repo

Q10: Add user John with user id 1099, find file owned by user Sam and copy into root/Findress

```
# useradd -u 1099 -s /bin/bash -c "John" -m
# find / -user Sam
# cp /root/.ssh/id_rsa.pub ~root/Findress/
```

Q11: Create an archive file /root/local.tgz for local. It should be compressed by gzip.

```
# ll -l /root/local
# cd /root/local
# tar -cvzf /root/local.tgz /root/local
```

Q12 Search the string /etc/passwd. Save output in root/file

```
# cat /etc/passwd | grep -i root > /root/file
```

Q13 Create a new 150 MB Phy using /dev/sd6 and mount

```
# fdisk -l
# fdisk /dev/sd6
# partx /dev/sd6
# hdid -I /dev/sd6
# fdisk -l
```

Q14 - Increase Swap - Using SwapPartition - Swap File

```
# Swapon -S → Create Swap
# free -m
# fdisk /dev/sd6 - make Partition for Swap
# fdisk mkswap /dev/sd61
# Swapon /dev/sd61 ; Swapoff -S
# add Entry into /etc/fstab
```

Q15 - Create yum Repo

```
# vi /etc/yum.repos.d/myServer
[localrepo]
name = local repo for RHEL7.0
baseurl = http://Content.example.com/Rhel7.0/x86_64/Hdd
gpgcheck = 0
enable = 1
```

# Yum Clean all

Q16 Installation of Kernel

update kernel is default kernel when system rebooted  
 Original remains available and bootable on /sys/to.

# vi /etc/yum.repos.d/myserver.repo → Clear the repo  
# yum repolist  
# yum n install kernel  
# reboot

### Configure LDAP Client

# yum n install auth\* -y  
# yum n install ~~auth~~ authconfig-gtk,sssd → GUI  
Krb5-Workstation  
# authconfig-gtk → Auth Config GUI  
# getent n passwd ldapuser9

### Q18 Configure autofs to auto mount the home directory of LDAP users.

# systemctl n enable sssd.service  
# systemctl n start sssd.service  
# getent n passwd ldapuser9  
# yum n install autofs  
# vi /etc/autofs.master /home/autofs  
/home/guests /etc/autofs/home/  
# vi /etc/autofs/home/  
ldapuser9 → new Sync n classroom.example.com:  
Guest &

# systemctl n restart n autofs  
# Svc n -n ldapuser9 → you will be able to login  
with home directory

### Q19 NTP Client of Classroom

# yum n install chrony -y  
# vi /etc/chrony.conf  
Server n classroom.example.com iburst  
# systemctl n restart n chronyd  
Status  
enable

\* chronyc -v → verification

### Q20 Volume group

# Create n -S n 16M datavg n /dev/Sdb1  
# lvcreate -L 5G n datacopy n /dev/datavg  
# mkfs.n -t ext4 n /dev/datavg/datacopy  
# mdadm n /data Store  
# mount n -a

### Q21 Resize the logical volume

fumount n /datacopy  
# e2fsck n -F n /dev/datavg/datacopy

CROWN

# resize2fs /dev/datavg /datacopy 400M

# lvreduce -L 400M /dev/datavg /datacopy  
+ mount -a

#### - Cron Job

\* yum install cronie -y

# yum enable cronie

# Crontab -c -malkiat

30 \* \* \* \* /bin/echo hello

\* Crontab -l -malkiat

#### Bonus - FESTIVE OFFER 99

Proxy Server - retrieves the data from Internet on behalf of the user.

- Privacy - allows you to surf the Internet anonymously. It hides your IP. without the privacy when you visit any website your public IP address is visible.
- Cred - Cached Web Page Database on Proxy Server  
- It saves the Bandwidth.

CROWN

Activity logging - It keeps the record of employee where they visiting. You can also block certain websites to keep staff from visiting them.

Proxy Server cannot encrypt your data.

NAT - Network Address Translation that use in Routers. NAT translates a set of IP to another set of IP.

It helps to Reserve the limited amount of IPv4 Public IP address.

POP3 vs IMAP - used for retrieving email from our Email server.

POP3 - Post office Protocol 3

In POP3 the email is deleted on the mail server once it's downloaded to a device.

IMAP - Internet message access protocol. IMAP allows you to view your emails that's on the server from multiple devices.

CROWN

# Networking Qst

Tuesday, December 1, 2020 9:36 PM

## **1) What is a Link?**

A link refers to the connectivity between two devices. It includes the type of cables and protocols used in order for one device to be able to communicate with the other.

## **2) What are the layers of the OSI reference model?**

There are 7 OSI layers: Physical Layer, Data Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer and Application Layer.

## **3) What is backbone network?**

A backbone network is a centralized infrastructure that is designed to distribute different routes and data to various networks. It also handles management of bandwidth and various channels.

## **4) What is a LAN?**

LAN is short for Local Area Network. It refers to the connection between computers and other network devices that are located within a small physical location.

## **5) What is a node?**

A node refers to a point or joint where a connection takes place. It can be computer or device that is part of a network. Two or more nodes are needed in order to form a network connection.

## **6) What are routers?**

Routers can connect two or more network segments. These are intelligent network devices that store information in its routing table such as paths, hops and bottlenecks. With this info, they are able to determine the best path for data transfer. Routers operate at the OSI Network Layer.

## **7) What is point to point link?**

It refers to a direct connection between two computers on a network. A point to point connection does not need any other network devices other than connecting a cable to the NIC cards of both computers.

## **8) What is anonymous FTP?**

Anonymous FTP is a way of granting user access to files in public servers. Users that are allowed access to data in these servers do not need to identify themselves, but instead log in as an anonymous guest.

## **9) What is subnet mask?**

A subnet mask is combined with an IP address in order to identify two parts: the extended network address and the host address. Like an IP address, a subnet mask is made up of 32 bits.

## **11) What is data encapsulation?**

Data encapsulation is the process of breaking down information into smaller manageable chunks before it is transmitted across the network. It is also in this process that the source and destination addresses are attached into the headers, along with parity checks.

## **12) Describe Network Topology**

Network Topology refers to the layout of a computer network. It shows how devices and cables are physically laid out, as well as how they connect to one another.

## **13) What is VPN?**

VPN means Virtual Private Network, a technology that allows a secure tunnel to be created across a network such as the Internet. For example, VPNs allow you to establish a secure dial-up connection to a remote server.

**14) Briefly describe NAT.**

NAT is Network Address Translation. This is a protocol that provides a way for multiple computers on a common network to share single connection to the Internet.

**15) What is the job of the Network Layer under the OSI reference model?**

The Network layer is responsible for data routing, packet switching and control of network congestion. Routers operate under this layer.

**17) What is RIP?**

RIP, short for Routing Information Protocol is used by routers to send data from one network to another. It efficiently manages routing data by broadcasting its routing table to all other routers within the network. It determines the network distance in units of hops.

**19) What is NIC?**

NIC is short for Network Interface Card. This is a peripheral card that is attached to a PC in order to connect to a network. Every NIC has its own MAC address that identifies the PC on the network.

**20) What is WAN?**

WAN stands for Wide Area Network. It is an interconnection of computers and devices that are geographically dispersed. It connects networks that are located in different regions and countries.

**21) What is the importance of the OSI Physical Layer?**

The physical layer does the conversion from data bits to electrical signal, and vice versa. This is where network devices and cable types are considered and setup.

**22) How many layers are there under TCP/IP?**

There are four layers: the Network Layer, Internet Layer, Transport Layer and Application Layer.

**23) What are proxy servers and how do they protect computer networks?**

Proxy servers primarily prevent external users who identifying the IP addresses of an internal network. Without knowledge of the correct IP address, even the physical location of the network cannot be identified. Proxy servers can make a network virtually invisible to external users.

**24) What is the function of the OSI Session Layer?**

This layer provides the protocols and means for two devices on the network to communicate with each other by holding a session. This includes setting up the session, managing information exchange during the session, and tear-down process upon termination of the session.

**36) What is the main purpose of OSPF?**

OSPF, or Open Shortest Path First, is a link-state routing protocol that uses routing tables to determine the best possible path for data exchange.

**37) What are firewalls?**

Firewalls serve to protect an internal network from external attacks. These external threats can be hackers who want to steal data or computer viruses that can wipe out data in an instant. It also prevents other users from external networks from gaining access to the private network.

# Linux Cheat Sheet

Wednesday, December 2, 2020 10:28 PM

SYSTEM		FILE PERMISSION RELATED	
uname -a	=>Display linux system information	chmod octal file-name	=>Change the permissions of file to octal
uname -r	=>Display kernel release information	Example	
uptime	=>Show how long the system has been running + load	chmod 777 /data/test.c	=>Set rwx permission for owner,group,world
hostname	=>Show system host name	chmod 755 /data/test.c	=>Set rwx permission for owner,rx for group and world
hostname -i	=>Display the IP address of the host	chown owner-user file	=>Change owner of the file
last reboot	=>Show system reboot history	chown owner-user:owner-group file-name	=>Change owner and group owner of the file
date	=>Show the current date and time	chown owner-user:owner-group directory	=>Change owner and group owner of the directory
cal	=>Show this month calendar		
w	=>Display who is online		
whoami	=>Who you are logged in as		
finger user	=>Display information about user		
HARDWARE		NETWORK	
dmesg	=>Detected hardware and boot messages	ip addr show	=>Display all network interfaces and ip address (a iproute2 command, powerful than ifconfig)
cat /proc/cpuinfo	=>CPU model	ip address add 192.0.1 dev eth0	=>Set ip address
cat /proc/meminfo	=>Hardware memory	ethtool eth0	=>Linux tool to show ethernet status
cat /proc/interrupts	=>Lists the number of interrupts per CPU per I/O device	miitool eth0	=>Linux tool to show ethernet status
lshw	=>Displays information on hardware configuration of the system	ping host	=>Send echo request to test connection
lsblk	=>Displays block device related information in Linux	whois domain	=>Get who is information for domain
free -m	=>Used and free memory (-m for MB)	dig domain	=>Get DNS information for domain
lspci -tv	=>Show PCI devices	dig -x host	=>Reverse lookup host
lsusb -tv	=>Show USB devices	host google.com	=>Lookup DNS ip address for the name
dmidecode	=>Show hardware info from the BIOS	hostname -i	=>Lookup local ip address
hdparm -i /dev/sda	=>Show info about disk sda	wget file	=>Download file
hdparm -T /dev/sda	=>Do a read speed test on disk sda	netstat -tulp	=>Listing all active listening ports
badblocks -s /dev/sda	=>Test for unreadable blocks on disk sda		
USERS		COMPRESSION / ARCHIVES	
id	=>Show the active user id with login and group	tar cf home.tar home	=>Create tar named home.tar containing home/
last	=>Show last logins on the system	tar xf file.tar	=>Extract the files from file.tar
who	=>Show who is logged on the system	tar czf file.tar.gz files	=>Create a tar with gzip compression
groupadd admin	=>Add group "admin"	gzip file	=>Compress file and renames it to file.gz
useradd -c "Sam Tomshi"	=>g admin -m sam #Create user "sam"		
userdel sam	=>Delete user sam		
adduser sam	=>Add user "sam"		
usermod	=>Modify user information		
FILE COMMANDS		INSTALL PACKAGE	
ls -al	=>Display all information about files/ directories	rpm -i pkgname.rpm	=>Install rpm based package
pwd	=>Show the path of current directory	rpm -e pkgname	=>Remove package
mkdir directory-name	=>Create a directory		
rm file-name	=>Delete file		
rm -r directory-name	=>Delete directory recursively		
rm -f file-name	=>Forcefully remove file		
rm -rf directory-name	=>Forcefully remove directory recursively		
cp file1 file2	=>Copy file1 to file2		
cp -r dir1 dir2	=>Copy dir1 to dir2, create dir2 if it doesn't exist		
mv file1 file2	=>Rename source to dest / move source to directory		
In -s /path/to/file-name link-name	=>#Create symbolic link to file-name		
touch file	=>Create or update file		
cat > file	=>Place standard input into file		
more file	=>Output contents of file		
head file	=>Output first 10 lines of file		
tail file	=>Output last 10 lines of file		
tail -f file	=>Output contents of file as it grows starting with the last 10 lines		
gpg -c file	=>Encrypt file		
gpg file.gpg	=>Decrypt file		
wc	=>print the number of bytes, words, and lines in files		
xargs	=>Execute command lines from standard input		
PROCESS RELATED		INSTALL FROM SOURCE	
ps	=>Display your currently active processes	/configure	
ps aux   grep 'telnet'	=>Find all process id related to telnet process	make	
pmap	=>Memory map of process	make install	
top	=>Display all running processes		
kill pid	=>Kill process with mentioned pid id		
killall proc	=>Kill all processes named proc		
pkill process-name	=>Send signal to a process with its name		
bg	=>Resumes suspended jobs without bringing them to foreground		
fg	=>Brings the most recent job to foreground		
fg n	=>Brings job n to the foreground		
FILE TRANSFER		SEARCH	
scp		grep pattern files	=>Search for pattern in files
scp file.txt server2:/tmp		grep -r pattern dir	=>Search recursively for pattern in dir
rsync -a /home/apps /backup/		locate file	=>Find all instances of file
		find /home/tom -name 'index*'	=>Find files names that start with "index"
		find /home -size +10000k	=>Find files larger than 10000k in /home
LOGIN (SSH AND TELNET)		DISK USAGE	
ssh user@host		df -h	=>Show free space on mounted filesystems
ssh -p port user@host		df -i	=>Show free inodes on mounted filesystems
telnet host		fdisk -l	=>Show disks partitions sizes and types
DIRECTORY TRAVERSE		FILE TRANSFER	
cd ..	=>To go up one level of the directory tree	du -ah	=>Display disk usage in human readable form
cd	=>Go to \$HOME directory	du -sh	=>Display total disk usage on the current directory
cd /test	=>Change to /test directory	findmnt	=>Displays target mount point for all filesystem
		mount device-path mount-point	=>Mount a device

## OpenSSL generate and install

Tuesday, December 8, 2020 12:40 PM

## Secure Sockets Layer (SSL Certificate)

SSL stands for Secure Sockets Layer. It is used to secure the connection between internet browsers and Web server or websites by transferring the encrypted data rather than plain text.

- \* SSL certificate issued by the Certificate Authority (CA) and.
  - \* Self-Signed SSL certificate.

### Step 1) Generate Private Key On The Server

OpenSSL is the open source SSL package that comes along with most of the linux distros. Make sure openssl package is installed. Refer the screenshot.  
We are generating private key with **openssl** command as shown below.

```
# openssl genrsa -des3 -out www.domain.com.key 2048
```

```
root@server ~# ls -l www.domain.com.key
-rw-r--r-- 1 root root 1743 Jul 6 15:51 www.domain.com.key
root@server ~# #
root@server ~# #
root@server ~# #
root@server ~# #
root@server ~# cat www.domain.com.key
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,D4923B0F64FA0DE2.

n26FLvW/tYgaKqpo9vcAqgb1lnR3f+Wv0y0BcZ79Y3Ute2f9q3T02gt21asXLR
X+X|R-+fb1Bt1ML0Rkp4zuJqfCqk2Ptk8zNjH/czp2n85VxTeJQkBwmpY
3L7R3Q4RzH6zsHnuJ9B8f4e-nu3tK4Ez2zgts4idhTrEsGQRb1POM78a232j
9os112z0Ln1ngfGtVC8MHTz+XZK15BKg9t1NonrVamVjKo1estHtvCx3Uv37u
K1arGuP03ZB2K9znD+Ev1aH+Hu0w1MyKh2nQnUsu71znaJL+03jBz1d3
vAvMeJ3x3HeJH_7TNNKhATyavv0rpBjZDQ+kYhV8x0Qn9821z0hy1vkz26R
dn9+UxLje2wLk15W5AM4GUtEnDdGe0pFfnuq+GtYq0pToFegpppBx/GBl0v
JLm+TzSs5b9hM+MB87zqGuh+CR5s0e0nJwcj11Tf24hN7Qz7cLMQG5s0b0K
YlW+0VksWGbKtsd7og4nT/UVGNz5TaxVuQh9LnANEdWQnaGorQlQu22x0j
DDNSvRERGeusKhe14x5pb5877Tx0xKwHk+zro2DEz+rxNraQsF5x9vWvM
L7Bp6zverpn4a0WpD+T7qborKf17K/nsslrlrgRCH1L2L3JEf0FuTx0vwxqaz
XzcFgBpuCu9tJ1e1J+U2pJNayFa/qExv3sxxn44iQKwC199EX1/NTRP
Lm1BPj1XzKw1Bg6sdJtmjy5+S1dnchxRrSL5F7naNS1YAB8xgn9hklBx
8fShH+etq+tpEoIjNY2088SPBZcpnbw+lw94V3A3YehPtSz+zz52751B10gBH
167b14B0EWPoQeByJNUw+VtrF9h/_Orc/CsD1MTN4cw2lJpK9zJ0f2B0uXGM
MBNfWAcPseuseC9Bv1u0epu3khbvxxy4pkulHOHchTE7v8LM+rpdJeq3gTy1
P0L1Qng5pB6p7mtD10V1Lm6yZjQbGtdGSlnV2X9y+Ma7qrNFC1tz151
3nTf08B1hhB8hL9u19z3s0Ea7x+39uIMt9n4E7KHyWv8/a/SVPHN08e4WwAj
VLSt3ebApEcseUyLB4an/Bm7f0lTsV7w9rsB5g0XSYt8RtIO0m+jqpxC92
57PSzkybKw3bpn6X0tA1Dp0zgevB12gr/dXgpnj+TwEvglLbtgcj72vad
c3z/Ew01uHdu50d549554SeHw/Hl/wTk-uaoJRL55QpUtd7e9+7LcdTrvCr
odCPOTX15Akqgeyp/H2cmNf8NQb9hcs+clc/_/x1rV0PLvxRw9p30P1lsBqjlo3
qk93XyL218z6N85h7gKpzb/TUnkrbV0L3r8NFnG6jkuC9uB12s9Pf0Mu
nTGHnUppnqd2z1d5t5vhLyKwghgBa8P0rfYgA7h70tX1Xda5504r+4SNHtJAHF
5nfTFCz4NCapKkqMgkByrC3u1N02xDz5TczmyE7BcYY3g7IRZhoUw==
-----END RSA PRIVATE KEY-----
root@server ~#
```

### Step 1) Generate Private Key On The Server

- # openssl genrsa -des3 -out [www.domain.com.key](#) 2048

#### Step 2) Generate Certificate Signing Request (CSR)

- # openssl req -new -key [www.domain.com.key](#) -out [www.domain.com.csr](#)

### Step 3) Create SSL Certificate

- CA verified certificate you need to provide CSR and Private key to the Certificate vendor.

## Generating Self-Signed certificate

- # openssl x509 -req -days 365 -in [www.domain.com.csr](#) -signkey [www.domain.com.key](#) -out [www.domain.com.crt](#)

To install this certificate for a website, you need to create a new VirtualHost for the domain name because SSL is using a different port and not the common port 80. SSL port is 443. So Apache will be listening to both 80 and 443 for the non-encrypted and encrypted data respectively. Or, you can create a separate conf file in `/etc/httpd/conf.d` directory and then ask Apache to refer the said directory with 'Include' directive as shown below.

```
# Include conf.d/*.conf
```

Now, add the below given code either in the VirtualHost or in the separate configuration (eg: ssl.conf) file created in the `/etc/httpd/conf.d` direcory.

```
# SSLEngine on  
# SSLCertificateFile /path_of_crt_file/www.domain.com.crt  
# SSLCertificateKeyFile /path_of_key_file/www.domain.com.key
```

## Step 2) Generate Certificate Signing Request (CSR)

After generating your private key, you need to generate a CSR (Certificate Signing Request). You can easily create that with openssl command.

```
# openssl req -new -key www.domain.com.key -out www.domain.com.csr
```

Few questions regarding the website identity will be asked and this will be checked by the certificate authority. You can refer the screenshot for this.

```
root@server [-]# openssl req -new -key www.domain.com.key -out www.domain.com.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank.
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:GB
State or Province Name (full name) []:london
Locality Name (eg, city) [Default City]:Old Gloucester Street
Organization Name (eg, company) [Default Company Ltd]:Pickaweb
Organizational Unit Name (eg, section) []::Pickaweb
Common Name (eg, your name or your server's hostname) []:server.domain.com
Email Address []:webmaster@domain.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:Pickaweb
root@server [-]#
```

CSR will be generated in the present working directory with the file name '[www.domain.com.csr](#)'. Here is the screenshot of the CSR file.

```

root@server [-]# ls -l www.domain.com.csr
-rw-r--r-- 1 root root 1123 Jul 6 16:18 www.domain.com.csr
root@server [-]#
root@server [-]#
root@server [-]# cat www.domain.com.csr
-----BEGIN CERTIFICATE REQUEST-----
MIIDBTCAe0CAoAwgaYXcZAjBgNVBAYTAKdCMQ8wDQYDVQQIDAzb25kb24xHjAc
BgNVBACMFU95ZCBhbG91Y2zdGvIiFN0cmVldERMA8GA1UECgwIUGljzAF3ZWIx
EjAQBgNVBAsMCTpQaWnRyXdlYjEaMBgGA1UEAwRc2VydmyLmRvbWFpb15jb20X
IxAhBgkqhklG9w0BCQENFHdlym1hc3RlcBkb21haW4U29M1IBjANBgkqhkg
9w0BAQEAAQAMIBcgKCAQEAOxoPUgFrzy/BV1eaknNQ1iejzN4CE3urus
JYYAXIn8PNbFL920ev0P1bQXGhbnsi0aJAn7oK02wp5Edqia3PszoArDhjXCH41R
cdzOzCwczK4pgaffzF/M+jcQ2AzaQ1/2XUsch9NYskauLskwM8QlqjWLM8CPK
t+bWe0B4C20Gmdrfu/8nc/x0S/4y3kVgFa3hUSHM/nr1+lA7nwM60zRR07
X9J9saylY2gwzotBpeXLhoiR/aAV04c7ew4mRYhd+o5lkx5iyBmwMg7kwbp0G
IPxcROGQHjlq1j0t3xyRgEKKJx4itGWFe3E0rUB0feD9VTCB7IQIDAQABoBkwFwG
KoZThvNAQkCMQCFBpY2thd2VlMAOGCSqSGSib3DQEBBQUA41BAQGc/4AT8Ak
8xgjIBdAtyH/OF6L6T6J/28NP3yc45dxsY181npAxCKzPjgrVRwn0ew0mow3r090
23likJNWQxP4gep7vXNzSkYFDaNdMt40TU810mZhf1cAnc18ikM5nYdzcnSm1Yv
HB9UKUivwQg04kpmBp3lgKgKLgHyEFCJ/z14tuXnxc0ZPzCtvC1qkQEt6Nady8
K4U29xhTkpgG8jod+ZCF8lbd5bb7L0PDqjx5ZF2mNMxueMvnDjesmtOpTF0x7LQ
Boyyb2K2d6GxTnj9P4S0eEkTvlMsMzI4jg0Nhjxj+dbb9j587VGCaKRZ2VsfpIqpj
JivHCux/68z/
-----END CERTIFICATE REQUEST-----
root@server [-]#

```

## Step 3) Create SSL Certificate

After generating Private key and CSR, you need to create the SSL certificate. Now is where the difference come into play.

For a CA verified certificate you need to provide CSR and Private key to the Certificate vendor. They will provide a CA verified certificate file (.crt file) and you can install it. But for a Self-Signed certificate, you need to generate the certificate manually.

### Generating Self-Signed certificate

Certificate file will be generated with private key and CSR encoded in it. All the information for in the Private key and CSR will be encoded in the .crt file. Command is given below.

```
# openssl x509 -req -days 365 -in www.domain.com.csr -signkey www.domain.com.key -out www.domain.com.crt
```

If you refer the screenshot, you can see that the data entered while creating CSR is encoded.

```

root@server [-]# openssl x509 -req -days 365 -in www.domain.com.csr -signkey www.domain.com.key -out www.domain.com.crt
signature ok
subject=/C=GB/ST=London/L=Old Gloucester Street/O=Pluckaweb/OU=Pluckaweb/CN=server.domain.com/emailAddress=webmaster@donaln.com
getting Private key
Enter pass phrase for www.domain.com.key:
root@server [-]#

```

Certificate file will be generated in the present working directory as '[www.domain.com.crt](#)', please note that domain.com is my domain name in this example and it should be replaced with the actual domain name. Here is the generated .crt file

```

root@server [-]# cat www.domain.com.crt
-----BEGIN CERTIFICATE-----
MIIDyjC0RCn51RVDAhjzANBgkqhklG9w0BAQUFADCBpjELMAkGA1UEBhM
R0IxZDAnBgNVBAGMbmxvbnRvbjEeMBwGA1UEBwVT2xkIEdsb3jZXN0ZXIGU3Y
ZWN0MREwDwYDQVQKDQhAqWnRyXdlYjESMBAGA1UECwJ0LByp2thd2VlMRowGAYD
VQQRvDBFzzXZKX1uzG9tYwlulmNvbTEjMCEGCSqSGSib3DQEBAQQA41BDwAwggEK
QGRvdbFpbis5jb2whlCNMTUWnzaMTQ0Nj04WhcNMTYhWnZa1MTQ0Nj04jCbpjEL
MAKGA1UEBhMCR0IXDzANBgNVBAGMbmxvbnRvbjEeMBwGA1UEBwVT2xkIEdsb3j
ZXN0ZXIgU3ryZWV0MREwDwYDQVQKDQhAqWnRyXdlYjESMBAGA1UECwJ0LByp2th
d2VlMRowGAYDQVQODBFzzXZKX1uzG9tYwlulmNvbTEjMCEGCSqSGSib3DQEBAQQA41BDwAwggEK
d2VlbwFzGvYQGRvbfpb15jb2whggElMA0GCSqSGSib3DQEBAQQA41BDwAwggEK
AoIBAQDTGg95ohNvLP8FXV5q5c2FCUh6PM3hwTdi6u51lhgBcfwB18v3bR5XQPV
tBcaFueyRokCfugo7Cnkr2qJrc+zogCs0GNCifjVfX3N07MjZMkrImBp9jMX8
z6nX9DyVlpAj/ZdsxxZj01iyQ80uyQxxAurVaIzw18q35Lz70Hhzbs5gZ00+kZt9
T/ydz/ESL/jleRmovreFR1f0+avX6UDufCbo7NFTtf0n2xrXkjjaDD0i0G15eUe
jWJH9oBu7hzt5b1zf1F36jmWTHmLIGbAyBXuTBumQYg/FxE4ZAEwVDK3fHJGAQ
oonHk0ZV7TStQe594Pi1M1HshAgMBAEwDQYJKoZIhvcaNAQEFBQAdggEBAB54
oDeobxdCjHJF/LiberAc1ybJw50K+Ke2yp+in1nx5HopC25q6nwULef10Q0g0l
EbjLYOCOfk1qtfsY1nkqY20NrbwHfrPU+ /arXr92Dclykjy4KcxJ0dw4ny9Fe
vg7Uj9n6a8Z8DKWl9PM5J9sD7+g+sVLWyoQAGCeBM2jfJkmW8
ID0ltZTpf7Ux81Z6o1kjldXjD+6enu2KTPGc8tnkAn6mKygyHxD9EweAh1Zj
I/XPTVVRoEN25wnCnddyMIS2QvmLJmmKMSN/BxwQGcsI+jIy8JXkyUvE6agPBGY
trzOSIEw6Y0xQ19T6NN=
-----END CERTIFICATE-----
root@server [-]#

```

To install this certificate for a website, you need to create a new VirtualHost for the domain name because SSL is using a different port and not the common port 80. SSL port is 443. So Apache will be listening to both 80 and 443 for the non-encrypted and encrypted data respectively. Or, you can create a separate config file, in **/etc/httpd/conf.d** directory and then ask Apache to refer the said directory with 'Include' directive as shown below.

```
# Include conf.d/.conf
```

Now, add the below given code either in the VirtualHost or in the separate configuration (eg: ssl.conf) file created in the **/etc/httpd/conf.d** direcory.

```
# SSLEngine on
# SSLCertificateFile /path_of_crt_file/www.domain.com.crt
# SSLCertificateKeyFile /path_of_key_file/www.domain.com.key
```

This will tell apache to refer the .crt (certificate) file and .key (Private key) file for SSL encrypted connection.

#### **Step 4) Restart Apache**

Final step is to restart Apache service for the changes to take effect.  
`/etc/init.d/httpd restart`

# YUM Cheat Sheet

Wednesday, December 2, 2020 10:45 PM

## YUM COMMAND CHEAT SHEET

### CLEANUP DUPLICATES in YUM

```
for i in $(package-cleanup --dups); do rpm -e --justdb $i --nodeps; done
```

### Yum Queries:

**help** - Display yum commands and options

**yum help** - Show yum subcommands and options

#### **list** - List package names from repositories

**yum list available** - List all available packages

**yum list installed** - List all installed packages

**yum list all** - List installed and available packages

**yum list kernel** - List installed and available kernel packages

#### **info** - Display information about a package

**yum info vsftpd** - List info about vsftpd package

#### **deplist** - Display dependencies for a package

**yum deplist nfs-utils** - List dependencies and packages providing them

#### **provides** - Find packages that provide the queried file

**yum provides “\*bin/top”** - Show package that contains top command

**yum provides “\*/README.top”** - Show package containing README.top file

#### **search** - Search package names and descriptions for a term

**yum search samba** - Find packages with samba in name or description

#### **updateinfo** - Get information about available package updates

**yum updateinfo security** - Get info on available security updates

**grouplist** - List names of installed and available package groups

**groupinfo** - Display description and contents of a package group

**yum groupinfo “Web Server”** - See packages in Web Server group

**check-update** - Query repositories for available package updates

### Manage YUM repositories:

**repolist** - Display enabled software repositories

**repoinfo** - Display information about enabled yum repositories

**yum repoinfo rhel-7-server-rpms** - See info on rhel-7-server-rpms repo

#### **repo-pkgs** - Work with packages in a particular repository

**yum repo-pkgs my-rpms** - list List packages from my-rpms repo

**yum repo-pkgs my-rpms install** - Install all packages from my-rpms repo

**yum repo-pkgs my-rpms** - remove Remove all packages from my-rpms repo

**makecache** - Download yum repository data to cache

## Troubleshoot and maintain YUM:

**check** - Check the local RPM database for problems (runs for a long time)

**history** - View and use yum transactions

**yum history list** - List all yum install, update and erase actions

**yum history info 3** - Show details of yum transaction 3

**yum history undo 3** - Undo the yum action from transaction 3

**yum history redo 3** - Redo the undone yum action from transaction 3

**clean** - Clear out cached package data

**yum clean packages** - Delete packages saved in cache

**yum clean all** - Clean out all packages and meta data from cache

**fssnapshot** - List LVM snapshots (helps roll back after package updates)

**fs** - Act on filesystem (prevent doc or language file install on minimal systems)

**yum fs filters** - List enabled filesystem filters

**yum fs documentation** - Filters all docs from being installed (careful!)

## Install, Remove and Upgrade packages with YUM:

**install** - Install a package from a repository to your system

**yum install vsftpd** - Install the vsftpd package

**update** - Update one or all packages on your system

**yum update** - Update all packages with available updates

**yum update httpd** - Update the httpd package (if available)

**yum update --security** - Apply security-related package updates

**update-to** - Update one or all packages to a particular version

**upgrade** - Update packages taking obsoletes into account

**localinstall** - Install a package from a local file, http, or ftp

**yum localinstall abc-1-1.i686.rpm** - Install abc package from local directory

**yum localinstall <http://myrepo/abc-1-1.i686.rpm>** - Install abc from FTP site

**downgrade** - Downgrade a package to an earlier version

**yum downgrade abc** - Downgrade the abc package to an earlier version

**reinstall** - Reinstall the current version of a package

**yum reinstall util-linux** - Reinstall util-linux (to replace any deleted files)

**swap** - Remove one package and install another

**yum swap ftp lftp** - Remove ftp package and install lftp package

**erase** - Erase a package (and possibly dependencies) from your system

**yum remove vsftpd** - Remove the vsftpd package and dependencies  
**remove** - Same as erase

**autoremove** - Same as erase, plus removes additional unneeded packages  
**yum autoremove httpd** - Remove httpd and other unneeded packages

**groupinstall** - Install all packages in the selected group  
**yum groupinstall "Web server"** - Install Web Server packages

## Manage language packages with YUM (RHEL 7 feature):

**langavailable** - List all available languages

**langinfo** - List packages available for a language  
**yum langinfo es** - List packages associated with Spanish language

**langinstall** - Install packages associated with a particular language \*  
**yum langinstall es** - Install packages associated with Spanish language

**langlist** - List languages that are installed

**langremove** - Remove installed language packs for a language  
**yum langremove es** - Remove packages associated with Spanish language

From <<https://sites.google.com/site/mydocsrkab/linux/redhat6-common-troubleshooting/yum-command-cheat-sheet>>

# Zypper Cheat Sheet

Saturday, December 12, 2020 5:43 PM

## Zypper Cheat Sheet

For Zypper version 1.0.9

### Basic Help

`zypper #list the available global options and commands`  
`zypper help [command] #Print help for a specific command`  
`zypper shell or zypper sh #Open a zypper shell session`

### Repository Management

#### Listing Defined Repositories

`zypper repos or zypper lr`

Examples:

`zypper lr -u #include repo URI on the table`  
`zypper lr -P #include repo priority and sort by it`

#### Refreshing Repositories

`zypper refresh or zypper ref`

Examples:

`zypper ref packman main #specify repos to be updated`  
`zypper ref -f upd #force update of repo 'upd'`

#### Modifying Repositories

`zypper modifyrepo or zypper mr`

Examples:

`zypper mr -d 6 #disable repo #6`  
`zypper mr -rK -p 70 upd #enable autorefresh and rpm files 'caching' for 'upd' repo and set its priority to 70`  
`zypper mr -Ka #disable rpm files caching for all repos`  
`zypper mr -kt #enable rpm files caching for remote repos`

#### Adding Repositories

`zypper addrepo or zypper ar #followed by the repo url and alias`

Example:

`zypper ar http://download.opensuse.org/update/11.1/ update`

#### Removing Repositories

`zypper removerepo or zypper rr`

Examples:

`zypper rr packman main`

#### Renaming Repositories (for the alias only)

`zypper namerepo or zypper nr`

Examples:

`zypper nr 3 upd`

More Information:

[https://en.opensuse.org/SDB:Zypper\\_usage](https://en.opensuse.org/SDB:Zypper_usage) or type `man zypper` on a terminal

Page 1



### Package Management

#### Selecting Packages

By capability name:

`zypper in perl(Log::Log4perl)`

`zypper in qt`

By capability name and/or architecture and/or version

`zypper in 'zypper <0.12.10'`

`zypper in zypper.i586=0.12.11`

By exact package name (--name)

`zypper in -n ftp`

By exact package name and repository (implies --name)

`zypper in factory:zypper`

By package name using wildcards

`zypper in yast*ftp*`

By specifying a .rpm file to install

`zypper in skype-2.0.0.72-suse.i586.rpm`

#### Installing Packages

`zypper install or zypper in`

Examples:

`zypper install git`

By capability they provide

`zypper in MozillaFirefox < 3`

Others

`zypper in yast* #install all yast modules`

`zypper in -t pattern lamp_server #install lamp_server pattern (packages needed for a LAMP server)`

`zypper in vim -e emacs #install vim and remove emacs`

`zypper in amarok upd:libxine1 #install libxine1 from upd`

#### Removing Packages

`zypper remove or zypper rm`

Examples:

`zypper remove sqlite`

#### Export/Import Repositories

`zypper repos --export or zypper lr -e`

Examples:

`zypper lr --export backups/repos/foo.repo`

`zypper ar backups/repos/foo.repo --import`

### Source Packages and Build Dependencies

`zypper source-install or zypper si`

Examples:

`zypper si zypper`

Install only the source package

`zypper in -D zypper`

Install only the build dependencies

`zypper in -d zypper`

### Updating Packages

`zypper update or zypper up`

Examples:

`zypper up #update all installed packages with newer version as far as possible`

`zypper up libzyppp zypper #update libzyppp and zypper`

`zypper in sqlite3 #update sqlite3 or install if not yet installed`

### Zypper in Scripts and Applications

#### Non Interactive Mode

`zypper --non-interactive`

Examples:

`zypper --non-interactive patch #skips all interactive patches which would require user confirmation`

#### No GPG Checks Mode

`zypper --no-gpg-checks`

#### Auto-agree with Licenses

`zypper --auto-agree-with-licenses`

#### Quiet Output

`zypper --quiet`

#### XML Output

`zypper --xmlout`

## Zypper Cheat Sheet

For Zypper version 1.0.9

### Querying

#### Searching Packages

`zypper search or zypper se`

Examples:

`zypper se -dC --match-words RSI #look for RSI acronym (case-sensitively), also in summaries and descriptions`

`zypper se 'yast*' #show all packages starting with 'yast'`

`zypper se -r upd #list all packages from 'upd' repository`

`zypper se -i sqlite #show all 'sqlite' installed packages`

`zypper se -t pattern -r upd #list all patterns available in the 'upd' repository`

#### Getting Information about Packages

`zypper info or zypper if`

Examples:

`zypper info amarok`

`zypper info -t patch amarok #show info for 'amarok' patch`

`zypper patch-info amarok #same as above`

`zypper info -t pattern lamp_server #info 'lamp_server' pattern`

#### Getting Information about Dependencies

`zypper what-provides or zypper wp`

Examples:

### Package Locks

#### Lock Packages

`zypper addlock or zypper al`

Examples:

`zypper al 'yast2*' #lock all packages starting with 'yast2'`

#### Remove Locks

`zypper removelock or zypper rl`

Examples:

`zypper rl 'yast2*' #remove locks to all packages starting with 'yast2'`

#### List Locks

`zypper locks or zypper ll`

### Update Management

#### Listing Needed Patches

`zypper list-patches or zypper lp`

#### Applying Patches

`zypper patch`

#### Listing All Patches

### Distribution Upgrade

`zypper dist-upgrade or zypper dup`

Note:

When doing a distribution update, the best is to work only with the repositories of the distribution you want to install.

### Vocabulary

#### Repositories

HTTP or FTP server, DVD, or a folder on a local disc, where a group or set of packages are located.

#### Resource Identifiers (URI)

To specify locations of repositories or other resources (RPM files, .repo files) you can use any type of URLs supported by libzypp. See <http://en.opensuse.org/Libzypp/URI> for a complete list and usage examples.

#### Refresh

Refreshing a repository means downloading metadata of packages from the medium (if needed), storing it in local cache (typically under `/var/cache/zypp/raw/<alias>` directory) and parsing the metadata into .solv files (building the solv cache), typically under `/var/cache/zypp/solv/<alias>`.

#### Services

Services are one level above repositories and serve to manage repositories or to do some special tasks.

#### Getting Information about Dependencies

`zypper what-provides` or `zypper wp`

Examples:

`zypper wp firefox`

#### Utilities

##### Verify Dependencies

`zypper verify` or `zypper ve`

Note:

This is useful in cases of a broken system

##### Install New Recommended Packages

`zypper install-new-recommends` or `zypper inr`

#### Applying Patches

`zypper patch`

#### Listing All Patches

`zypper patches`

#### Checking Patches

`zypper patch-check` or `zypper pchk`

#### Getting Information About Patches

`zypper patch-info`

`zypper info -t patch`

#### Packages Updates

`zypper list-updates` or `zypper lu`

`zypper update` or `zypper up`

into .solv files (building the solv cache), typically under `/var/cache/zypp/solv/<alias>`.

#### Services

Services are one level above repositories and serve to manage repositories or to do some special tasks. Libzypp currently supports only one type of services, the Repository Index Service (RIS).

#### Package Types

zypper works with several types of resource objects, called resolvables. A resolvable is a package, patch, pattern, or a product.

package - an ordinary RPM package

patch - update of one or more packages.

pattern - group of packages required or recommended to install some functionality

product - group of packages which are necessary to install a product

# Firewalld

Wednesday, December 2, 2020 3:56 PM

## Useful firewall-cmd Examples

### 1. List all zones

Use the following command to list information for all zones. Only partial output is displayed.

```
# firewall-cmd --list-all-zones
work
target: default
icmp-block-inversion: no
interfaces:
sources:
services: dhcpcv6-client ssh
ports:
protocols:
masquerade: no
forward-ports:
sourceports:
icmp-blocks:
rich rules:
drop
target: DROP
icmp-block-inversion: no
interfaces:
sources:
services:
ports:
protocols:
masquerade: no
forward-ports:
sourceports:
icmp-blocks:
rich rules:
....
```

Public is the default zone set, if you do not change it. To check the currently set default zone use the below command:

```
# firewall-cmd --get-default-zone
public
```

### 2. List allowed service and ports on the system

To show currently allowed service on your system use the below command.

```
# firewall-cmd --list-services
dhcpcv6-client ssh
```

To list the ports that are open on your system:

```
# firewall-cmd --list-ports
```

You would normally see no ports listed here when you have just enabled the

firewalld.

### 3. To Enable all the incoming ports for a service

You can also open the required ports for a service by using the **--add-service** option.  
To permit access by HTTP clients for the public zone:

```
# firewall-cmd --zone=public --add-service=http  
success
```

To list services that are allowed for the public zone:

```
# firewall-cmd --zone=work --list-services  
dhcpcv6-client http ssh
```

Using this command only changes the Runtime configuration and does not update the configuration files. The following sequence of commands shows that configuration changes made in Runtime configuration mode are lost when the firewalld service is restarted:

```
# systemctl restart firewalld  
# firewall-cmd --zone=work --list-services  
dhcpcv6-client ssh
```

To make changes permanent, use the **--permanent** option. Example:

```
# firewall-cmd --permanent --zone=public --add-service=http  
success
```

Changes made in Permanent configuration mode are not implemented immediately.  
Example:

```
# firewall-cmd --zone=work --list-services  
dhcpcv6-client ssh
```

However, changes made in a Permanent configuration are written to configuration files. Restarting the firewalld service reads the configuration files and implements the changes.

Example:

```
# systemctl restart firewalld  
# firewall-cmd --zone=work --list-services  
dhcpcv6-client http ssh
```

### 4. Allow traffic on an incoming port

The command below will open the port 2222 effective immediately, but will not persist across reboots:

```
# firewall-cmd --add-port=[YOUR PORT]/tcp  
For example, to open TCP port 2222 :
```

```
# firewall-cmd --add-port=2222/tcp
```

The following command will create a persistent rule, but will not be put into effect immediately:

```
# firewall-cmd --permanent --add-port=[YOUR PORT]/tcp  
For Example, to open TCP port 2222 :
```

```
# firewall-cmd --permanent --add-port=2222/tcp  
To list the open ports, use the command :
```

```
# firewall-cmd —list-ports  
2222/tcp
```

## 5. Start and stop firewalld service

To start/stop/status firewalld service use the below commands:

```
# systemctl start firewalld.service  
# systemctl stop firewalld.service  
To check the status of the firewalld service:
```

```
# systemctl status firewalld.service
```

From <<https://www.thegeekdiary.com/5-useful-examples-of-firewall-cmd-command/>>

# Zip

Tuesday, October 13, 2020 9:40 AM

## How to ZIP Files and Directories

```
zip archivename.zip filename1 filename2 filename3
```

To create a Zip archive of a directory you would use:

```
zip -r archivename.zip directory_name
```

You can also add multiple files and directories in the same archive:

```
zip -r archivename.zip directory_name1 directory_name2 file1 file1
```

### Compression Methods and Levels #

The default compression method of Zip is **deflate**. If the zip utility determines that a file cannot be compressed it simply stores the file in the archive without compressing it using the **store** method. In most Linux distributions the zip utility also supports the **bzip2** compression method.

To specify a compression method, use the **-Z** option.

```
zip -r -Z bzip2 archivename.zip directory_name
```

...

```
adding: sub_dir/ (stored 0%)
adding: sub_dir/file1 (bzipped 52%)
adding: sub_dir/file2 (bzipped 79%)
```

**The zip command allows you to specify a compression level using number prefixed with a dash from 0 to 9. The default compression level is -6. When using -0, all files will be stored without compression. -9 will force the zip command to use an optimal compression for all files.**

For example, to use the compression level -9, you would type something like this:

```
zip -9 -r archivename.zip directory_name
```

**The higher the compression level, the more CPU-intensive the zip process is, and it will take more time to complete.**

### Creating a Password Protected ZIP file

**If you have sensitive information that needs to be stored in the archive you can encrypt it using the **-e** option:**

```
zip -e archivename.zip directory_name
```

You will be prompted to enter and verify the archive password:

Enter password:

Verify password:

### Creating Split Zip File

Imagine you want to store the Zip archive on a file hosting service that has a file size upload limit of 1GB and your Zip archive is 5GB.

You can create a new split Zip file using the **-s** option followed by specified size. The multiplier can be k (kilobytes), m (megabytes), g (gigabytes), or t (terabytes).

```
zip -s 1g -r archivename.zip directory_name
```

The command above will keep creating new archives in a set after it reaches the specified size limit.

```
archivename.zip
archivename.z01
archivename.z02
archivename.z03
archivename.z04
```

## ZIP Examples

Create a Zip archive named archivename.zip containing all the files in the current directory.

`zip archivename *`

Same as above including the hidden files (files starting with a dot):

`zip archivename .*`

Create a Zip archive named archivename.zip containing all MP3 files in the current directory without compressing the files.

`zip -0 archivename *.mp3`

# Unzip

Tuesday, October 13, 2020 12:52 PM

## How to Unzip a ZIP File With the `unzip` Command

To extract the files from a ZIP file, use the `unzip` command, and provide the name of the ZIP file. Note that you *do* need to provide the “.zip” extension.

`unzip source_code.zip`

```
dave@howtogeek:~$ unzip source_code.zip
```

As the files are extracted they are listed to the terminal window.

```
inflating: work/constant.h
inflating: work/makefile
inflating: work/sl.c
inflating: work/getval.c
inflating: work/debug.c
inflating: work/string.c
inflating: work/sl.h
inflating: work/global.c
inflating: work/label.c
inflating: work/at.c
inflating: work/if.c
inflating: work/for.c
inflating: work/files.c
inflating: work/keyval.h
inflating: work/logic.c
inflating: work/interp.c
inflating: work/error.c
inflating: work/cursor.c
inflating: work/variable.c
inflating: work/math.c
```

```
dave@howtogeek:~$ █
```

ZIP files don't carry details of file ownership. All of the files that are extracted have the owner set to the user who is extracting them.

Just like `zip`, `unzip` has a `-q` (quiet) option, so that you do not need to see the file listing as the files are extracted.

`unzip -q source_code.zip`

```
dave@howtogeek:~$ unzip -q source_code.zip
```

```
dave@howtogeek:~$ █
```

## Extracting Files to a Target Directory

To have the files extracted in a specific directory, use the `-d` (directory) option, and provide the path to the directory you wish the archive to be extracted into.

`unzip -q source_code.zip -d ./development`

```
dave@howtogeek:~$ unzip -q source_code.zip -d ./development
```

```
dave@howtogeek:~$ █
```

# Extract Password Protected ZIP Files

If a ZIP file has been created with a password, `unzip` will ask you for the password. If you do not provide the correct password, `unzip` will not extract the files.

`unzip -q source_code.zip`

```
dave@howtogeek:~$ unzip -q source_code.zip  
[source_code.zip] work/archive/caps.c password:  
dave@howtogeek:~$ █
```

If you don't care about your password being seen by others—or about it being stored in your command history—you can provide the password on the command line with the `-P` (password) option. (You must use a capital "P".)

`unzip -P fifty.treacle.cutlass -q source_code.zip`

```
dave@howtogeek:~$ unzip -P fifty.treacle.cutlass -q source_code.zip  
dave@howtogeek:~$ █
```

## Excluding Files

If you do not want to extract a particular file or group of files, use the `-x` (exclude) option. In this example, we want to extract all of the files apart from those ending in a ".h" extension.

`unzip -q source_code.zip -x *.h`

```
dave@howtogeek:~$ unzip -q source_code.zip -x *.h  
dave@howtogeek:~$ █
```

## Overwriting Files

Suppose you have extracted an archive but you have deleted a few of the extracted files by mistake.

A quick fix for that would be to extract the files once again. But if you try to extract the ZIP file in the same directory as before, `unzip` will prompt you for a decision regarding overwriting the files. It will expect one of the following responses.

Apart from the `r` (rename) response, these responses are case sensitive.

- `y`: Yes, overwrite this file
- `n`: No, don't overwrite this file
- `A`: All, overwrite all of the files
- `N`: None, overwrite none of the files
- `r`: Rename, extract this file but give it a new name. You will be prompted for a new name.

```
dave@howtogeek:~$ unzip -q source_code.zip  
replace work/archive/caps.c? [y]es, [n]o, [A]ll, [N]one, [r]ename: █
```

To force `unzip` to overwrite any existing files use the `-o` (overwrite) option.

`unzip -o -q source_code.zip`

```
dave@howtogeek:~$ unzip -o -q source_code.zip  
dave@howtogeek:~$
```

The most efficient way to replace the missing files would be to have `unzip` only extract any files in the archive that are *not* in the target directory. To do this, use the `-n` (never overwrite) option.

```
unzip -n source_code.zip
```

```
dave@howtogeek:~$ unzip -n source_code.zip  
Archive: source_code.zip  
  inflating: work/rdescent.c  
  inflating: work/date.c  
  inflating: work/header.h  
dave@howtogeek:~$
```

## Looking Inside a ZIP File

It is often useful and instructive to see a list of the files inside a ZIP file before you extract it. You can do this with the `-l` (list archive) option. It is [piped](#) through `less` to make the output manageable.

```
unzip -l source_code.zip | less
```

```
dave@howtogeek:~$ unzip -l source_code.zip | less
```

The output shows the directories and files within the ZIP file, their length and the time and date they were added to the archive. Press “q” to quit from `less`.

```
Archive: source_code.zip  
      Length      Date    Time     Name  
-----  -----  
        0  2019-05-11 15:36  work/  
        0  2019-05-11 15:35  work/archive/  
    1800  2019-05-11 15:27  work/archive/caps.c  
   1660  2019-05-11 15:27  work/archive/do.c  
   3452  2019-05-11 15:27  work/archive/rdescent.c  
   8769  2019-05-11 15:27  work/archive/proc.c  
    820  2019-05-11 15:27  work/archive/structs.h  
    225  2019-05-11 15:27  work/archive/system.c  
    684  2019-05-11 15:27  work/archive/goto.c  
   2924  2019-05-11 15:27  work/archive/while.c  
   2515  2019-05-11 15:27  work/archive/date.c  
 119376  2019-05-11 15:27  work/archive/slang.c  
   7936  2019-05-11 15:27  work/archive/param.c  
  73070  2019-05-11 15:35  work/archive/.zip  
   7765  2019-05-11 15:27  work/archive/print.c  
   2601  2019-05-11 15:27  work/archive/julian.c  
   6090  2019-05-11 15:27  work/archive/header.h  
:
```

There are other ways to peek inside a ZIP file which give different types of information, as we shall see.

## Add a Password With the `zipcloak` Command

If you've created a ZIP file but forgot to add a password, what can you do? You can

quickly add a password to ZIP file using the zipcloak command. Pass the name of the ZIP file on the command line. You will be prompted for a password. You need to verify the password by entering it a second time.

```
zipcloak source_code.zip
```

```
dave@howtogeek:~$ zipcloak source_code.zip  
Enter password:  
Verify password: ■
```

## View File Details With the zipdetails Command

The zipdetails command will show you a *lot* of information regarding the ZIP file. The only sensible way to handle the amount of output this command can give is to pipe it through less .

```
zipdetails source_code.zip | less
```

```
dave@howtogeek:~$ zipdetails source_code.zip | less■
```

Note that the information will include filenames even if the ZIP file is password protected. This type of information is stored within the ZIP file as meta-data and is not part of the encrypted data.

```
0064B Uncompressed Length      00000D7C  
0064F Filename Length         0017  
00651 Extra Length           001C  
00653 Filename               'work/archive/rdescent.c'  
0066A Extra ID #0001          5455 'UT: Extended Timestamp'  
0066C Length                 0009  
0066E Flags                  '03 mod access'  
0066F Mod Time               5CD72231 'Sat May 11 15:27:45 2019'  
00673 Access Time             5CD72B13 'Sat May 11 16:05:39 2019'  
00677 Extra ID #0002          7875 'ux: Unix Extra Type 3'  
00679 Length                 000B  
0067B Version                01  
0067C UID Size               04  
0067D UID                    000003E8  
00681 GID Size               04  
00682 GID                    000003E8  
00686 PAYLOAD  
  
00A85 LOCAL HEADER #6        04034B50  
00A89 Extract Zip Spec       14 '2.0'  
:■
```

## Search Inside the File With the zipgrep Command

The zipgrep command allows you to search *within the files* in a ZIP file. In the following example, we want to know which files within the ZIP file have the text “keyval.h” in them.

```
zipgrep keyval.h source_code.zip
```

```
dave@howtogeek:~$ zipgrep keyval.h source_code.zip
work/archive/slang.c:#include "keyval.h"
work/archive/.zip:Binary file (standard input) matches
work/archive/getval.c:#include "keyval.h"
work/slang.c:#include "keyval.h"
work/getval.c:#include "keyval.h"
dave@howtogeek:~$ █
```

We can see that the files `slang.c` and `getval.c` contain the string “`keyval.h`”. We can also see that there are two copies of each of these files in different directories in the ZIP file.

## View Information With the `zipinfo` Command

The `zipinfo` command gives you yet another way to look inside a ZIP file. As before, we pipe the output through `less`.

```
zipinfo source_code.zip | less
```

```
dave@howtogeek:~$ zipinfo source_code.zip | less
```

From left to right the output shows:

- The file permissions
- The version of the tool used to create the ZIP file
- The original file size
- A file descriptor (described below)
- The method of compression (deflation, in this case)
- The data and time stamp
- The name of the file and any directory

The file descriptor is made up of two characters. The first character will be a “`t`” or a “`b`” to indicate a text or binary file. If it is a capital letter the file is encrypted. The second character may be one of four characters. This character represents what type of meta-data is included for this file: none, an extended local header, an “extra field”, or both.

- -: If neither exists, the character will be a hyphen
- l: if there is an extended local header but no extra field
- x: if there is no extended local header but there is an extra field
- X: if there is an extended local header and there is an extra field

```
-rw-rw-r-- 3.0 unx      684 tx defN 96-Jul-17 18:55 work/goto.c
-rw-rw-r-- 3.0 unx     2924 tx defN 96-Jul-27 18:18 work/while.c
-rw-rw-r-- 3.0 unx     2515 tx defN 13-Jul-31 16:57 work/date.c
-rw-rw-r-- 3.0 unx   119376 tx defN 15-Dec-13 07:13 work/slang.c
-rw-rw-r-- 3.0 unx     7936 tx defN 96-Aug-04 15:50 work/param.c
-rw-rw-r-- 3.0 unx     7765 tx defN 96-Nov-01 15:09 work/print.c
-rw-rw-r-- 3.0 unx     2601 tx defN 96-Jul-26 16:48 work/julian.c
-rw-rw-r-- 3.0 unx     6090 tx defN 13-Aug-04 05:59 work/header.h
-rw-rw-r-- 3.0 unx     4260 tx defN 13-Aug-04 05:59 work/constant.h
-rw-rw-r-- 3.0 unx     1 394 tx defN 13-Jul-27 16:46 work/makefile
-rw-rw-r-- 3.0 unx     6907 tx defN 13-Jul-28 11:53 work/sl.c
-rw-rw-r-- 3.0 unx     7614 tx defN 96-Jul-26 18:00 work/getval.c
-rw-rw-r-- 3.0 unx     718 tx defN 96-Jul-17 18:55 work/debug.c
-rw-rw-r-- 3.0 unx     5459 tx defN 96-Jul-27 18:18 work/string.c
-rw-rw-r-- 3.0 unx    12975 tx defN 13-Aug-04 05:59 work/sl.h
-rw-rw-r-- 3.0 unx     770 tx defN 96-Jul-17 18:55 work/global.c
-rw-rw-r-- 3.0 unx     1794 tx defN 96-Jul-27 18:18 work/label.c
-rw-rw-r-- 3.0 unx     4382 tx defN 13-Jul-29 17:01 work/at.c
-rw-rw-r-- 3.0 unx     2058 tx defN 96-Jul-17 18:55 work/if.c
-rw-rw-r-- 3.0 unx     6245 tx defN 96-Jul-27 18:18 work/for.c
:
```

█

## Split the File With the zipsplit Command

If you need to send the ZIP file to someone else but there are size restrictions or problems with the transmission of the file, you can use the `zipsplit` command to split the original ZIP file into a set of smaller ZIP files.

The `-n (size)` option allows you to set a maximum size for each of the new ZIP files. In this example, we're splitting the `source_code.zip` file. We don't want any of the new ZIP files to be bigger than 100 KB (102400 bytes).

```
zipsplit -n 102400 source_code.zip
```

```
dave@howtogeek:~$ zipsplit -n 102400 source_code.zip
3 zip files will be made (100% efficiency)
creating: source_1.zip
creating: source_2.zip
creating: source_3.zip
dave@howtogeek:~$ 
dave@howtogeek:~$ ls -lh source_?.*
-rw-r--r-- 1 dave dave 37K May 11 16:39 source_1.zip
-rw-r--r-- 1 dave dave 100K May 11 16:39 source_2.zip
-rw-r--r-- 1 dave dave 80K May 11 16:39 source_3.zip
dave@howtogeek:~$ █
```

The size that you choose cannot be smaller than the size of any of the files in the ZIP file.

Using these commands, you can create your own ZIP files, unzip ZIP files you receive, and perform various other operations on them without ever leaving the Linux terminal.

# Mem Check

Tuesday, July 14, 2020 8:42 AM

```
p54adm@pepldh00112:/usr/sap/P54/HDB00> free -h; free | awk 'FNR == 3 {print "Memory utilization: " $3/($3+$4)*100 "%" } ; free | awk 'FNR == 4 {print "Swap utilization: " $3/($2)*100 "%" }'
      total    used    free   shared  buffers   cached
Mem:     3.0T    2.9T    10G    13G   338M    123G
-/+ buffers/cache:   2.8T    134G
Swap:    32G     32G     0B
Memory utilization: 95.5445%
Swap utilization: 100%
```

```
qb1adm@pepldh00103:/usr/sap/QB1/HDB20> uname -a;w;date; free -h; free | grep Mem | awk '{print "Memory Utilization is " $3/$2 * 100.0}'; free | grep Swap | awk '{print "Swap Utilization is " $3/$2 * 100.0}'
Linux pepldh00103 4.4.180-94.113-default #1 SMP Fri Dec 13 14:20:57 UTC 2019 (c6649f6) x86_64 x86_64 x86_64 GNU/Linux
00:01:10 up 235 days, 19:14, 2 users, load average: 0.71, 0.70, 0.87
USER   TTY   FROM         LOGIN@ IDLE JCPU PCPU WHAT
pb1adm pts/0  30.25.231.241 17:21  6:39m 0.05s 0.05s -sh
qb1adm pts/1  30.25.231.241 00:00  3.00s 0.06s 0.02s w
Fri Oct 16 00:01:10 CDT 2020
      total    used    free   shared  buffers   cached
Mem:     3.0T    2.9T    22G    1.1G   193M    2.6G
-/+ buffers/cache:   2.9T    24G
Swap:    32G     6.9G    25G
Memory Utilization is 99.2659
Swap Utilization is 21.5436
```

## How to cause kernel panic with a single command?

Wednesday, September 16, 2020 10:00 AM

```
echo c > /proc/sysrq-trigger
```

From <<https://unix.stackexchange.com/questions/66197/how-to-cause-kernel-panic-with-a-single-command>>

# Linux Command Syntex

Ahad, 31 Mei 2020 11:17 PTG

## Command options and arguments

- Commands typically have the syntax:

Options

Arguments

### **Options:**

Modify the way that a command works

Usually consist of hyphen or dash followed by single letter.

Some commands accept multiple options which can usually be grouped together after a single hyphen.

### **Arguments:**

Most commands are used together with one or more arguments

Some commands assume a default arguments if none is supplied

Arguments are optional for some cmds and required by others.

```
# Whoami  
# pwd  
# ls -l  
# man ls
```

# System Journal Logs

Wednesday, December 2, 2020 3:00 PM

## Lesson 4 Configuring system logging

---

\* Configuring system logging

who is logging

- Services
- syslog
- systemd-journald

/var/log will contain all the information you need. depends on the config behind the services

systemd-journald feeds syslog so syslog is centralized element that drives all the information to /var/log dir

syslog loggin in system using by facilities. (news, cron, kern many more)

you can configure many services to log in syslog.syslog can also do remote login.

Syslog services

handle logging by facilities.priorities and sending to destination.

Ex: kern:crit \* (every user current login in system)

syslog-*ng* (next generation but replaced by these days with rsyslog) can add modules.

# nano /etc/rsyslog.conf (rsyslog configuration file)

---

\* Configure secure remote logging creating the CA.

- time sync
- TLS certificates (certtool)
- 6514 tcp port (log client)
- gtls driver on client

# cd /usr/share/doc/rsyslog-8.24.0/html/ (contains all the content of packages that we just installed)

# systemctl status crond (Time server on RHEL7)

# certtool (allow you to create certificate)

# certtool --generate-prvkey --outfile ca-key.pem

# ls

# chmod 400 ca-key.pem (for root only)

# certtool --generate-self-signed --load-prvkey ca-key.pem --outfile ca.pem

# certtool --generate-prvkey --outfile server1-key.pem --bits 2048

# certtool --generate-request --load-prvkey server1-key.pem outfile server1-request.pem

# certtool --generate-certificate --load-request server1-request.pem --outfile server1-cert.pem --load-ca-certificate ca.pem --load-ca-prvkey ca-key.pem

# Kernel Security

Wednesday, December 2, 2020 3:01 PM

## Lesson 7 Managing kernel security

---

\* Understanding the kernel security architecture.

Linux came from Unix and using same guidance principle as Unix.

It has not been designed as secure kernel from default.

How linux kernel operate There are some security considerations.

1. ring0: Kernel space

Linux doesn't implement ring 1

2. ring3: which is user space.

If user starts program, these programs are running in user space. program will use restricted amount of memory.

If program has to do something special on kernel level. will use "syscall" (System Call)

if program wants to go another program Then you required -

IPC (Inter process communication)

mmap (mmap system call) allows program to communicate to another program.

chroot - Runs process in fake root environment - max isolate between those programs

cgroup - hardware resource available for this processes

containers - dockers or linux container

---

### \* Managing kernel security

Risk in kernel

- buffer overflow

when application starts it reserves buffer memory

restricted access to memory.

patch it - applying patches really important

Privilege escalation - User getting more privilege than he was supposed to have (Kernel bug)

suid permission / su / sudo dangerous

Rootkit - root fake programs that can handover complete program to someone else on system

run FS integrity check - aide

no kernel modules

---

### \* Fixing Linux kernel vulnerabilities

```
# lsmod (kernel modules)
# lsmod | grep ext
# lsmod | grep btrfs
# modprobe btrfs (will load btrfs module in kernel)
# modprobe -r btrfs (unload the module from kernel)
# cd /proc/sys/kernel/ (modules for kernel parameters)
# cat modules_disabled
0
# echo 1 > modules_disabled
# cat randomize_va_space (secure)
2
# grep nx /proc/cpuinfo
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr
sse sse2 syscall nx rdtscp lm constant_tsc rep_good nopl xtopology nonstop_tsc eagerfpu dni pclmulqdq
```

```
monitor ssse3 cx16 pcid sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx rdrand hypervisorlahf_lm  
abm 3dnowprefetch fsgsbase avx2 invpcid rdseed clflushopt  
nx (stands for no execute)
```

# By Sandeep

Monday, January 25, 2021 8:25 PM

LVM chunk,  
LVM backup, Archive, Extent  
Floating IP, VIP, Check bandwith, VLAN tagging,

Multipath -

LVM filter

# /etc/passwd - Understand

Thursday, February 18, 2021 12:13 PM

## Understanding /etc/passwd file fields

The /etc/passwd contains one entry per line for each user (user account) of the system. All fields are separated by a colon (:) symbol. Total of seven fields as follows. Generally, /etc/passwd file entry looks as follows:

The diagram shows a line of text: "oracle:x:1021:1020:Oracle user:/data/network/oracle:/bin/bash". Seven vertical arrows point downwards from the colon-separated fields to the numbers 1 through 7 respectively. The fields are: 1. Username (oracle), 2. Password (x), 3. User ID (UID) (1021), 4. Group ID (GID) (1020), 5. User ID Info (Oracle user), 6. Home directory (/data/network/oracle), and 7. Command/shell (/bin/bash).

(Fig.01: /etc/passwd file format – click to enlarge)

### /etc/passwd Format

From the above image:

1. **Username:** It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password:** An x character indicates that encrypted password is stored in /etc/shadow file. Please note that you need to use the passwd command to compute the hash of a password typed at the CLI or to store/update the hash of the password in /etc/shadow file.
3. **User ID (UID):** Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID):** The primary group ID (stored in /etc/group file)
5. **User ID Info:** The comment field. It allows you to add extra information about the users such as user's full name, phone number etc. This field is used by finger command.
6. **Home directory:** The absolute path to the directory the user will be in when they log in. If this directory does not exist then user's directory becomes /
7. **Command/shell:** The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell.

For example, sysadmin can use the nologin shell, which acts as a replacement shell for the user accounts. If shell set to `/sbin/nologin` and the user tries to log in to the Linux system directly, the `/sbin/nologin` shell closes the connection.

From <<https://www.cyberciti.biz/faq/understanding-etcpassword-file-format/>>

# Ubuntu Cheats

Wednesday, February 24, 2021 12:07 PM

# Ubuntu Reference

# FOSSwire

Privileges	Network
<b>sudo command</b> - run <i>command</i> as root <b>sudo su</b> - open a root shell <b>sudo su user</b> - open a shell as <i>user</i> <b>sudo -k</b> - forget sudo passwords <b>gksudo command</b> - visual sudo dialog (GNOME) <b>kdesudo command</b> - visual sudo dialog (KDE) <b>sudo visudo</b> - edit /etc/sudoers <b>gksudo nautilus</b> - root file manager (GNOME) <b>kdesudo konqueror</b> - root file manager (KDE) <b>passwd</b> - change your password	<b>ifconfig</b> - show network information <b>iwconfig</b> - show wireless information <b>sudo iwlist scan</b> - scan for wireless networks <b>sudo /etc/init.d/networking restart</b> - reset network (file) <b>/etc/network/interfaces</b> - manual configuration <b>ifup interface</b> - bring <i>interface</i> online <b>ifdown interface</b> - disable <i>interface</i>
Display	Special Packages
<b>sudo /etc/init.d/gdm restart</b> - restart X (GNOME) <b>sudo /etc/init.d/kdm restart</b> - restart X (KDE) (file) <b>/etc/X11/xorg.conf</b> - display configuration <b>sudo dpkg-reconfigure -phigh xserver-xorg</b> - reset X configuration <b>Ctrl+Alt+Bksp</b> - restart X display if frozen <b>Ctrl+Alt+FN</b> - switch to tty <i>N</i> <b>Ctrl+Alt+F7</b> - switch back to X display	<b>ubuntu-desktop</b> - standard Ubuntu environment <b>kubuntu-desktop</b> - KDE desktop <b>xubuntu-desktop</b> - XFCE desktop <b>ubuntu-minimal</b> - core Ubuntu utilities <b>ubuntu-standard</b> - standard Ubuntu utilities <b>ubuntu-restricted-extras</b> - non-free, but useful <b>kubuntu-restricted-extras</b> - KDE of the above <b>xubuntu-restricted-extras</b> - XFCE of the above <b>build-essential</b> - packages used to compile programs <b>linux-image-generic</b> - latest generic kernel image <b>linux-headers-generic</b> - latest build headers
System Services <sup>1</sup>	Firewall <sup>1</sup>
<b>start service</b> - start job service (Upstart) <b>stop service</b> - stop job service (Upstart) <b>status service</b> - check if service is running (Upstart) <b>/etc/init.d/service start</b> - start service (SysV) <b>/etc/init.d/service stop</b> - stop service (SysV) <b>/etc/init.d/service status</b> - check service (SysV) <b>/etc/init.d/service restart</b> - restart service (SysV) <b>runlevel</b> - get current runlevel	<b>ufw enable</b> - turn on the firewall <b>ufw disable</b> - turn off the firewall <b>ufw default allow</b> - allow all connections by default <b>ufw default deny</b> - drop all connections by default <b>ufw status</b> - current status and rules <b>ufw allow port</b> - allow traffic on <i>port</i> <b>ufw deny port</b> - block <i>port</i> <b>ufw deny from ip</b> - block <i>ip</i> address
Package Management <sup>1</sup>	Application Names
<b>apt-get update</b> - refresh available updates <b>apt-get upgrade</b> - upgrade all packages <b>apt-get dist-upgrade</b> - upgrade Ubuntu version <b>apt-get install pkg</b> - install <i>pkg</i> <b>apt-get remove pkg</b> - uninstall <i>pkg</i> <b>apt-get autoremove</b> - remove obsolete packages <b>apt-get -f install</b> - try to fix broken packages <b>dpkg --configure -a</b> - try to fix broken packages <b>dpkg -i pkg.deb</b> - install file <i>pkg.deb</i> (file) <b>/etc/apt/sources.list</b> - APT repository list	<b>nautilus</b> - file manager (GNOME) <b>dolphin</b> - file manager (KDE) <b>konqueror</b> - web browser (KDE) <b>kate</b> - text editor (KDE) <b>gedit</b> - text editor (GNOME)
System	
	<b>Recovery</b> - Type the phrase "REISUB" while holding down Alt and SysRq (PrintScrn) with about 1 second between each letter. Your system will reboot. <b>lsb_release -a</b> - get Ubuntu version <b>uname -r</b> - get kernel version <b>uname -a</b> - get all kernel information

1. Prefix commands with sudo to run.

Ubuntu is a trademark of Canonical Ltd. Licensed under CC-BY-SA 3.0. Free to redistribute; see creativecommons.org for details.

# Wget

Monday, April 19, 2021 3:20 PM

```
$ wget --user=vivek --password='myPassword' http://theos.in/protected/area/foo.pdf
OR
wget --user=vivek --ask-password http://192.168.1.10/docs/foo.pdf
```

# RPM

Monday, April 19, 2021 3:21 PM

## 1. How to Check an RPM Signature Package

Always check the PGP signature of packages before installing them on your Linux systems and make sure its integrity and origin is **OK**. Use the following command with **--checksig (check signature)** option to check the signature of a package called **pidgin**.

```
[root@tecmint]# rpm --checksig pidgin-2.7.9-5.el6.2.i686.rpm
```

```
pidgin-2.7.9-5.el6.2.i686.rpm: rsa sha1 (md5) pgp md5 OK
```

## 2. How to Install an RPM Package

For installing an rpm software package, use the following command with **-i** option. For example, to install an rpm package called **pidgin-2.7.9-5.el6.2.i686.rpm**.

```
[root@tecmint]# rpm -ivh pidgin-2.7.9-5.el6.2.i686.rpm
```

```
Preparing...      ##### [100%]
1:pidgin        ##### [100%]
```

### RPM command and options

- **-i** : install a package
- **-v** : verbose for a nicer display
- **-h**: print hash marks as the package archive is unpacked.

## 3. How to check dependencies of RPM Package before Installing

Let's say you would like to do a dependency check before installing or upgrading a package. For example, use the following command to check the dependencies of **BitTorrent-5.2.2-1-Python2.4.noarch.rpm** package. It will display the list of dependencies of package.

```
[root@tecmint]# rpm -qpR BitTorrent-5.2.2-1-Python2.4.noarch.rpm
```

```
/usr/bin/python2.4
python >= 2.3
python(abi) = 2.4
python-crypto >= 2.0
```

```
python-psycod
python-twisted >= 2.0
python-zopeinterface
rpmlib(CompressedFileNames) = 2.6
```

## RPM command and options

- **-q** : Query a package
- **-p** : List capabilities this package provides.
- **-R**: List capabilities on which this package depends..

## 4. How to Install a RPM Package Without Dependencies

If you know that all needed packages are already installed and RPM is just being stupid, you can ignore those dependencies by using the option **--nodeps (no dependencies check)** before installing the package.

```
[root@tecmint]# rpm -ivh --nodeps BitTorrent-5.2.2-1-Python2.4.noarch.rpm
```

```
Preparing...      ##### [100%]
1:BitTorrent      ##### [100%]
```

The above command forcefully install rpm package by ignoring dependencies errors, but if those dependency files are missing, then the program will not work at all, until you install them.

## 5. How to check an Installed RPM Package

Using **-q** option with package name, will show whether an rpm installed or not.

```
[root@tecmint]# rpm -q BitTorrent
```

```
BitTorrent-5.2.2-1.noarch
```

## 6. How to List all files of an installed RPM package

To view all the files of an installed rpm packages, use the **-ql (query list)** with rpm command.

```
[root@tecmint]# rpm -ql BitTorrent
```

```
/usr/bin/bittorrent
/usr/bin/bittorrent-console
/usr/bin/bittorrent-curses
```

```
/usr/bin/bittorrent-tracker  
/usr/bin/changetracker-console  
/usr/bin/launchmany-console  
/usr/bin/launchmany-curses  
/usr/bin/maketorrent  
/usr/bin/maketorrent-console  
/usr/bin/torrentinfo-console
```

## 7. How to List Recently Installed RPM Packages

Use the following rpm command with **-qa (query all)** option, will list all the recently installed rpm packages.

```
[root@tecmint]# rpm -qa --last
```

```
BitTorrent-5.2.2-1.noarch           Tue 04 Dec 2012 05:14:06 PM BDT  
pidgin-2.7.9-5.el6.2.i686          Tue 04 Dec 2012 05:13:51 PM BDT  
cyrus-sasl-devel-2.1.23-13.el6_3.1.i686      Tue 04 Dec 2012 04:43:06 PM BDT  
cyrus-sasl-2.1.23-13.el6_3.1.i686      Tue 04 Dec 2012 04:43:05 PM BDT  
cyrus-sasl-md5-2.1.23-13.el6_3.1.i686     Tue 04 Dec 2012 04:43:04 PM BDT  
cyrus-sasl-plain-2.1.23-13.el6_3.1.i686    Tue 04 Dec 2012 04:43:03 PM BDT
```

## 8. How to List All Installed RPM Packages

Type the following command to print the all the names of installed packages on your Linux system.

```
[root@tecmint]# rpm -qa
```

```
initscripts-9.03.31-2.el6.centos.i686  
polkit-desktop-policy-0.96-2.el6_0.1.noarch  
thunderbird-17.0-1.el6.remi.i686
```

## 9. How to Upgrade a RPM Package

If we want to upgrade any RPM package “**-U**” (**upgrade**) option will be used. One of the major advantages of using this option is that it will not only upgrade the latest version of any package, but it will also maintain the backup of the older package so that in case if the newer upgraded package does not run the previously installed package can be used again.

```
[root@tecmint]# rpm -Uvh nx-3.5.0-2.el6.centos.i686.rpm  
Preparing...          ##### [100%]  
1:nx                ##### [100%]
```

## 10. How to Remove a RPM Package

To un-install an RPM package, for example we use the package name **nx**, not the original package name **nx-3.5.0-2.el6.centos.i686.rpm**. The **-e (erase)** option is used to remove package.

```
[root@tecmint]# rpm -evv nx
```

## 11. How to Remove an RPM Package Without Dependencies

The **--nodeps (Do not check dependencies)** option forcefully remove the rpm package from the system. But keep in mind removing particular package may break other working applications.

```
[root@tecmint]# rpm -ev --nodeps vsftpd
```

## 12. How to Query a file that belongs which RPM Package

Let's say, you have list of files and you would like to find out which package belongs to these files. For example, the following command with **-qf (query file)** option will show you a file **/usr/bin/htpasswd** is own by package **httpd-tools-2.2.15-15.el6.centos.1.i686**.

```
[root@tecmint]# rpm -qf /usr/bin/htpasswd
```

```
httpd-tools-2.2.15-15.el6.centos.1.i686
```

## 13. How to Query a Information of Installed RPM Package

Let's say you have installed an rpm package and want to know the information about the package. The following **-qi (query info)** option will print the available information of the installed package.

```
[root@tecmint]# rpm -qi vsftpd
```

```
Name      : vsftpd                                Relocations: (not relocatable)
Version   : 2.2.2                                    Vendor: CentOS
Release   : 11.el6                                   Build Date: Fri 22 Jun 2012 01:54:24 PM BDT
Install Date: Mon 17 Sep 2012 07:55:28 PM BDT       Build Host: c6b8.bsys.dev.centos.org
Group     : System Environment/Daemons           Source RPM: vsftpd-2.2.2-11.el6.src.rpm
Size      : 351932                                  License: GPLv2 with exceptions
Signature : RSA/SHA1, Mon 25 Jun 2012 04:07:34 AM BDT, Key ID 0946fca2c105b9de
Packager  : CentOS BuildSystem <http://bugs.centos.org>
URL      : http://vsftpd.beasts.org/
```

Summary : Very Secure Ftp Daemon

Description :

vsftpd is a Very Secure FTP daemon. It was written completely from scratch.

## 14. Get the Information of RPM Package Before Installing

You have download a package from the internet and want to know the information of a package before installing. For example, the following option **-qip (query info package)** will print the information of a package [sqlbuddy](#).

```
[root@tecmint]# rpm -qip sqlbuddy-1.3.3-1.noarch.rpm
```

```
Name      : sqlbuddy          Relocations: (not relocatable)
Version   : 1.3.3             Vendor: (none)
Release   : 1                 Build Date: Wed 02 Nov 2011 11:01:21 PM BDT
Install Date: (not installed) Build Host: rpm.bar.baz
Group     : Applications/Internet    Source RPM: sqlbuddy-1.3.3-1.src.rpm
Size      : 1155804            License: MIT
Signature  : (none)
Packager   : Erik M Jacobs
URL       : http://www.sqlbuddy.com/
Summary   : SQL Buddy â Web based MySQL administration
Description :
SQLBuddy is a PHP script that allows for web-based MySQL administration.
```

## 15. How to Query documentation of Installed RPM Package

To get the list of available documentation of an installed package, use the following command with option **-qdf (query document file)** will display the manual pages related to [vmstat](#) package.

```
[root@tecmint]# rpm -qdf /usr/bin/vmstat
```

```
/usr/share/doc/procps-3.2.8/BUGS
/usr/share/doc/procps-3.2.8/COPYING
/usr/share/doc/procps-3.2.8/COPYING.LIB
/usr/share/doc/procps-3.2.8/FAQ
/usr/share/doc/procps-3.2.8/NEWS
/usr/share/doc/procps-3.2.8/TODO
```

## 16. How to Verify a RPM Package

Verifying a package compares information of installed files of the package against the rpm database. The **-Vp (verify package)** is used to verify a package.

```
[root@tecmint downloads]# rpm -Vp sqlbuddy-1.3.3-1.noarch.rpm
```

```
S.5....T. c /etc/httpd/conf.d/sqlbuddy.conf
```

## 17. How to Verify all RPM Packages

Type the following command to verify all the installed rpm packages.

```
[root@tecmint]# rpm -Va
```

```
S.5....T. c /etc/rc.d/rc.local
```

```
.....T. c /etc/dnsmasq.conf
```

```
.....T. /etc/ld.so.conf.d/kernel-2.6.32-279.5.2.el6.i686.conf
```

```
S.5....T. c /etc/yum.conf
```

```
S.5....T. c /etc/yum.repos.d/epel.repo
```

## 18. How to Import an RPM GPG key

To verify **RHEL/CentOS/Fedora** packages, you must import the **GPG** key. To do so, execute the following command. It will import **CentOS 6** GPG key.

```
[root@tecmint]# rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6
```

## 19. How to List all Imported RPM GPG keys

To print all the imported **GPG** keys in your system, use the following command.

```
[root@tecmint]# rpm -qa gpg-pubkey*
```

```
gpg-pubkey-0608b895-4bd22942  
gpg-pubkey-7fac5991-4615767f  
gpg-pubkey-0f2672c8-4cd950ee  
gpg-pubkey-c105b9de-4e0fd3a3  
gpg-pubkey-00f97f56-467e318a  
gpg-pubkey-6b8d79e6-3f49313d  
gpg-pubkey-849c449f-4cb9df30
```

## 20. How To rebuild Corrupted RPM Database

Sometimes rpm database gets corrupted and stops all the functionality of rpm and other applications on the system. So, at the time we need to rebuild the rpm

database and restore it with the help of following command.

```
[root@tecmint]# cd /var/lib  
[root@tecmint]# rm __db*  
[root@tecmint]# rpm --rebuilddb  
[root@tecmint]# rpmdb_verify Packages
```

From <<https://www.tecmint.com/20-practical-examples-of-rpm-commands-in-linux/>>

## Introduction

Saturday, September 11, 2021 5:27 PM

[Ansible](#) is a modern configuration management tool that facilitates the task of [setting up and maintaining remote servers](#).

- **Control Machine / Node:** a system where Ansible is installed and configured to connect and execute commands on nodes.
- **Node:** a server controlled by Ansible.
- **Inventory File:** a file that contains information about the servers Ansible controls, typically located at [/etc/ansible/hosts](#).
- **Playbook:** a file containing a series of tasks to be executed on a remote server.
- **Role:** a collection of playbooks and other files that are relevant to a goal such as installing a web server.
- **Play:** a full Ansible run. A *play* can have several playbooks and roles, included from a single playbook that acts as entry point.

Ansible is a combination of multiple pieces working together to become an automation tool.

- **Modules** are small codes that will get executed. There are multiple built-in modules that serve as a starting point for building tasks.
- **Playbooks** contain plays which further is a group of tasks. This is the place to define the workflow or the steps needed to complete a process
- **Plugins** are special kinds of modules that run on the main control machine for logging purposes. There are other types of plugins also.

### Playbook

The playbooks ran via an Ansible automation engine.

It contains modules that are basically actions that run in host machines.

The mechanism followed here is the push mechanism, so ansible pushes small programs to these host machines which are written to be resource models of the desired state of the system.

- **Agentless** – Unlike puppet or chef there is no software or agent managing the nodes.

• **Python** – Built on top of python which is very easy to learn and write scripts and one of the robust programming languages.

• **SSH** – Passwordless network authentication which makes it more secure and easy to set up.

• **Push architecture** – The core concept is to push multiple small codes to the configure and run the action on client nodes.

• **Setup** – This is very easy to set up with a very low learning curve and any open source so that anyone can get hands-on.

• **Manage Inventory** – Machines' addresses are stored in a simple text format and we can add different sources of truth to pull the list using plugins such as Openstack, Rackspace, etc.

### Ansible Galaxy

Galaxy is a repository of Ansible roles

can be directly dropped into playbooks for execution

distribution of packages containing roles, plugins, and modules also known as collection

the ansible-galaxy command implements similar to init, build, install

#### Roles:

```
1 ---  
2   - name: Ensure yum-utils is installed  
3     yum:  
4       update_cache: yes  
5       state: present  
6       name: "yum-utils"  
7  
8   - name: Run "yum update"  
9     yum:  
10    update_cache: yes  
11    name: "*"  
12  
13  - name: Check if reboot is required.  
14    command: needs-restarting -r  
15    failed_when: false  
16    register: reboot_required  
17    changed_when: false  
18  
19  - debug:  
20    msg: "Reboot is required!!!"  
21    when: reboot_required.rc != 0  
22  
23  - name: Reboot host(s).  
24    include_role:  
25      name: reboot_host  
26    when: reboot_required.rc != 0 and reboot_ok
```

#### Reboot

```
1 ---  
2  
3   - name: Rebooting host!  
4     shell: "sleep 5 && reboot"  
5     async: 1  
6     poll: 0  
7  
8   - name: Wait for the reboot...  
9     wait_for_connection:  
10    connect_timeout: 20  
11    sleep: 5  
12    delay: 5  
13    timeout: 300
```

#### Structure

 ibehren1 Merge pull request #5 from ibehren1/linted_code ...	
.github/workflows	Add test playbook run.
roles	Name reboot hosts roles.
Jenkinsfile	Add Jenkinsfile.
Readme.md	Update Readme.md.
hosts	Working version to share.
main.yml	Add support for yum based distro

#### Host

```
1 [Dev]  
2 host01.example.com  
3  
4 [Prod]  
5 host11.prod.example.com  
6 host12.prod.example.com
```

#### Main.yml

```
1 ---  
2   - hosts: all  
3     become: yes  
4     roles:  
5       - { role: apt_update, when: ( ansible_distribution == "Ubuntu" ) or  
6             ( ansible_distribution == "Debian" ) }  
7       - { role: yum_update, when: ( ansible_distribution == "CentOS" ) or  
8             ( ansible_distribution == "Amazon" ) or  
9             ( ansible_distribution == "RedHat" ) }  
10      vars:  
11        reboot_ok: false  
12  
13  # How to run  
14  # ansible-playbook -b -u <user> --private-key <path to key> -i hosts -l <host-group> main.yml [--extra-vars reboot_ok=true]
```

# How ansible works.

Tuesday, October 4, 2022 1:45 PM

Multiple pieces working together to become an automation tool.

Mainly these are **modules, playbooks, and plugins.**

- Module are small codes that serve as a starting point for building tasks.
  - Playbooks contain plays group of tasks and workflow or the steps needed to complete a process.
  - Plugins are special kinds of modules that run on the main control machine for logging purposes.
- 
- Playbooks ran via ansible automation engine
  - It contain modules that are basically actions that run in host machine.
  - It's push mechanism.

## Features

- Agentless
- Written in Python
- SSH Passwordless architecture
- Manage Inventory, Address are stored in simple text format and we can add different sources of truth to pull the list.

# Galaxy

Tuesday, 25 October 2022 9:09 AM

Repository of Ansible roles that can share among user.

Also used for distribution of packages containing roles, plugins and modules also known as collection.

# Inventory

Tuesday, 25 October 2022 9:16 AM

## Static Inventory:

List of managed hosts declared under a host group using either hostname or IP in plain text file.

## Dynamic Inventory:

Generated by script written in python or any other prog language.

```
plugin: aws_ec2
regions:
ap-south-1
filters:
tag:tagtype: testing
```

Now we can fetch using this command

```
ansible-inventory -i demo_aws_ec2.yaml -graph
```

# Vault

Tuesday, 25 October 2022 9:23 AM

Ansible vault is used to keep sensitive data such as passwords instead of placing it as plaintext in playbooks or roles. Any structured data file or any single value inside the YAML file can be encrypted by Ansible.

To encrypt a file

```
ansible-vault encrypt foo.yml bar.yml baz.yml
```

And similarly to decrypt

```
ansible-vault decrypt foo.yml bar.yml baz.yml
```

# Handlers

Tuesday, 25 October 2022 9:23 AM

Handlers are like special tasks which only run if the Task contains a “notify” directive.

Executed at the end

Typical used to start, reload restart and stop services

tasks:

```
- name: install nginx
  apt: pkg=nginx state=installed update_cache=true
  notify:
    - start nginx
handlers:
- name: start nginx
  service: name=nginx state=started
```

In the above example after installing NGINX we are starting the server using a `start nginx` handler.

# Ansible Cheat Sheet

Saturday, September 11, 2021 5:20 PM

## ANSIBLE CHEAT SHEET

Learn DevOps from experts at [edureka.co](#)

### What Is Ansible?

Ansible is a continuous deployment and configuration tool which provides large productivity gains to a wide variety of automation challenges.



### Ansible Architecture



### SSH Key Generation & Install Ansible

#### SSH Key Generation

Ansible uses SSH to communicate between the nodes.

```
#Setting Up SSH Command
$ sudo apt-get install openssh-server
#Generating SSH Key
$ ssh-keygen
#Copy the SSH Key on the Hosts
$ ssh-copy-id hostname
#Check the SSH Connection
$ ssh <nodeName>
```

#### Install Ansible

To install Ansible in Debian Linux, follow the following steps:

```
#Add Ansible repository
$ sudo apt-add-repository ppa:ansible/ansible
#Run the update command
$ sudo apt-get update
#Install Ansible package
$ sudo apt-get install ansible
#Check Ansible Version
$ ansible --version
```

### Ad-Hoc Commands

Ad-Hoc commands are quick commands which are used to perform the actions, that won't be saved for later.

#### Parallelism & Shell Commands

```
#To set up SSH agent
$ ssh-agent bash $ ssh-add ~/.ssh/id_rsa
#To use SSH with a password instead of keys, you can use --ask-pass (-K)
$ ansible europe -a "/sbin/reboot" -f 20
#To run /usr/bin/ansible from a user account, not the root
$ ansible europe -a "/usr/bin/foo" -u username
#To run commands through privilege escalation and not through user account
$ ansible europe -a "/usr/bin/foo" -u username --become [--ask-become-pass]
#If you are using password less method then use --ask-become-pass (-k) to interactively get the password to be used
#You can become a user, other than root by using --become-user
$ ansible europe -a "/usr/bin/foo" -u username --become --become-user otheruser [--ask-become-pass]
```

#### File Transfer

```
#Transfer a file directly to many servers
$ ansible europe -m copy -a "src=/etc/hosts dest=/tmp/hosts"
#To change the ownership and permissions on files
$ ansible webservers -m file -a "dest=/srv/foo/a.txt mode=600" $ ansible webservers -m file -a "dest=/srv/foo/b.txt mode=600 owner=example group=example"
#To create directories
$ ansible webservers -m file -a "dest=/path/to/c mode=755 owner=example group=example state=directory"
#To delete directories (recursively) and delete files
$ ansible webservers -m file -a "dest=/path/to/c state=absent"
```

#### Manage Packages

```
#To ensure that a package is installed, but doesn't get updated
$ ansible webservers -m apt -a "name=acme state=present"
#To ensure that a package is installed to a specific version
$ ansible webservers -m apt -a "name=acme-1.5 state=present"
#To ensure that a package at the latest version
$ ansible webservers -m apt -a "name=acme state=latest"
#To ensure that a package is not installed
$ ansible webservers -m apt -a "name=acme state=absent"
```

#### Manage Services

```
#To ensure a service is started on all web servers
$ ansible webservers -m service -a "name=httpd state=started"
#To restart a service on all web servers
$ ansible webservers -m service -a "name=httpd state=restarted"
#To ensure a service is stopped
$ ansible webservers -m service -a "name=httpd state=stopped"
```

#### Deploying From Source Control

```
#GitRep:https://foo.example.org/repo.git          #Destination:/src/myapp
$ ansible webservers -m git -a "repo=https://foo.example.org/repo.git dest=/src/myapp version=HEAD"
```

### Inventory Files & Hosts Patterns

Ansible's inventory lists all the platforms you want to automate across. Ansible can at a single instance work on multiple hosts in the infrastructure.

#### Setup & Hosts Connection

Follow the below steps to set hosts and then check their connection.

```
#Set up hosts by editing the hosts' file in the Ansible directory
$ sudo nano /etc/ansible/hosts
#To check the connection to hosts
#First change the directory to /etc/Ansible
$ cd /etc/ansible
#To check whether Ansible is connecting to hosts, use ping command
$ ansible -m ping <hosts>
#To check on servers individually
$ ansible -m ping server_name
#To check a particular server group
$ ansible -m ping servergroupname
```

#### Ansible Hosts Patterns

##### Ansible Hosts Patterns

all	All hosts in inventory
*	All hosts in inventory
ungrouped	All hosts in inventory not appearing within a group
10.0.0.*	All hosts with an IP starting 10.0.0.*
webservers	The group webservers
webservers:moscow	Only hosts in webservers, not also in group moscow
webservers:&moscow	Only hosts in the group's webservers and moscow

### Playbooks

#### Sample Playbooks

```
#Every YAML file starts with ---
---
- hosts: webservers
  vars:
    http_port: 80
    max_clients: 200
    remote_user: root
  tasks:
    - name: ensure apache is at the latest version
      apt: name=httpd state=latest
    - name: write the apache config file
      template: src=/srv/httpd.j2 dest=/etc/httpd.conf
    notify:
      - restart apache
    - name: ensure apache is running (and enable it at boot)
      service: name=httpd state=started enabled=yes
    handlers:
      - name: restart apache
        service: name=httpd state=restarted
```

#### Writing Playbooks

```
#Generate the SSH Key and connect hosts to control machine before writing and running playbooks.
#Create a Playbook
$ vi <name of your file>.yml
#To write the playbook refer to the snapshot here.
#Run the playbook
$ ansible-playbook <name of your file>.yml
```



# Playbook

Thursday, September 23, 2021 3:00 PM

nano~/ansible-practice/files/landing-page.html.j2

~/ansible-practice/files/landing-page.html.j2

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>{{ page_title }}</title>
  <meta name="description" content="Created with Ansible">
</head>
<body>
  <h1>{{ page_title }}</h1>
  <p>{{ page_description }}</p>
</body>
</html>
```

```
---
- hosts: all
  become: yes
  vars:
    page_title: My Landing Page
    page_description: This is my landing page description.
  tasks:
    - name: Install Nginx
      apt:
        name: nginx
        state: latest

    - name: Apply Page Template
      template:
        src: files/landing-page.html.j2
        dest: /var/www/html/index.nginx-debian.html

    - name: Allow all access to tcp port 80
      ufw:
        rule: allow
        port: '80'
        proto: tcp
```

Remember to provide the `-K` option if you run this playbook, since it requires `sudo` permissions:

```
ansible-playbook -i inventoryplaybook-11.yml -u sammy-K
```

# Playbook Example

Wednesday, 15 February 2023 12:18 PM

```
--  
become: true  
become_method: sudo  
become_user: root  
gather_facts: true  
hosts: mysj-nginx-docker-QA  
tasks:  
-  
    name: "take backup of existing html folder"  
    register: backup_folders  
    stat:  
        path: /opt/backup  
  
    file:  
        path: /opt/backup  
        state: directory  
    name: "create backup folder if not exists"  
    when: "backup_folders.stat.exists == false"  
  
    file:  
        path: "/opt/backup/html_backup{{ansible_date_time.date}}"  
        state: directory  
    name: "create today's backup folder"  
  
    command: "mv /opt/mycloud/home /opt/backup/home{{ansible_date_time.time}}"  
    name: "mv the html folder to backup directory"  
  
    command: "mv /opt/mycloud/admin /opt/backup/admin{{ansible_date_time.time}}"  
    name: "mv the admin folder to backup directory"  
  
    command: "mv /opt/mycloud/search /opt/backup/search{{ansible_date_time.time}}"  
    name: "mv the search folder to backup directory"  
  
    name: "copy home folder to deploy folder of nginx server"  
    synchronize:  
        dest: "/opt/mycloud/home"  
        src: "{{static_files}}/home"  
  
    name: "copy admin folder to deploy folder of nginx server"  
    synchronize:  
        dest: "/opt/mycloud/admin"  
        src: "{{static_files}}/admin"  
  
    name: "copy search folder to deploy folder of nginx server"  
    synchronize:  
        dest: "/opt/mycloud/search"  
        src: "{{static_files}}/search"  
vars:  
    backup_folder: "/opt/backup/html_backup{{ansible_date_time.date}}"  
    deploy_folder: /opt/mycloud  
    static_files: /data/staticfiles/mycloudqa-epms
```

# Ansible Dry Run – Check Mode

Tuesday, 4 April 2023 11:31 AM

## Check Mode

Use the **-C** or **--check** flag with the **ansible-playbook** command to do a dry run of an Ansible playbook:

```
ansible-playbook playbook.yaml --check -k
```

---

**check\_mode** parameter to the playbook content:

```
---
```

```
- hosts: all
  tasks:
    - name: A command to run in check mode
      command: /your/command
      check_mode: on
```

---

what changes were made while executing the playbook:

```
ansible-playbook playbook.yaml --diff
```

However, if your playbook contains conditional or result-based tasks, it won't work in check mode.

## ad-hoc Commands

Saturday, September 11, 2021 5:36 PM

Ad hoc commands are **commands which can be run individually to perform quick functions**.

This will execute `uname -a` on all the nodes in your inventory:

```
ansible all-a "uname -a"
```

[Copy](#)

It is also possible to run Ansible modules with the option `-m`. The following command would install the package `vim` on `server1` from your inventory:

```
ansible server1-m apt -a "name=vim"
```

[Copy](#)

Before making changes to your nodes, you can conduct a *dry run* to predict how the servers would be affected by your command. This can be done by including the `--check` option:

```
ansible server1-m apt -a "name=vim" --check
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Running Playbooks

Saturday, September 11, 2021 5:38 PM

To run a playbook and execute all the tasks defined within it, use the `ansible-playbook` command:

```
ansible-playbook myplaybook.yml
```

[Copy](#)

To overwrite the default `hosts` option in the playbook and limit execution to a certain group or host, include the option `-l` in your command:

```
ansible-playbook -l server1 myplaybook.yml
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Getting Information about a Play

Saturday, September 11, 2021 5:40 PM

The option `--list-tasks` is used to list all tasks that would be executed by a play without making any changes to the remote servers:

```
ansible-playbook myplaybook.yml--list-tasks
```

[Copy](#)

Similarly, it is possible to list all hosts that would be affected by a play, without running any tasks on the remote servers:

```
ansible-playbook myplaybook.yml--list-hosts
```

[Copy](#)

You can use `tags` to limit the execution of a play. To list all tags available in a play, use the option `--list-tags`:

```
ansible-playbook myplaybook.yml--list-tags
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Testing Connectivity to Nodes

Saturday, September 11, 2021 5:30 PM

To test that Ansible is able to connect and run commands and playbooks on your nodes, you can use the following command:

```
ansible all -m ping
```

[Copy](#)

The `ping` module will test if you have valid credentials for connecting to the nodes defined in your inventory file, in addition to testing if Ansible is able to run Python scripts on the remote server. A *pong* reply back means Ansible is ready to run commands and playbooks on that node.

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

## Connecting as a Different User

Saturday, September 11, 2021 5:32 PM

By default, Ansible tries to connect to the nodes as your current system user, using its corresponding SSH keypair. To connect as a different user, append the command with the `-u` flag and the name of the intended user:

```
ansible all -m ping -u sammy
```

[Copy](#)

The same is valid for `ansible-playbook`:

```
ansible-playbook myplaybook.yml -u sammy
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Using a Custom SSH Key

Saturday, September 11, 2021 5:32 PM

If you're using a custom SSH key to connect to the remote servers, you can provide it at execution time with the `--private-key` option:

```
ansible all -m ping --private-key=~/ssh/custom_id
```

[Copy](#)

This option is also valid for `ansible-playbook`:

```
ansible-playbook myplaybook.yml --private-key=~/ssh/custom_id
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Using Password-Based Authentication

Saturday, September 11, 2021 5:33 PM

If you need to use *password-based authentication* in order to connect to the nodes, you need to append the option `--ask-pass` to your Ansible command.

This will make Ansible prompt you for the password of the user on the remote server that you're attempting to connect as:

```
ansible all -m ping --ask-pass
```

[Copy](#)

This option is also valid for `ansible-playbook`:

```
ansible-playbook myplaybook.yml --ask-pass
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Providing the sudo Password

Saturday, September 11, 2021 5:34 PM

If the remote user needs to provide a password in order to run `sudo` commands, you can include the option `--ask-become-pass` to your Ansible command. This will prompt you to provide the remote user sudo password:

```
ansible all -m ping --ask-become-pass
```

[Copy](#)

This option is also valid for `ansible-playbook`:

```
ansible-playbook myplaybook.yml --ask-become-pass
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Using a Custom Inventory File

Saturday, September 11, 2021 5:35 PM

The default inventory file is typically located at `/etc/ansible/hosts`, but you can also use the `-i` option to point to custom inventory files when running Ansible commands and playbooks. This is useful for setting up per-project inventories that can be included in version control systems such as Git:

```
ansible all -m ping -i my_custom_inventory
```

[Copy](#)

The same option is valid for `ansible-playbook`:

```
ansible-playbook myplaybook.yml -i my_custom_inventory
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Using a Dynamic Inventory File

Saturday, September 11, 2021 5:36 PM

Ansible supports *inventory scripts* for building dynamic inventory files. This is useful if your inventory fluctuates, with servers being created and destroyed often.

You can find a number of [open source inventory scripts](#) on the official Ansible GitHub repository. After downloading the desired script to your Ansible control machine and setting up any required information — such as API credentials — you can use the executable as custom inventory with any Ansible command that supports this option.

The following command uses Ansible's [DigitalOcean inventory script](#) with a `ping` command to check connectivity to all current active servers:

```
ansible all -m ping-i digital_ocean.py
```

[Copy](#)

For more details on how to use dynamic inventory files, please refer to the [official Ansible documentation](#).

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Controlling Playbook Execution

Saturday, September 11, 2021 5:41 PM

You can use the option `--start-at-task` to define a new entry point for your playbook. Ansible will then skip anything that comes before the specified task, executing the remaining of the play from that point on. This option requires a valid *task name* as argument:

```
ansible-playbook myplaybook.yml --start-at-task="Set Up Nginx"
```

[Copy](#)

To only execute tasks associated with specific tags, you can use the option `--tags`. For instance, if you'd like to only execute tasks tagged as `nginx` or `mysql`, you can use:

```
ansible-playbook myplaybook.yml --tags=mysql,nginx
```

[Copy](#)

If you want to skip all tasks that are under specific tags, use `--skip-tags`. The following command would execute `myplaybook.yml`, skipping all tasks tagged as `mysql`:

```
ansible-playbook myplaybook.yml --skip-tags=mysql
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Ansible Vault to Store Sensitive Data

Saturday, September 11, 2021 5:43 PM

If your Ansible playbooks deal with sensitive data like passwords, API keys, and credentials, it is important to keep that data safe by using an encryption mechanism. Ansible provides `ansible-vault` to encrypt files and variables.

Even though it is possible to encrypt any Ansible data file as well as binary files, it is more common to use `ansible-vault` to encrypt *variable files* containing sensitive data. After encrypting a file with this tool, you'll only be able to execute, edit or view its contents by providing the relevant password defined when you first encrypted the file.

## Creating a New Encrypted File

You can create a new encrypted Ansible file with:

```
ansible-vault create credentials.yml
```

[Copy](#)

This command will perform the following actions:

- First, it will prompt you to enter a new password. You'll need to provide this password whenever you access the file contents, whether it's for editing, viewing, or just running playbooks or commands using those values.
- Next, it will open your default command-line editor so you can populate the file with the desired contents.
- Finally, when you're done editing, `ansible-vault` will save the file as encrypted data.

## Encrypting an Existing Ansible File

To encrypt an existing Ansible file, you can use the following syntax:

```
ansible-vault encrypt credentials.yml
```

[Copy](#)

This will prompt you for a password that you'll need to enter whenever you access the file `credentials.yml`.

## Viewing the Contents of an Encrypted File

If you want to view the contents of a file that was previously encrypted with `ansible-vault` and you don't need to change its contents, you can use:

```
ansible-vault view credentials.yml
```

[Copy](#)

This will prompt you to provide the password you selected when you first encrypted the file with `ansible-vault`.

## Editing an Encrypted File

To edit the contents of a file that was previously encrypted with Ansible Vault, run:

```
ansible-vault edit credentials.yml
```

[Copy](#)

This will prompt you to provide the password you chose when first encrypting the file `credentials.yml` with `ansible-vault`. After password validation, your default command-line editor will open with the unencrypted contents of the file, allowing you to make your changes. When finished, you can save and close the file as you would normally, and the updated contents will be saved as encrypted data.

## Decrypting Encrypted Files

If you wish to permanently revert a file that was previously encrypted with `ansible-vault` to its unencrypted version, you can do so with this syntax:

```
ansible-vault decrypt credentials.yml
```

[Copy](#)

This will prompt you to provide the same password used when first encrypting the file `credentials.yml` with `ansible-vault`. After password validation, the file contents will be saved to the disk as unencrypted data.

## Using Multiple Vault Passwords

Ansible supports multiple vault passwords grouped by different vault IDs. This is useful if you want to have dedicated vault passwords for different environments, such as development, testing, and production environments.

To create a new encrypted file using a custom vault ID, include the `--vault-id` option along with a *label* and the location where `ansible-vault` can find the password for that vault. The label can be any identifier, and the location can either be `prompt`, meaning that the command should prompt you to enter a password, or a valid path to a password file.

```
ansible-vault create --vault-id dev@prompt credentials_dev.yml
```

[Copy](#)

This will create a new vault ID named `dev` that uses `prompt` as password source. By combining this method with group variable files, you'll be able to have separate ansible vaults for each application environment:

```
ansible-vault create --vault-id prod@prompt credentials_prod.yml
```

[Copy](#)

We used `dev` and `prod` as vault IDs to demonstrate how you can create separate vaults per environment, but you can create as many vaults as you want, and you can use any identifier of your choice as vault ID.

Now to view, edit, or decrypt these files, you'll need to provide the same vault ID and password source along with the `ansible-vault` command:

```
ansible-vault edit credentials_dev.yml --vault-id dev@prompt
```

[Copy](#)

## Using a Password File

If you need to automate the process of provisioning servers with Ansible using a third-party tool, you'll need a way to provide the vault password without being prompted for it. You can do that by using a *password file* with `ansible-vault`.

A password file can be a plain text file or an executable script. If the file is an executable script, the output produced by this script will be used as the vault password. Otherwise, the raw contents of the file will be used as vault password.

To use a password file with `ansible-vault`, you need to provide the path to a password file when running any of the vault commands:

```
ansible-vault create --vault-id dev@path/to/passfilecredentials_dev.yml
```

[Copy](#)

Ansible doesn't make a distinction between content that was encrypted using `prompt` or a password file as password source, as long as the input password is the same. In practical terms, this means it is OK to encrypt a file using `prompt` and then later use a password file to store the same password used with the `prompt` method. The opposite is also true: you can encrypt content using a password file and later use the `prompt` method, providing the same password when prompted by Ansible.

For extended flexibility and security, instead of having your vault password stored in a plain text file, you can use a Python script to obtain the password from other sources. The official Ansible repository contains [a few examples of vault scripts](#) that you can use for reference when creating a custom script that suits the particular needs of your project.

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Data Encrypted via Ansible Vault

Saturday, September 11, 2021 5:43 PM

Whenever you run a playbook that uses data previously encrypted via `ansible-vault`, you'll need to provide the vault password to your playbook command.

If you used default options and the `prompt` password source when encrypting the data used in this playbook, you can use the option `--ask-vault-pass` to make Ansible prompt you for the password:

```
ansible-playbook myplaybook.yml --ask-vault-pass
```

[Copy](#)

If you used a password file instead of prompting for the password, you should use the option `--vault-password-file` instead:

```
ansible-playbook myplaybook.yml --vault-password-file my_vault_password.py
```

[Copy](#)

If you're using data encrypted under a vault ID, you'll need to provide the same vault ID and password source you used when first encrypting the data:

```
ansible-playbook myplaybook.yml --vault-id dev@prompt
```

[Copy](#)

If using a password file with your vault ID, you should provide the label followed by the full path to the password file as password source:

```
ansible-playbook myplaybook.yml --vault-id dev@vault_password.py
```

[Copy](#)

If your play uses multiple vaults, you should provide a `--vault-id` parameter for each of them, in no particular order:

```
ansible-playbook myplaybook.yml --vault-id dev@vault_password.py --vault-id test@prompt --vault-id ci
```

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

## Debugging

Saturday, September 11, 2021 5:44 PM

If you run into errors while executing Ansible commands and playbooks, it's a good idea to increase output verbosity in order to get more information about the problem. You can do that by including the `-v` option to the command:

```
ansible-playbook myplaybook.yml -v
```

[Copy](#)

If you need more detail, you can use `-vvv` and this will increase verbosity of the output. If you're unable to connect to the remote nodes via Ansible, use `-vvvv` to get connection debugging information:

```
ansible-playbook myplaybook.yml -vvvv
```

[Copy](#)

From <<https://www.digitalocean.com/community/cheatsheets/how-to-use-ansible-cheat-sheet-guide>>

# Intro

Wednesday, 29 March 2023 10:55 PM

Ansible:

- Simple
- Powerful
- Agentless

Used Case:

- Provisioning
- Configuration Management
- Continues Delivery
- Application Deployment
- Security Compliance

# Inventory

Thursday, 30 March 2023 1:50 PM

Default file /etc/ansible/hosts

server1.abc.com

[db]

server2.abc.com

server3.abc.com

## Alias

```
web  ansible_host=server1.abc.com ansible_connection=ssh      ansible_user=root
db    ansible_host=server2.abc.com ansible_connection=winrm.  Ansible_user=admin
web  ansible_host=server1.abc.com ansible_connection=ssh      ansible_ssh_pass=P@ssword
```

localhost ansible\_connection=localhost

## Inventory Parameters:

ansible\_connection - ssh/winrm/localhost

ansible\_port -22/5986

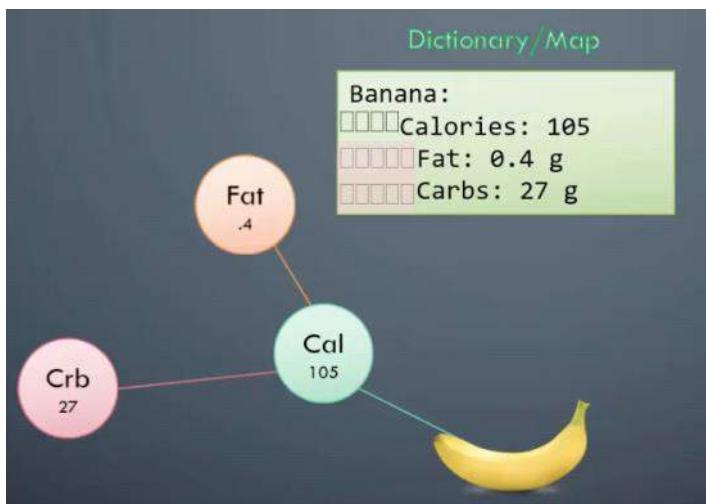
ansible\_user - root/administrator

ansible\_ssh\_pass - password

# YAML

Saturday, 1 April 2023 6:23 PM

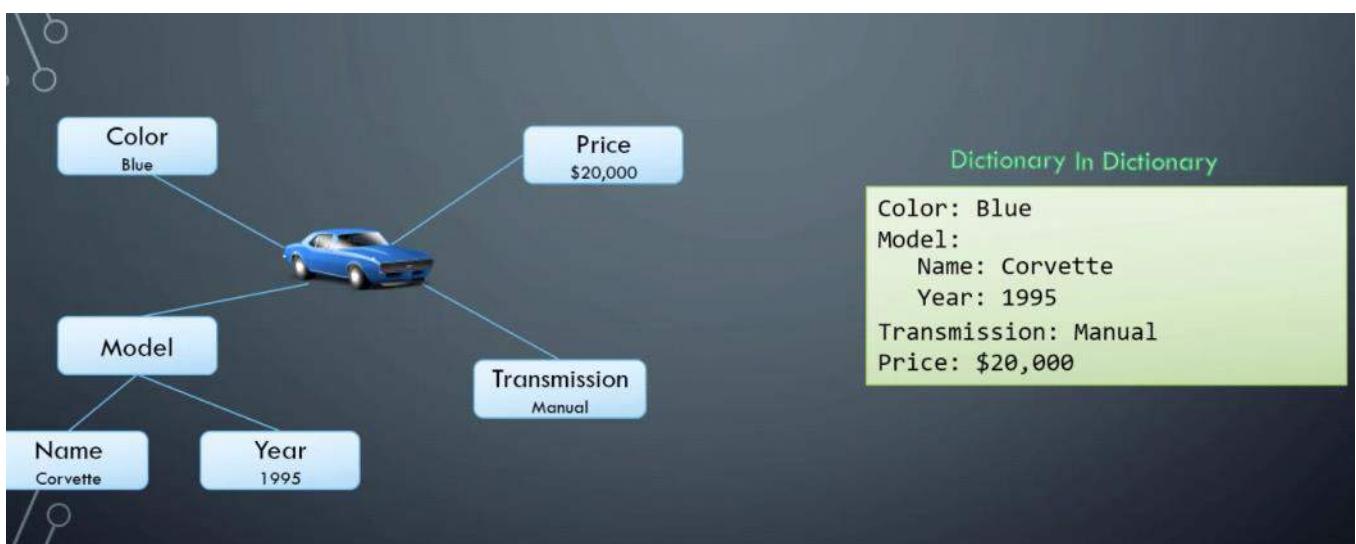
```
Servers:  
- name: Server1  
  owner: John  
  created: 12232012  
  status: active
```



## Key Value/Dictionary/Lists

### Fruits:

- Banana:
  - Calories: 105
  - Fat: 0.4 g
  - Carbs: 27 g
  
- Grape:
  - Calories: 62
  - Fat: 0.3 g
  - Carbs: 16 g



## Dictionary/Map

```
Banana:  
  Calories: 105  
  Fat: 0.4 g  
  Carbs: 27 g
```



```
Banana:  
  Calories: 105  
  Carbs: 27 g  
  Fat: 0.4 g
```



Dictionary – Unordered  
List – Ordered

## Array/List

```
Fruits:  
  - Orange  
  - Apple  
  - Banana
```



```
Fruits:  
  - Orange  
  - Banana  
  - Apple
```

```
# List of Fruits  
Fruits:  
  - Orange  
  - Apple  
  - Banana
```



Hash # – Comments

# Playbooks

Sunday, 2 April 2023 10:04 PM

## Ansible playbooks

### # Simple Ansible Playbook

- Run command1 on server1
- Run command2 on server2
- Run command3 on server3
- Run command4 on server4
- Run command5 on server5
- Run command6 on server6
- Run command7 on server7
- Run command8 on server8
- Run command9 on server9
- Restarting Server1
- Restarting Server2
- Restarting Server3
- Restarting Server4
- Restarting Server5
- Restarting Server6
- Restarting Server7

### # Complex Ansible Playbook

- Deploy 50 VMs on Public Cloud
- Deploy 50 VMs on Private Cloud
- Provision Storage to all VMs
- Setup Network Configuration on Private VMs
- Setup Cluster Configuration
- Configure Web server on 20 Public VMs
- Configure DB server on 20 Private VMs
- Setup Loadbalancing between web server VMs
- Setup Monitoring components
- Install and Configure backup clients on VMs
- Update CMDB database with new VM Information

```
# ansible-playbook playbook.yml  
# ansible-playbook --help
```

# RUN

Sunday, 2 April 2023 10:18 PM

## Ansible

```
# ansible <host> -a <command>
# ansible all -a "/sbin/reboot"
# ansible <hosts> -m <module>
# ansible target -m ping
```

## Ansible Playbooks

```
# ansible-playbook <playbook name>
# ansible-playbook playbook-webserver.yaml
```

## Inventory

```
target1 ansible_host=192.168.100.89
target2 ansible_host=192.168.100.41
```

```
# ansible-playbook playbook-test.yaml -l inventory.txt
```

```
-  
  name: Test Connectivity to target servers  
  hosts: all  
  tasks:  
    - name: Ping Test  
      ping:
```

# Module

Sunday, 2 April 2023 10:14 PM

Different task run by playbook call modules.

- Command
- Script
- Yum
  - o Name: httpd
  - o State - Present
- Service
  - o Name: httpd
  - o State - Started

```
# ansible-doc -l
```

command

Executes a command on a remote node

parameter	comments
chdir	cd into this directory before running the command
creates	a filename or (since 2.0) glob pattern, when it already exists, this step will not be run.
executable	change the shell used to execute the command. Should be an absolute path to the executable.
free_form	the command module takes a free form command to run. There is no parameter actually named 'free form'. See the examples!
removes	a filename or (since 2.0) glob pattern, when it does not exist, this step will not be run.
warn (added in 1.8)	if command warnings are on in ansible.cfg, do not warn about this particular line if set to no/false.

playbook.yml

```
- name: Play 1
hosts: localhost
tasks:
  - name: Execute command 'date'
    command: date

  - name: Display resolv.conf contents
    command: cat /etc/resolv.conf

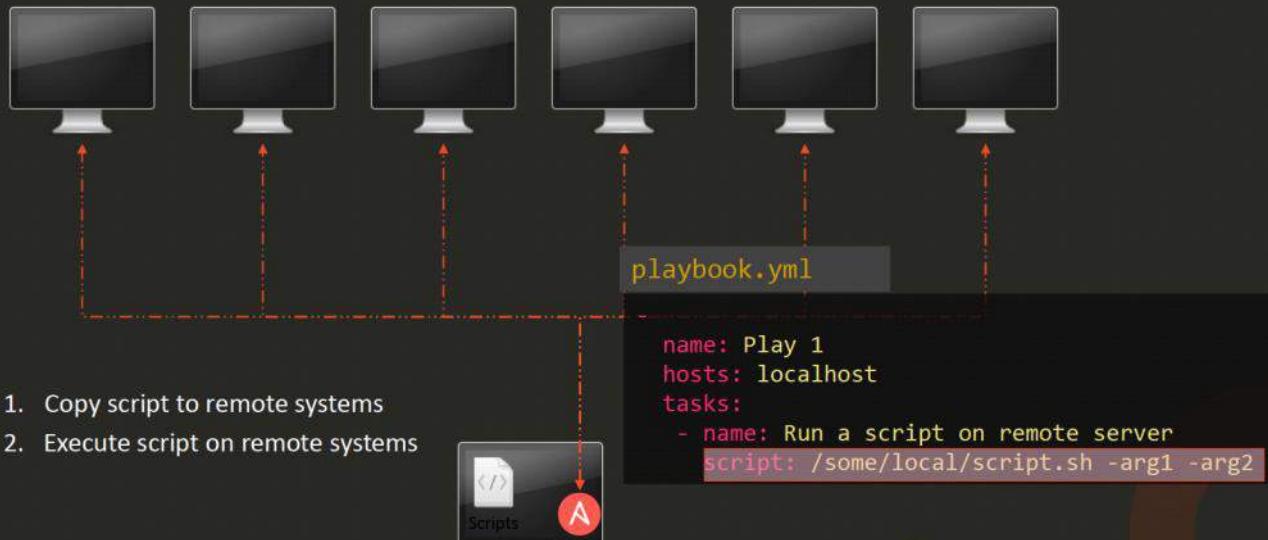
  - name: Display resolv.conf contents
    command: cat resolv.conf chdir=/etc

  - name: Display resolv.conf contents
    command: mkdir /folder creates=/folder

  - name: Copy file from source to destination
    copy: src=/source_file dest=/destination
```

## script

- Runs a local script on a remote node after transferring it



## Service

- Manage Services – Start, Stop, Restart

### playbook.yml

```
- name: Start Services in order
  hosts: localhost
  tasks:
    - name: Start the database service
      service: name=postgresql state=started

    - name: Start the httpd service
      service: name=httpd state=started

    - name: Start the nginx service
      service:
        name: nginx
        state: started
```

### playbook.yml

```
- name: Start Services in order
  hosts: localhost
  tasks:
    - name: Start the database service
      service:
        name: postgresql
        state: started
```

# lineinfile

- Search for a line in a file and replace it or add it if it doesn't exist.

```
/etc/resolv.conf
```

```
nameserver 10.1.250.1
nameserver 10.1.250.2
```

```
nameserver 10.1.250.10
```

```
playbook.yml
```

```
->
  name: Add DNS server to resolv.conf
  hosts: localhost
  tasks:
    - lineinfile:
        path: /etc/resolv.conf
        line: 'nameserver 10.1.250.10'
```

```
script.sh
```

```
#Sample script
```

```
echo "nameserver 10.1.250.10" >> /etc/resolv.conf
```

```
/etc/resolv.conf
```

```
nameserver 10.1.250.1
nameserver 10.1.250.2
nameserver 10.1.250.10
```

```
/etc/resolv.conf
```

```
nameserver 10.1.250.1
nameserver 10.1.250.2
nameserver 10.1.250.10
nameserver 10.1.250.10
nameserver 10.1.250.10
```

# List Modules

Monday, 3 April 2023 11:52 AM

- System
  - User
  - Group
  - Hostname
  - Iptables
  - Lvg
  - Lvol
  - Make
  - Mount
  - Ping
  - Timezone
  - Systemd
  - Service

- Commands
  - Commands
  - Expect
  - Raw
  - Script
  - Shell

- Files
  - Acl
  - Archive
  - Copy
  - File
  - Find
  - Lineinfile
  - Replace
  - Stat
  - Template
  - Unarchive

- Database
  - MongoDB
  - Mssql
  - Mysql
  - Postgresql
  - Proxysql
  - Vertica

- Cloud
  - Amazon
  - Atomic
  - Azure
  - Centrylink
  - Cloudscale
  - Digital Ocean
  - Docker
  - Google
  - Linode
  - Openstack
  - Rackspace
  - Smartos
  - Softlayer
  - VMware

- Windows
  - Win\_copy
  - Win\_command
  - Win\_domain
  - Win\_file
  - Win\_iis\_website
  - Win\_msg
  - Win\_msi
  - Win\_package
  - Win\_ping
  - Win\_path
  - Win\_robocopy
  - Win\_regedit
  - Win\_shell
  - Win\_service
  - Win\_user
  - And More

## Basic file

```
---hosts:production
remote_user:root
tasks:....
```

Place your modules inside tasks.

## Task formats

### One-line

```
-apt:pkg=vim state=present
```

### Map

```
-apt:pkg:vim
  state:present
```

### Foldable scalar

```
-apt:>
  pkg=vim
  state=present
```

Define your tasks in any of these formats. One-line format is preferred for short declarations, while maps are preferred for longer.

## #Modules

## Aptitude

### Packages

```
-apt:pkg:nodejs
  state:present # absent | latestupdate_cache:yes
  force:no
```

### Deb files

```
-apt:deb:"https://packages.erlang-solutions.com/erlang-solutions_1.0_all.deb"
```

### Repositories

```
-apt_repository:repo:"deb https://... raring main"state:present
```

### Repository keys

```
-apt_key:id:AC40B2F7
  url:"http://..."state:present
```

## git

```
-git:repo:git://github.com/
  dest:/srv/checkout
  version:master
  depth:10bare:yes
```

See: [git module](#)

## git\_config

```
-git_config:name:user.email
  scope:global # local | systemvalue:hi@example.com
```

See: [git\\_config module](#)

## user

```
-user:state:present
  name:git
  system:yes
  shell:/bin/sh
  groups:admin
  comment:"Git Version Control"
```

See: [user module](#)

## service

```
-service:name:nginx
  state:started
  enabled:yes  # optional
```

See: [service module](#)

## #Shell

### shell

```
-shell:apt-get install nginx -y
Extra options
-shell:echo hello
  args:creates:/path/file # skip if this existsremoves:/path/file # skip if this is missingchdir:/path
Multiline example
-shell:|
  echo "hello there"
  echo "multiple lines"
See: shell module
```

### script

```
-script:/x/y/script.sh
  args:creates:/path/file # skip if this existsremoves:/path/file # skip if this is missingchdir:/path
See: script module
```

## #Files

### file

```
-file:path:/etc/dir
  state:directory # file | link | hard | touch | absent# Optional:owner:bin
  group:wheel
  mode:0644recurse:yes # mkdir -pforce:yes  # ln -nfs
See: file module
```

### copy

```
-copy:src:/app/config/nginx.conf
  dest:/etc/nginx/nginx.conf
# Optional:owner:user
  group:user
  mode:0644backup:yes
See: copy module
```

### template

```
-template:src:config/redis.j2
  dest:/etc/redis.conf
# Optional:owner:user
  group:user
  mode:0644backup:yes
See: template module
```

## #Local actions

### local\_action

```
-name:do something locally
  local_action:shell echo hello
```

### debug

```
-debug:msg:"Hello {{ var }}"
```

See: [debug module](#)

# Variables

Monday, 3 April 2023 12:58 PM

Stores information that varies with each host

```
Playbook.yml

-
  name: Add DNS server to resolv.conf
  hosts: localhost
  vars:
    dns_server: 10.1.250.10
  tasks:
    - lineinfile:
        path: /etc/resolv.conf
        line: 'nameserver {{ dns_server }}'
```

```
- name: Set Firewall Configurations
  hosts: web
  tasks:
    - firewalld:
        service: https
        permanent: true
        state: enabled

    - firewalld:
        port: '{{ http_port }}'/tcp
        permanent: true
        state: disabled

    - firewalld:
        port: '{{ snmp_port }}'/udp
        permanent: true
        state: disabled

    - firewalld:
        source: '{{ inter_ip_range }}'/24
        Zone: internal
        state: enabled
```

```
#Sample Inventory File
Web http_port=      snmp_port=      inter_ip_range=
```

```
#Sample variable File - web.yml
http_port: 8081
snmp_port: 161-162
inter_ip_range: 192.0.2.0
```

```
{}      }
```

Jinja2 Templating

 source: {{ inter_ip_range }}	 source: '{{ inter_ip_range }}'
 source: SomeThing{{ inter_ip_range }}SomeThing	

# Conditionals

Monday, 3 April 2023 1:21 PM

```
- name: Test Connectivity to target servers
  hosts: all
  tasks:
    - name: 'Install NGINX Debian'
      apt:
        name: nginx
        state: present

      name: Test Connectivity to target servers
      hosts: all
      tasks:
        - name: 'Install NGINX on Redhat'
          yum:
            name: nginx
            state: present

---  

- name: Test Connectivity to target servers
  hosts: all
  tasks:
    - name: 'Install NGINX on Debian'
      apt:
        name: nginx
        state: present
      when: ansible_os_family == "Debian" and
            ansible_distribution_version == "16.04"

    - name: 'Install NGINX on redhat'
      yum:
        name: nginx
        state: present
      when: ansible_os_family == "Redhat" or
            ansible_os_family == "SUSE"

when: 'ansible_host=="node02"'
```

# Conditionals in Loops

Monday, 3 April 2023 1:37 PM

```
---
- name: Test Connectivity to target servers
  hosts: all
  vars:
    packages:
      - name: nginx
        required: true
      - name: mysql
        required: true
      - name: apache
        required: true

  tasks:
    - name: Install "{{item.name}}" on Debian
      apt:
        name: "{{ item.name }}"
        state: present

        when: item.required == True
      loop: "{{ packages }}"
```

Enables you to execute a set of commands repeatedly  
Creating multiple users at once  
installing many packages on hundreds of servers

# Conditional & Register

Monday, 3 April 2023 1:43 PM

```
- name: Check status of Service and email if it's down
hosts: localhost
tasks:
  - command: service httpd status
    register: result
  - mail:
      to: admin@abc.com
      subject: sErvice Alert
      body: Httpd Service is down
      when: result.stdout.find('down') != -1
```

# Loops

Tuesday, 4 April 2023 10:40 AM

## LOOPS - Visualize

```
name: Create users
hosts: localhost
tasks:
- user: name='{{ ???? }}' state=present uid='{{ ? }}'
  loop:
    - name: joe
      uid: 1010
    - name: george
      uid: 1011
    - name: ravi
      uid: 1012
    - name: mani
      uid: 1013
    - name: kiran
      uid: 1014
    - name: jazlan
      uid: 1015
    - name: emaan
      uid: 1016
    - name: mazin
      uid: 1017
    - name: izaan
      uid: 1018
    - name: mike
```

```
name: Create users
hosts: localhost
tasks:
- var:
  item:

    user: name='{{ ???? }}' state=present uid='{{ ? }}'

- var:
  item:

    user: name='{{ ???? }}' state=present uid='{{ ? }}'

- var:
  item:

    user: name='{{ ???? }}' state=present uid='{{ ? }}'

- var:
  item:

    user: name='{{ ???? }}' state=present uid='{{ ? }}'

- var:
  item:

    user: name='{{ ???? }}' state=present uid='{{ ? }}'
```

## LOOPS - Visualize

```
name: Create users
hosts: localhost
tasks:
- user: name='{{ item.name }}' state=present uid='{{ item.uid }}'
  loop:
    - name: joe      - { name: joe, uid: 1010 }
      uid: 1010
    - name: george   - { name: george, uid: 1011 }
      uid: 1011
    - name: ravi     - { name: ravi, uid: 1012 }
      uid: 1012
    - name: mani     - { name: mani, uid: 1013 }
      uid: 1013
    - name: kiran    - { name: kiran, uid: 1014 }
      uid: 1014
    - name: jazlan   - { name: jazlan, uid: 1015 }
      uid: 1015
    - name: emaan    - { name: emaan, uid: 1016 }
      uid: 1016
    - name: mazin    - { name: mazin, uid: 1017 }
      uid: 1017
    - name: izaan    - { name: izaan, uid: 1018 }
      uid: 1018
    - name: mike     - { name: mike, uid: 1019 }
```

```
name: Create users
hosts: localhost
tasks:
- var:
  item:
    name: joe
    uid: 1010
    user: name='{{ item.name }}' state=present uid='{{ item.uid }}'

- var:
  item:
    name: george
    uid: 1011
    user: name='{{ item.name }}' state=present uid='{{ item.uid }}'

- var:
  item:
    name: ravi
    uid: 1012
    user: name='{{ item.name }}' state=present uid='{{ item.uid }}'

- var:
  item:
    name: mani
    uid: 1013
    user: name='{{ item.name }}' state=present uid='{{ item.uid }}'
```

## With\_\*

```
- name: Create users
hosts: localhost
tasks:
- user: name='{{ item }}' state=present
loop:
- joe
- george
- ravi
- mani
```

```
- name: Create users
hosts: localhost
tasks:
- user: name='{{ item }}' state=present
with_items:
- joe
- george
- ravi
- mani
```

## With\_\*

```
- name: Create users
hosts: localhost
tasks:
- user: name='{{ item }}' state=present
with_items:
- joe
- george
- ravi
- mani
```

```
- name: Get from multiple URLs
hosts: localhost
tasks:
- debug: var=item
with_url:
- "https://site1.com/get-servers"
- "https://site2.com/get-servers"
- "https://site3.com/get-servers"
```

```
- name: View Config Files
hosts: localhost
tasks:
- debug: var=item
with_file:
- "/etc/hosts"
- "/etc/resolv.conf"
- "/etc/ntp.conf"
```

```
- name: Check multiple mongodbs
hosts: localhost
tasks:
- debug: msg="DB={{ item.database }} PID={{ item.pid }}"
with_mongodb:
- database: dev
connection_string: "mongodb://dev.mongo/"
- database: prod
connection_string: "mongodb://prod.mongo/"
```

## With\_\*

```
with_items      With_redis
with_file       With_sequence
with_url        With_skydive
with_mongodb    With_subelements
with_dict        With_template
with_etcd        With_together
with_env         With_varnames
with_filetree   With_ini
With_inventory_hostnames With_k8s
With_k8s        With_manifold
With_manifold   With_nested
With_nested     With_nios
With_nios       With_openshift
With_openshift  With_password
With_password   With_pipe
With_pipe       With_rabbitmq
```

```
---
- name: 'Install required packages'
hosts: localhost
become: yes
vars:
  packages:
    - httpd
    - make
    - vim
tasks:
  - yum:
      name: vim
      state: present
  with_items: '{{ packages }}'
```

# Roles

Tuesday, 4 April 2023 11:05 AM

```
- name: Install and Configure MySQL
hosts: db-server
tasks:
  - name: Install Pre-Requisites
    yum: name=pre-req-packages state=present

  - name: Install MySQL Packages
    yum: name=mysql state=present

  - name: Start MySQL Service
    service: name=mysql state=started

  - name: Configure Database
    mysql_db: name=db1 state=present
```



mysql



- Installing Pre-requisites
- Installing mysql packages
- Configuring mysql service
- Configuring database and users



nginx



- Installing Pre-requisites
- Installing nginx packages
- Configuring nginx service
- Configuring custom web pages



Re-Use



mysql



- Installing Pre-requisites
- Installing mysql packages
- Configuring mysql service
- Configuring database and users

```
- name: Install and Configure MySQL
hosts: db-server1.....db-server100
roles:
  - mysql
```

## MySQL-Role

```
tasks:
  - name: Install Pre-Requisites
    yum: name=pre-req-packages state=present

  - name: Install MySQL Packages
    yum: name=mysql state=present

  - name: Start MySQL Service
    service: name=mysql state=started

  - name: Configure Database
    mysql_db: name=db1 state=present
```



This diagram illustrates the Ansible Role structure and its components:

- Organize:** Represented by a tree icon.
- Re-Use:** Represented by a circular arrow icon.
- Run:** Represented by a monitor icon.
- mysql:** Represented by a monitor icon with a database icon.

The mysql role consists of the following components:

- tasks:**

```

tasks:
  - name: Install Pre-Requisites
    yum: name=pre-req-packages state=present

  - name: Install MySQL Packages
    yum: name=mysql state=present

  - name: Start MySQL Service
    service: name=mysql state=started

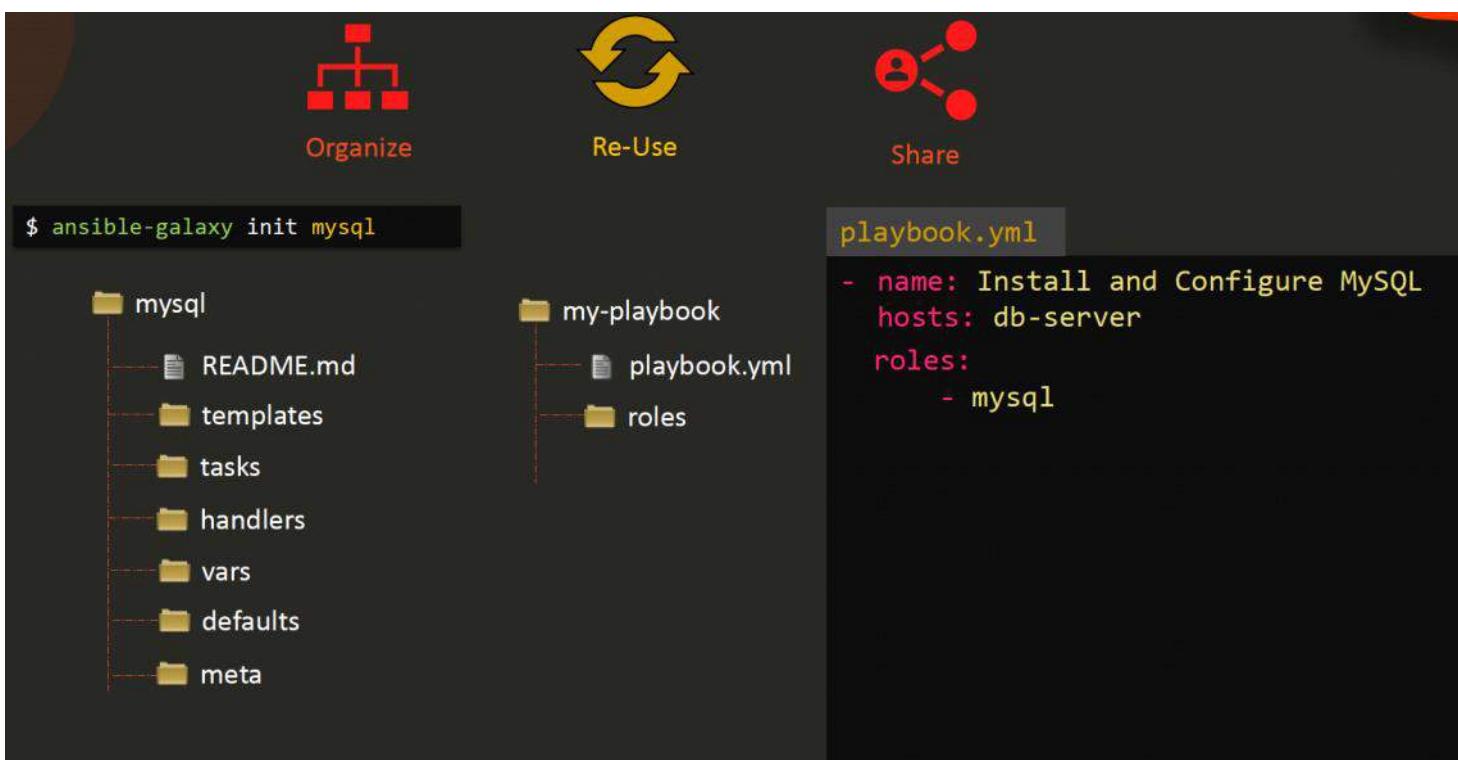
  - name: Configure Database
    mysql_db: name=db1 state=present
  
```
- vars:**

```

mysql_packages:
  - mysql
  - mysql-server
db_config:
  db_name: db1
  
```
- handlers:** An empty box.
- defaults:**

```

mysql_user_name: root
mysql_user_password: root
  
```
- templates:** An empty box.



This diagram shows the structure of a MySQL role and a sample playbook:

- Organize:** Represented by a tree icon.
- Re-Use:** Represented by a circular arrow icon.
- Share:** Represented by a network icon.

The MySQL role structure is as follows:

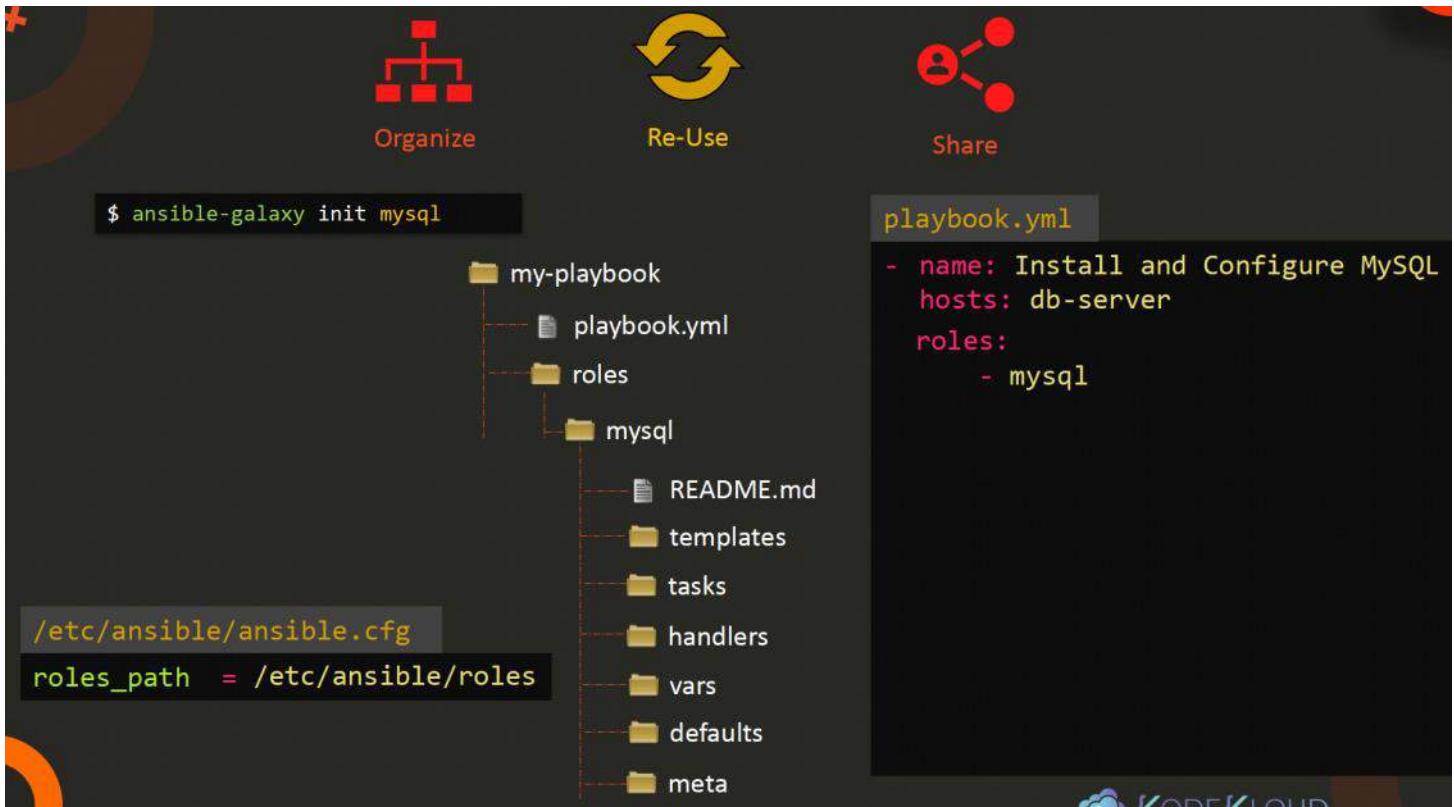
```

$ ansible-galaxy init mysql
.
├── mysql
│   ├── README.md
│   ├── templates
│   ├── tasks
│   ├── handlers
│   ├── vars
│   ├── defaults
│   └── meta
└── my-playbook
    ├── playbook.yml
    └── roles
        └── mysql
            ├── tasks
            ├── handlers
            ├── vars
            ├── defaults
            └── meta
  
```

The `playbook.yml` content is:

```

playbook.yml
- name: Install and Configure MySQL
  hosts: db-server
  roles:
    - mysql
  
```



## Use Role

```
$ ansible-galaxy install geerlingguy.mysql
```

```
- downloading role 'mysql', owned by geerlingguy
- downloading role from https://github.com/geerlingguy/ansible-role-mysql/archive/2.9.5.tar.gz
- extracting geerlingguy.mysql to /etc/ansible/roles/etc/ansible/roles/geerlingguy.mysql
- geerlingguy.mysql (2.9.5) was installed successfully
```

```
playbook.yml
```

```
-  
  name: Install and Configure MySQL  
  hosts: db-server  
  roles:  
    - geerlingguy.mysql
```

```
-  
  name: Install and Configure MySQL  
  hosts: db-server  
  roles:  
    - role: geerlingguy.mysql  
      become: yes  
      vars:  
        mysql_user_name: db-user
```

# List Roles

```
$ ansible-galaxy list
```

- geerlingguy.mysql
- kodekloud1.mysql

```
$ ansible-config dump | grep ROLE
```

```
DEFAULT_PRIVATE_ROLE_VARS(default) = False
DEFAULT_ROLES_PATH(default) = [u'/root/.ansible/roles', u'/usr/share/ansible/roles', u'/etc/ansible/roles']
GALAXY_ROLE_SKELETON(default) = None
GALAXY_ROLE_SKELETON_IGNORE(default) = ['^.git$', '^.*\.git_keep$']
```

```
$ ansible-galaxy install geerlingguy.mysql -p ./roles
```

# Sample

Monday, 8 May 2023 8:13 PM

# Inventory

Monday, 30 May, 2022 10:42 AM

```
# This is the default ansible 'hosts' file.  
#  
# It should live in /etc/ansible/hosts  
#  
# - Comments begin with the '#' character  
# - Blank lines are ignored  
# - Groups of hosts are delimited by [header] elements  
# - You can enter hostnames or ip addresses  
# - A hostname/ip can be a member of multiple groups
```

```
# Ex 1: Ungrouped hosts, specify before any group headers.
```

```
## green.example.com  
## blue.example.com  
## 192.168.100.1  
## 192.168.100.10
```

```
# Ex 2: A collection of hosts belonging to the 'webservers' group
```

```
## [webservers]  
## alpha.example.org  
## beta.example.org  
## 192.168.1.100  
## 192.168.1.110
```

```
# If you have multiple hosts following a pattern you can specify  
# them like this:
```

```
## www[001:006].example.com
```

```
# Ex 3: A collection of database servers in the 'dbservers' group
```

```
## [dbservers]  
##  
## db01.intranet.mydomain.net  
## db02.intranet.mydomain.net  
## 10.25.1.56  
## 10.25.1.57
```

```
# Here's another example of host ranges, this time there are no  
# leading 0s:
```

```
## db-[99:101]-node.example.com
```

```
#myvia-uat ansible_host=10.1.117.63 ansible_user=root  
mysjswarmpit ansible_host=10.1.116.222 ansible_user=root  
testingswarm ansible_host=10.1.116.214 ansible_user=root
```

```
#testingswarm ansible_host=47.254.234.62 ansible_user=kpisoft
#cdc-uat ansible_host=10.1.116.226 ansible_user=root
myclouduat-docker ansible_host=10.1.116.220 ansible_user=root
myclouduat-nginx ansible_host=10.1.116.214 ansible_user=root
mycloudprodnginx1 ansible_host=10.1.116.252 ansible_user=root
mycloudprodnginx3 ansible_host=10.1.117.33 ansible_user=root
mycloudprodnginx4 ansible_host=10.2.93.27 ansible_user=root
mycloudprodnginx5 ansible_host=10.1.117.141 ansible_user=root
mycloudprodnginx6 ansible_host=10.1.117.142 ansible_user=root
#MYSJ-QA
mysj-docker-qa ansible_host=10.2.95.129 ansible_user=root
mysj-nginx-docker-QA ansible_host=10.2.95.130 ansible_user=root
```

# Copy UI to Nginx.yml

Monday, 30 May, 2022 10:42 AM

```
---
- hosts: testingswarm
  gather_facts: true
  become: true
  become_user: root
  become_method: sudo
  vars:
    deploy_folder: /opt/mycloud
    backup_folder: /opt/backup/html_backup{{ansible_date_time.date}}
    static_files: /data/staticfiles/myclouduat-epms
  tasks:
    - name: take backup of existing html folder
      stat:
        path: /opt/backup
      register: backup_folders
    - name: create backup folder if not exists
      file:
        path: /opt/backup
        state: directory
      when: backup_folders.stat.exists == false
    - name: create today's backup folder
      file:
        path: /opt/backup/html_backup{{ansible_date_time.date}}
        state: directory
    - name: mv the html folder to backup directory
      command: mv {{deploy_folder}}/home {{backup_folder}}/home{{ansible_date_time.time}}
    - name: mv the admin folder to backup directory
      command: mv {{deploy_folder}}/admin {{backup_folder}}/admin{{ansible_date_time.time}}
    - name: mv the search folder to backup directory
      command: mv {{deploy_folder}}/search {{backup_folder}}/search{{ansible_date_time.time}}
    - name: copy home folder to deploy folder of nginx server
      synchronize:
        src: "{{static_files}}/home"
        dest: "{{deploy_folder}}"
    - name: copy admin folder to deploy folder of nginx server
      synchronize:
        src: "{{static_files}}/admin"
        dest: "{{deploy_folder}}"
    - name: copy search folder to deploy folder of nginx server
      synchronize:
        src: "{{static_files}}/search"
        dest: "{{deploy_folder}}"
```

# Nginx\_reload.yml

Monday, 30 May, 2022 10:43 AM

```
---
- hosts: mycloudprodnginx1
  tasks:
    - name: reload nginx
      service:
        name: nginx
        state: reloaded
        enabled: yes
```

# Myclouduat-onetouch.yml

Monday, 30 May, 2022 10:45 AM

```
---
- hosts: mysjswarmpit
  become: yes
  become_method: sudo
  become_user: root
  gather_facts: true
  tasks:
    - name: copy the generic compose file to swarm location
      copy:
        src: /data/composefiles/myclouduat-stack.yml
        dest: /opt/compose/myclouduat-stack.yml
    - name: Replace the image tag for EPMS service
      shell: >
        sed -i -e 's/EPMSTAGNAME/{{ tagname }}/g' myclouduat-stack.yml
      args:
        chdir: "/opt/compose/"
    - name: Replace the image tag for EPMS Admin service
      shell: >
        sed -i -e 's/EPMSADMINTAGNAME/{{ tagname }}/g' myclouduat-stack.yml
      args:
        chdir: "/opt/compose/"

    - name: Replace the config file in the compose file for EPMS
      shell: >
        sed -i -e 's/EPMSCONFIGFILENAME/{{ epmsconfigfile }}/g' myclouduat-stack.yml
      args:
        chdir: "/opt/compose"

    - name: Replace the config file in the compose file for EPMS ADMIN
      shell: >
        sed -i -e 's/EPMSADMINCONFIGFILENAME/{{ epmsadminconfigfile }}/g' myclouduat-stack.yml
      args:
        chdir: "/opt/compose"
    - name: delete the log file
      file:
        path: /opt/skylark/logs/app.log
        state: absent
        delegate_to: myclouduat-docker
    - name: create empty log file and change the permission
      file:
        path: /opt/skylark/logs/app.log
        owner: root
        group: root
        mode: '1777'
        state: touch
        delegate_to: myclouduat-docker
    - name: Deploy the stack
```

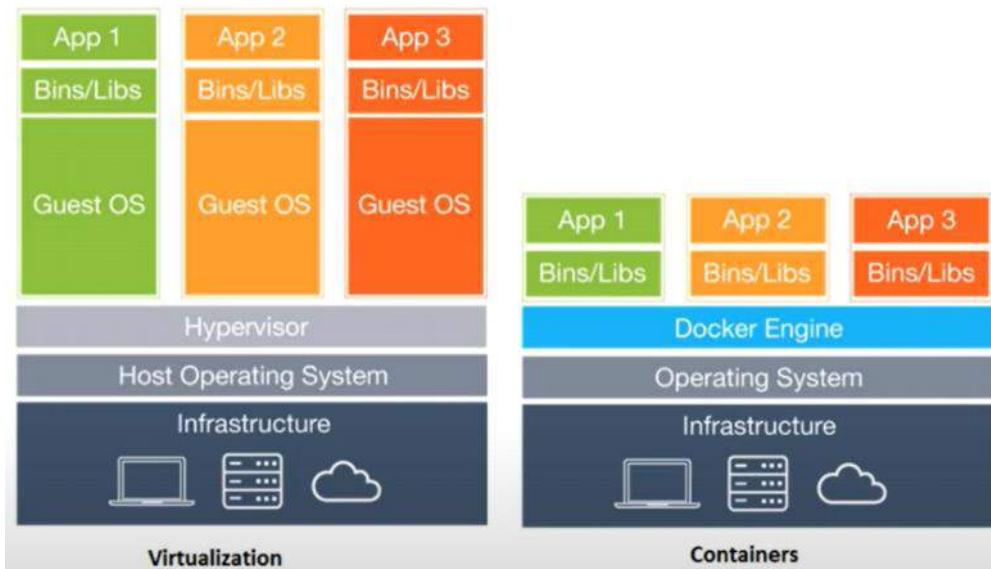
```
shell: >
  docker stack deploy -c /opt/compose/myclouduat-stack.yml mycloud
- name: check the running status
  wait_for:
    path: /opt/skylark/logs/app.log
    search_regex: "Started ApplicationInitializer"
    delay: 10
    timeout: 1800
    msg: "application startup failed"
  delegate_to: myclouduat-docker
```

```
--  
-  
  become: true  
  become_method: sudo  
  become_user: root  
  gather_facts: true  
  hosts: mysj-nginx-docker-QA  
  tasks:  
    -  
      name: "take backup of existing html folder"  
      register: backup_folders  
      stat:  
        path: /opt/backup  
    -  
      file:  
        path: /opt/backup  
        state: directory  
      name: "create backup folder if not exists"  
      when: "backup_folders.stat.exists == false"  
    -  
      file:  
        path: "/opt/backup/html_backup{{ansible_date_time.date}}"  
        state: directory  
      name: "create today's backup folder"  
    -  
      command: "mv /opt/mycloud/home /opt/backup/home{{ansible_date_time.time}}"  
      name: "mv the html folder to backup directory"  
    -  
      command: "mv /opt/mycloud/admin /opt/backup/admin{{ansible_date_time.time}}"  
      name: "mv the admin folder to backup directory"  
    -  
      command: "mv /opt/mycloud/search /opt/backup/search{{ansible_date_time.time}}"  
      name: "mv the search folder to backup directory"  
    -  
      name: "copy home folder to deploy folder of nginx server"  
      synchronize:  
        dest: "/opt/mycloud/home"  
        src: "{{static_files}}/home"  
    -  
      name: "copy admin folder to deploy folder of nginx server"  
      synchronize:  
        dest: "/opt/mycloud/admin"  
        src: "{{static_files}}/admin"  
    -  
      name: "copy search folder to deploy folder of nginx server"  
      synchronize:  
        dest: "/opt/mycloud/search"  
        src: "{{static_files}}/search"
```

```
vars:  
  backup_folder: "/opt/backup/html_backup{{ansible_date_time.date}}"  
  deploy_folder: /opt/mycloud  
  static_files: /data/staticfiles/mycloudqa-epms
```

# Docker vs Virtualization

Thursday, October 21, 2021 9:11 PM



You install your OS that's ur hypervisor and it runs mutliple VMs

You pay performance premium here because you have one kernel on OS and others on VM running machine

In Container you have same Host OS and same kernel.

You always boot to same OS

Just has bunch of process on host now

Those processes run inside the container

**Container holds** Cgroup, Namespase, Union-Capable FS

So container acts like a Linux OS

Namespace provides **isolation and limit for what you can see and what you can use**

**C group** will limit how much you can use

## Cgroup - Control Groups

- Resource metering and limiting
- Memory
- CPU
- Block I/O
- Network
- Device node (/dev/\*) access control

## Namespaces

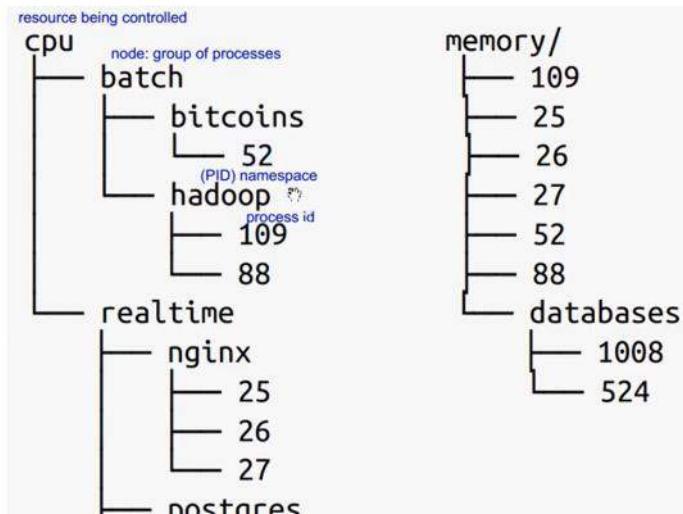
- Pid
- Net
- Mnt
- Uts
- Ipc

Each process is in one namespace of each type

## Pid namespace

- Processes within a PID namespace only see processes in the same PID namespace
- Each PID namespace has its own numbering (starting at 1)
- When PID 1 goes away, the whole namespace is killed
- Those namespaces can be nested
- A process ends up having multiple PIDs (one per namespace in which its nested)

## Example



# Docker

Tuesday, September 7, 2021 12:58 AM

## Docker

open-source **lightweight containerization technology**

allows you to automate the deployment of applications in lightweight and portable containers. Offers an efficient and easy initial set up

Allows you to describe your application lifecycle in detail

Simple configuration and interacts with Docker Compose.

Documentation provides every bit of information.

### Features of Docker:

- Easy Modeling
- Version control
- Placement/Affinity
- Application Agility
- Developer Productivity
- Operational Efficiencies

### Drawbacks of Docker?

- Doesn't provide a storage option
- Offer a poor monitoring option.
- No automatic rescheduling of inactive Nodes
- Complicated automatic horizontal scaling set up

# Docker Commands

Monday, August 30, 2021 5:02 PM

## Docker Installation

### Install yum-utils

```
# yum install -y yum-utils
```

### Add docker repo

```
# yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo
```

### Enable Docker repo

```
# yum-config-manager --enable docker-ce-nightly
```

### Test repo

```
# yum-config-manager --enable docker-ce-test
```

### Disable the repo

```
# yum-config-manager --disable docker-ce-nightly
```

### Docker packages install

```
# yum install -y docker-ce docker-ce-cli containerd.io
```

### Start docker service

```
# systemctl start docker; systemctl enable docker
```

### Run hello-world

```
# docker run hello-world
```

### Docker Compose

```
# curl -L "https://github.com/docker/compose/releases/download/1.26.2/docker-compose-\$\(uname -s\)-\$\(uname -m\)" -o /usr/local/bin/docker-compose; chmod +x /usr/local/bin/docker-compose; ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose; docker-compose --version
```

Then Run

```
# mv Dockerfile docker-compose.yml; docker-compose up -d; docker images
```

```
=====
```

### To build image

```
# docker build -t jenkins-test1 .
```

### Create local directory to map jenkins data to local

```
# mkdir /docker/jenkins/home_jenkins
```

### Run Docker

```
# docker run -d --name jenkins-docker -p 8080:8080 -p 50000:50000 -v /docker/jenkins/home_jenkins:/var/jenkins_home jenkins-test1  
# docker run -u 0 -d --name jenkins1 -p 8080:8080 -p 50000:50000 -v /jenkins-data/jenkins_home:/var/jenkins_home jenkins/jenkins  
# docker run -d --name jenkins-svr -p 8080:8080 -p 50000:50000 -v /docker/jenkins:/var/jenkins_home dockerfile
```

## Restart policy

Flag	Description
no	Do not automatically restart the container. (the default)
on-failure	Restart the container if it exits due to an error, which manifests as a non-zero exit code.
always	Always restart the container if it stops. If it is manually stopped, it is restarted only when Docker daemon restarts or the container itself is manually restarted. (See the second bullet listed in <a href="#">restart policy details</a> )
unless-stopped	Similar to <code>always</code> , except that when the container is stopped (manually or otherwise), it is not restarted even after Docker daemon restarts.

The following example starts a Redis container and configures it to always **restart** unless it is explicitly stopped or Docker is restarted.

```
$ docker run -d--restartunless-stopped redis
```

This command changes the **restart policy** for an already running container named **redis**.

```
$ docker update --restartunless-stopped redis
```

And this command will ensure all currently running containers will be restarted unless stopped.

```
$ docker update --restartunless-stopped $(docker ps -q)
```

- kill all running containers with **docker kill \$(docker ps -q)**
- delete all stopped containers with **docker rm \$(docker ps -a -q)**
- delete all images with **docker rmi \$(docker images -q)**

```
docker kill $(docker ps -q)  
docker rm $(docker ps -a -q)  
docker rmi $(docker images -q)
```

#### run the image as a container?

```
$ sudo docker run -i -t alpine /bin/bash
```

#### To get the IP Address from the container

```
[root@docker ~]# docker inspect -f 8c24ba549e91  
172.17.0.2
```

#### Bridge

```
# docker run ubuntu
```

#### None

```
# docker run ubuntu --network=none
```

#### Host

```
# docker run ubuntu --network=host
```

#### Login as root in container

```
# docker exec -u 0 -it jenkins bash
```

#### Run - start a container

```
# docker run nginx
```

#### Delete all stop Container in one time

```
# docker container prune
```

#### ps - list containers

```
# docker ps
```

```
# docker ps -a
```

#### STOP - stop a container

```
# docker ps
```

```
# docker stop silly_sammet
```

#### Rm - remove a container

```
# docker rm silly_sammet
```

```
# docker ps -a

Copy file inside Container
# docker cp script.sh shub_jenkins_1:/tmp/script.sh
```

```
images - List images
# docker images
```

```
rmi - Remove images
# docker rmi nginx
```

```
Pull - download an image
# docker run nginx
# docker pull nginx
# docker run ubuntu sleep 5
```

```
Exec - execute a command
# docker ps -a
# docker exec distracted_mcclintock cat /etc/hosts
```

```
Run - attach and detach
# docker run kodekloud/simple-webapp
# docker run -d kodekloud/simple-webapp
# docker attach a043d
```

```
Copy file inside Container
# docker cp script.sh shub_jenkins_1:/tmp/script.sh
```

```
Run container
# docker run centos
```

```
Log into the centOS image - use hub.docker.com for check images
# docker run -it centos bash
```

```
Run in background for 20 Sec as sleep mode
# docker run -d centos sleep 20
```

```
Container that exist
# docker ps -a
```

```
Stop container
# docker run -d centos sleep 2000
```

```
[root@dns ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
bc8b4b78beb0        centos              "sleep 2000"       10 seconds ago    Up 10 seconds          cocky_brahmagupta
```

**Name of the container: cocky\_brahmagupta**

**Force Kill**

```
# docker stop cocky_brahmagupta
```

#### **Removing the container ID**

```
# docker rm e32aec51f29b
```

#### **Remove multiple container same time (Add first three letters)**

```
# docker rm ccd 045 952 54f
```

#### **Check image**

```
# docker images
```

#### **Rm image**

```
# docker rmi centos
```

#### **Pull image (Not to run)**

```
# docker pull ubuntu
```

```
# docker image
```

#### **Execute command in container without login in**

```
# docker run -d ubuntu sleep 1000
```

```
# docker exec containerID cat /etc/*release*
```

#### **run - PORT mapping**

```
# docker run -p 3306:3306 mysql
```

<https://hub.docker.com/explore>

Official Images

[https://hub.docker.com/search?image\\_filter=official&type=image](https://hub.docker.com/search?image_filter=official&type=image)

# Dockerfile

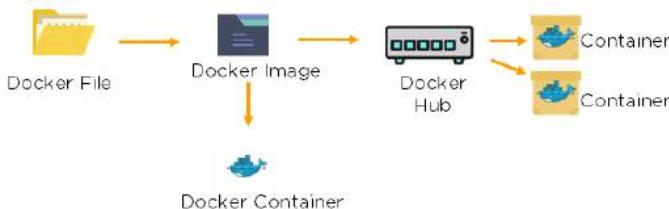
Tuesday, September 7, 2021 12:09 AM

## Dockerfile

- We can assemble an images using commands.
- You can put Multiple commands into a single document to fulfill a single task.

What is a Dockerfile used for?

- A Dockerfile is used for creating Docker images using the build command.
- With a Docker image, you can run the code to create Docker containers.
- After Docker image is built it uploaded to Docker registry.
- From the Docker registry, users can get the Docker image and build new containers whenever they want.



## Dockerfile Commands

**FROM** - Base parent image. Alpine version is the minimal docker image based of Alpine linux. Size 5MB

**RUN** - Runs linux commands, Install packages, create folders, users, assign permissions.

**ENV** - Sets Env variable. Multiple variables in single dockerfile.

**COPY** - Copy files and directories to the container.

**EXPOSE** - expose ports.

**ENTRYPOINT** - Provides command and arguments for an executing containers.

**CMD** - Provides command and arguments for an executing containers. Only one CMD

**VOLUME** - Create directory mount point to access and store persistent data

**WORKDIR** - Sets the working directory for instructions that follow

**LABLE** - Provides metadata like maintainer

**ADD** - Copies files and Dir to container. Also can unpack compressed files.

**ARG** - Define build time variables.

build command is used to create an image from the Dockerfile.

```
$ docker build
```

You can name your image as well.

```
$ docker build -t my-image
```

If your Dockerfile is placed in another path,

```
$ docker build -f /path/to/a/Dockerfile .
```

## COPY vs. ADD

Both commands serve a similar purpose, to copy files into the image.

- **COPY** - let you copy files and directories from the host.
- **ADD** - does the same. Additionally it lets you use URL location and unzip files into image.

## ENTRYPOINT vs. CMD

- **CMD** -

you can set a default command

which will be executed only when you run a container without specifying a command.

If a Docker container runs with a command, the default command will be ignored.

```
docker run -d -p 5003:5000 mslab/hello-world-python:0.0.2.RELEASE ping google.com
```

- **ENTRYPOINT** - allows you to configure a container that will run as an executable.

ENTRYPOINT command and parameters are not ignored when Docker container runs with command line parameters.

CMD - It will execute when you run a container without specifying a command. It will ignore

ENTP- ENTRYPOINT command and parameters are not ignored when Docker container runs with command line parameters.

## VOLUME

You declare **VOLUME** in your Dockerfile to denote where your container will write application data. When you run your container using **-v** you can specify its mounting point.

```
FROM debian:stretch-slim
```

```
# add our user and group first to make sure their IDs get assigned consistently, regardless of whatever dependencies get added
```

```
RUN groupadd -r mysql && useradd -r -g mysql mysql
```

```
RUN apt-get update && apt-get install -y --no-install-recommends gnupg dirmngr && rm -rf /var/lib/apt/lists/*
```

```
RUN mkdir /docker-entrypoint-initdb.d
```

```
ENV MYSQL_MAJOR 8.0
```

```
ENV MYSQL_VERSION 8.0.15-1debian9
```

```
VOLUME /var/lib/mysql
```

```
# Config files
```

```
COPY config/ /etc/mysql/
```

```
COPY docker-entrypoint.sh /usr/local/bin/
```

```
RUN ln -s /usr/local/bin/docker-entrypoint.sh /entrypoint.sh      # backwards compat
```

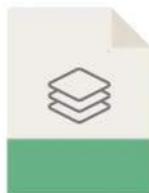
```
ENTRYPOINT ["docker-entrypoint.sh"]
```

```
EXPOSE 3306 33060
```

```
CMD ["mysqld"]
```



Dockerfile  
(Build)



Docker Image  
(Ship)



Containers  
(Run)

- A stack of multiple layers created from Dockerfile Instructions
- Each layer apart from the top one is R/O
- The top layer is R/W type
- Recognized by name or Image ID
- They are pushed to and can be pulled from Docker Hub

# Docker-Compose

Isnin, 6 September 2021 2:42 PTG

## Docker-compose

- For multiple container applications we can use the [docker-compose tool](#).
  - It can manage multiple containers and define the dependent services.
  - We can just define the application environment with a [Dockerfile](#), so it can be reproduced anywhere.
- 
- We need to define the services that make up the application in docker-compose.yml so they can be run together in an isolated environment.
  - Then you can Run docker-compose commands to run or stop the container or if you want to deploy/undeploy the application.
- 
- We need a docker-compose.yml file to write the services.
  - In a Dockerfile, we defined the environment of the application, and in docker-compose file we write down the other properties of services,
  - Like which service will run on which port, which service will be dependent on other services, which port will be forwarded to other port for public access, define network, cluster applications, etc.

## How do you run multiple containers using a single service?

- It is possible to run multiple containers as a single service with [Docker Compose](#).
- Here, each container runs in isolation but can interact with each other.
- All Docker Compose files are YAML files.



```
# nano Dockerfile
```

```
FROM centos

RUN yum -y install openssh-server

RUN useradd remote_user && \
    echo "1234" | passwd remote_user --stdin && \
    mkdir /home/remote_user/.ssh && \
    chmod 700 /home/remote_user/.ssh

COPY remote-key.pub /home/remote_user/.ssh/authorized_keys

RUN chown remote_user:remote_user -R /home/remote_user/.ssh/ && \
    chmod 600 /home/remote_user/.ssh/authorized_keys

RUN ssh-keygen -A

CMD /usr/sbin/sshd -D
```

```
=====
```

## Docker-compose

```
version: '3'
services:
  jenkins:
    container_name: jenkins
    image: jenkins/jenkins
    ports:
      - "8080:8080"
    volumes:
      - $PWD/jenkins_home:/var/jenkins_home
    networks:
      - net
  remote_host:
    container_name: remote-host
    image: remote-host
    build:
      context: centos7          # Directory that you have created
    networks:
      - net
networks:
  net:
```

# Wordpress- MySQL

Monday, October 25, 2021 10:41 PM

## [docker-compose.yml](#)

```
version: '3.3'

services:
  db:
    image: mysql:5.7
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: somewordpress
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress

  wordpress:
    depends_on:
      - db
    image: wordpress:latest
    ports:
      - "8000:80"
    restart: always
    environment:
      WORDPRESS_DB_HOST: db:3306
      WORDPRESS_DB_USER: wordpress
      WORDPRESS_DB_PASSWORD: wordpress
      WORDPRESS_DB_NAME: wordpress
  volumes:
    db_data: {}
```

## [mysql\\_docker-compose.yml](#)

```
environment:
  MYSQL_DATABASE: 'db'
  #So you don't have to use root, but you can if you like
  MYSQL_USER: 'root'
  #You can use whatever password you like
  MYSQL_PASSWORD: 'admin'
  #Password for root access
  MYSQL_ROOT_PASSWORD: 'admin'
ports:
  #<Port exposed> : < MySQL Port running inside container>
  - '3306:3306'
expose:
  #Opens port 3306 on the container
  - '3306'
  #Where our data will be persisted
volumes:
  - my-db:/var/lib/mysql
#Names our volume
volumes:
  my-db:
```

```
$ docker run -it --link mysql_database:mysql --rm mysql sh -c 'exec mysql -h"$MYSQL_PORT_3306_TCP_ADDR" -P"$MYSQL_PORT_3306_TCP_PORT" -uroot -p"$MYSQL_ENV_MYSQL_ROOT_PASSWORD"'
```

```
# docker-compose up -d  
# docker-compose config --services  
# docker-compose images  
# docker-compose logs --tail=10  
# docker-compose ps  
# docker-compose top  
# docker-compose down
```

# Docker Compose CLI Cheat sheet

## docker-compose build

**docker-compose build [options] [SERVICE...]**  
Use: Builds docker images for the listed services in compose file.  
Flag: `--no-cache` avoids image cache while building the docker image.

**docker-compose build --pull nginx-service**  
It will pull the latest version of nginx image and build the latest docker image for nginx-service.

## docker-compose run

**docker-compose run [options] SERVICE [COMMAND] [ARGS...]**  
Use: Runs a one-time command in a new container with same configurations of the mentioned service.  
Flag: `--entrypoint CMD` overrides the entrypoint of the image.

**docker-compose run ubuntu-service bash**  
It will run a bash session in a new container, with the same configurations of running ubuntu-service.

## docker-compose ps

**docker-compose ps [options] [SERVICE...]**  
Use: Lists out existing containers along with their properties.  
Flag: `-q / --quiet` shows only the list of container IDs.

**docker-compose ps --all**  
It will display all running as well as stopped containers initiated by docker compose with containers' properties such as container IDs, default execution commands, state of container, etc.

## docker-compose down

**docker-compose down [options]**  
Use: Stops all running containers, images, volumes, networks initiated by docker-compose up command.  
Flag: `--remove-orphans` removes containers for services not defined in compose file.

## docker-compose down -v

It will remove both named volumes (mentioned in the compose file) and anonymous volumes attached to containers.

## docker-compose kill

**docker-compose kill [options] [SERVICE...]**  
Use: Forces running containers to stop with SIGKILL.  
Flag: `-s [SIGNAL_NAME]` passes different types of signals to running containers.

## docker-compose kill -s SIGTERM

It will send a termination signal to all the compose initiated running containers.

## docker-compose up

**docker-compose up [options] [SERVICE...]**  
Use: Builds, (re)creates, starts, and attaches to containers for a service.  
Flag: `-d / --detach` runs containers in detach mode.

## docker-compose up -d

It will start containers in background and keeps them running.

## docker-compose help

**docker-compose help COMMAND**  
Use: Shows user instructions and details for a particular command.

## docker-compose help up

It will show usage instruction and option details for up command.

## docker-compose exec

**docker-compose exec [options] SERVICE [COMMAND] [ARGS...]**  
Use: Runs new commands in a running service container with default pseudo-TTY allocation.  
Flag: `--privileged` gives extended privileges to processes.

**docker-compose exec ubuntu-service bash**  
It will run a new bash session in the default directory of the running ubuntu-service container.

## docker-compose logs

**docker-compose logs [options] [SERVICE...]**  
Use: Displays log output from services.  
Flag: `--tail="10"` shows the last 10 lines of logs from each containers.

**docker-compose logs -t >> compose-logs.txt**  
It will get logs of all compose initiated containers with their respective timestamps and save it to the "compose-logs.txt" file.

## docker-compose rm

**docker-compose rm [options] [SERVICE...]**  
Use: Removes all stopped containers for services.  
Flag: `-v` removes any anonymous volume attached to the containers.

## docker-compose rm --force

It will remove all the stopped service containers without asking for user's approval.

# Storage - Volume -Mount drive to docker

Friday, September 24, 2021 1:10 PM

you don't need volumes-from unless it is ANOTHER volume in addition to the named volume

the volume create has nothing to do with a container

```
docker volume create --name fred /media/pi/mydrive      <--- only do once  
docker run -v fred:/var/www/html/
```

## 1. Anonymous volumes

If we run the following docker-compose.yml file, an anonymous volume will be created. If we restart our container, the data will be visible, but not after we remove the container. Also, it's not accessible by other containers. It is helpful if we want to persist data temporarily. These volumes are created inside /var/lib/docker/volume local host directory.

## 2. Named volumes

Named volumes can persist data after we restart or remove a container. Also, it's accessible by other containers. These volumes are created inside /var/lib/docker/volume local host directory.

## 3. Bind mounts

Bind mounts can persist data after we restart or remove a container. As we can see, named volumes and bind mounts are the same, except the named volumes can be found under a specific host directory, and bind mounts can be any host directory.

```
docker volume --help
```

Commands:

```
create Create a volume  
inspect Display detailed information on one or more volumes  
ls List volumes  
prune Remove all unused local volumes  
rm Remove one or more volumes
```

```
# Create a volume  
docker volume create test-vol  
# test-vol# Inspect a volume
```

```
docker inspect test-vol
```

```
# [
```

```
# {
```

```
# "CreatedAt": "2021-07-17T07:23:25Z",
```

```
#     "Driver": "local",

#     "Labels": {},

#     "Mountpoint": "/var/lib/docker/volumes/test-vol/_data",

#     "Name": "test-vol",

#     "Options": {},

#     "Scope": "local"

# }
```

```
# ]# List all volumes
```

```
docker volume create test-vol-2
```

```
docker volume ls
```

```
# DRIVER VOLUME NAME
```

```
# local test-vol
```

```
# local test-vol-2# Remove all volumes
```

```
docker volume prune
```

```
# WARNING! This will remove all local volumes not used by at least one container.
```

```
# Are you sure you want to continue? [y/N] y
```

```
# Deleted Volumes:
```

```
# test-vol

# test-vol-2# Remove volumes

docker volume create test-vol-3

docker volume rm test-vol-3

# test-vol-3docker volume create test-vol-4

docker volume create test-vol-5

docker volume rm test-vol-4 test-vol-5

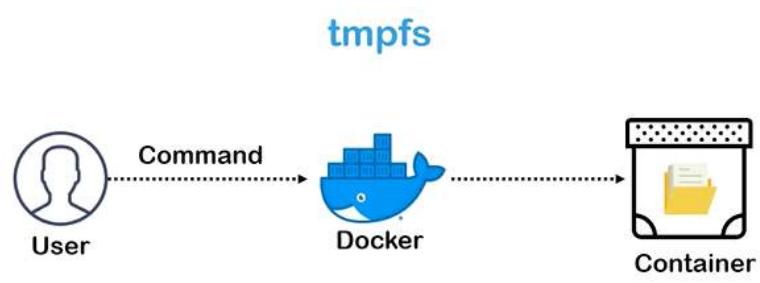
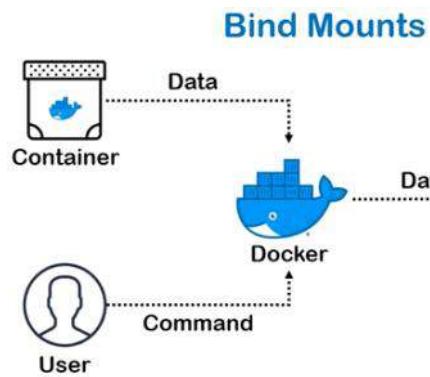
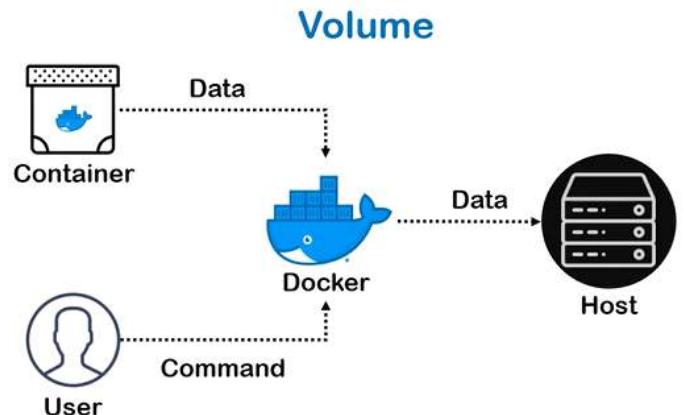
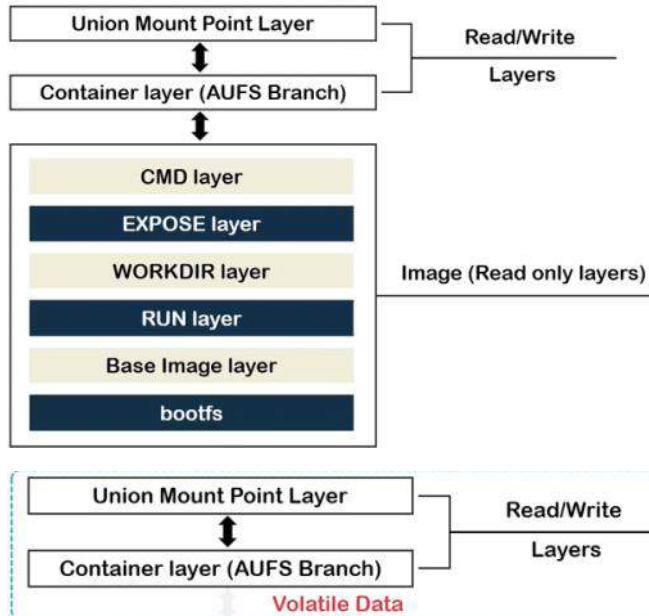
# test-vol-4

# test-vol-5
```

From <<https://towardsdatascience.com/the-complete-guide-to-docker-volumes-1a06051d2cce>>

# Docker storage

Monday, October 25, 2021 9:32 PM



## Create Volume

```
# docker volume create vol-busybox
```

## Adding volume to container

```
# docker run -d --volume vol-busybox:/tmp test_nginx
# docker run -v keysvolume:/data --user root uidgen
```

## List Volume

```
[root@docker dockertech]# docker volume ls
DRIVER    VOLUME NAME
local    vol-busybox
```

## Filter the Volume

```
# docker volume ls --filter "dangling=true"
```

## Check details inspect

```
# docker volume inspect vol-busybox
```

## Remove the volume

```
# docker volume rm vol-busybox
```

## Creating Volume with container

```
# docker run -itd --name ubuntu1 --volume vol-ubuntu:/var/log ubuntu:latest
```

**Volume details**

```
# docker container inspect --format "{{json .Mounts}}" ubuntu1 | python -m json.tool
```

**Volume path on host**

```
# cd /var/lib/docker/volumes/  
[root@docker volumes]# ls  
backingFsBlockDev metadata.db vol-ubuntu
```

# Networking

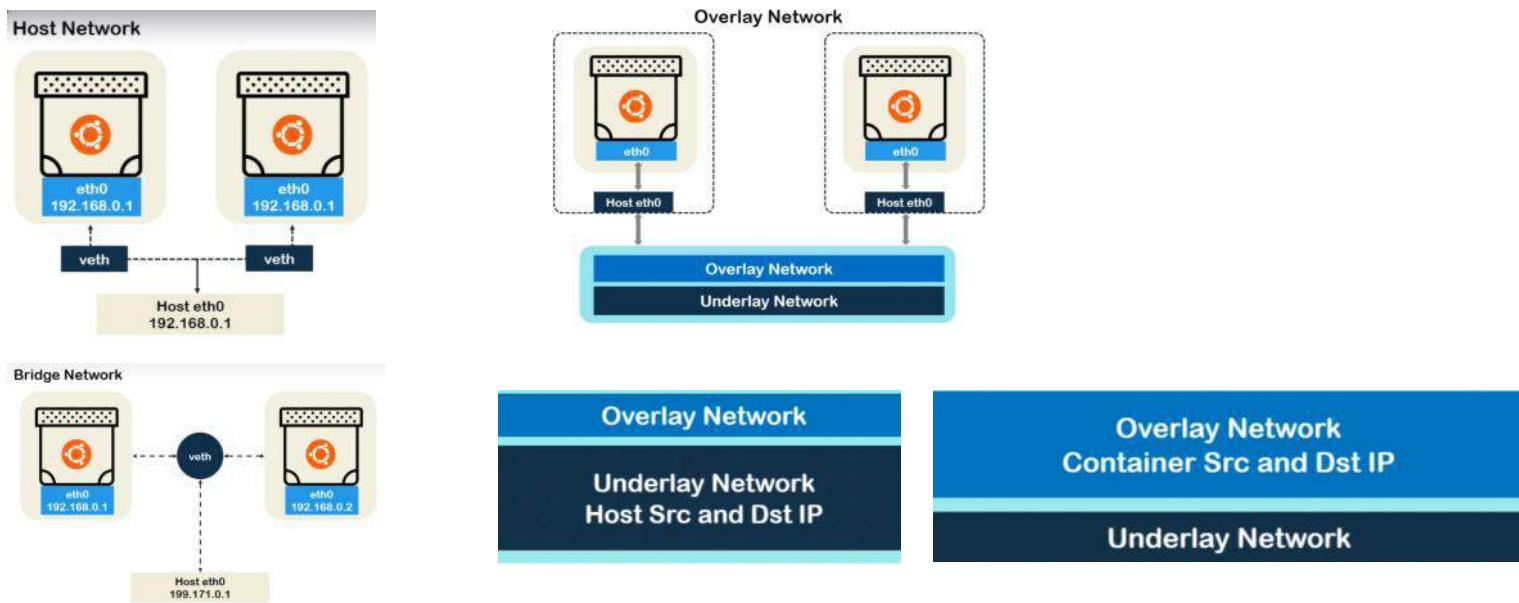
Thursday, October 21, 2021 9:39 PM

## Docker Default Networking (docker0)

- A default bridge network named docker0 is created.
- Container is automatically attached to this network, unless a custom network is specified.
- Besides docker0, two other networks get created automatically by Docker:
  - host (no isolation between host and containers) on this network, to the outside world they are on the same network
  - none (attached containers run on container-specific network stack).

## Docker Network Types

Network types: **bridge**, **overlay**, and **macvlan**.



## Bridge Networks

- common network type.
- It is limited to containers within a single host running the Docker engine.
- Easy to create, manage and troubleshoot.
- Containers on bridge network to communicate, The port mapping needs to be configured.
- Docker container running a web service on port 80
- Container is attached to the bridge network on a private subnet
- A port on the host system like 8000 needs to be mapped to port 80 on the container for outside traffic to reach the web service.

To create a bridge network named my-bridge-net, pass the argument `bridge` to the `-d` (driver) parameter  
\$ docker network create -d bridge my-bridge-net

## Overlay Networks

- It uses software virtualization to create additional layers of network abstraction running on top of a physical network.
- overlay network driver is used for multi-host network communication.
- This driver utilizes Virtual Extensible LAN (VXLAN) technology which provide portability between cloud, on-premise and virtual environments.
- VXLAN solves common portability limitations by extending layer 2 subnets across layer 3 network boundaries, hence containers can run on foreign IP subnets.

To create an overlay network named my-overlay-net, you'll also need the `--subnet` parameter to specify the network block that Docker will use to assign IP addresses to the containers:  
\$ docker network create -d overlay --subnet=192.168.10.0/24 my-overlay-net

## Macvlan Networks

- it is used to connect Docker containers directly to the host network interfaces through layer 2 segmentation.
- No use of port mapping or network address translation (NAT) is needed and containers can be assigned a public IP address which is accessible from the outside world.
- Latency in macvlan networks is low since packets are routed directly from Docker host network interface controller (NIC) to the containers.

- Note that macvlan has to be configured per host, and has support for physical NIC, sub-interface, network bonded interfaces and even teamed interfaces.
- Traffic is explicitly filtered by the host kernel modules for isolation and security.
- 
- To create a macvlan network named macvlan-net, you'll need to provide a --gateway parameter to specify the IP address of the gateway for the subnet, and a -o parameter to set driver specific options. In this example, the parent interface is set to eth0 interface on the host:

```
$ docker network create -d macvlan \
--subnet=192.168.40.0/24 \
--gateway=192.168.40.1 \
-o parent=eth0 my-macvlan-net
```

## Common Operations

- **Inspect a network:** To see a specific network's configuration details like subnet information, network name, IPAM driver, network ID, network driver, or connected containers, use the docker network inspect command.
- **List all networks:** Run docker network ls to display all networks (along with their type and scope) present on the current host.
- **Create a new network:** To create a new network, use the docker network create command and specify if it's of type bridge (default), overlay or macvlan.
- **Run or connect a container to a specific network:** Note first of all, the network must exist already on the host. Either specify the network at container creation/startup time (docker create or docker run) with the --net option; or attach an existing container by using the docker network connect command. For example:  
docker network connect my-network my-container
- **Disconnect a container from a network:** The container must be running to disconnect it from the network using the docker network disconnect command.
- **Remove an existing network:** A network can only be removed using the command docker network rm if there are no containers attached to it. When a network is removed, the associated bridge will be removed as well.

From <<https://www.aquasec.com/cloud-native-academy/docker-container/docker-networking/>>

# Creating network

Monday, October 25, 2021 9:05 PM

## Creating bridge

```
# docker network create --driver bridge my-bridge
```

## Create bridge with subnet

```
# docker network create --driver bridge --subnet=192.168.0.0/16 --ip-range=192.168.5.0/24 my-bridge-1
```

## List network

```
# docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
7891c6a9cd6c	bridge	bridge	local

## Filter the search

```
# docker network ls --filter driver=bridge
```

## Connect container to network

```
# docker network connect my-bridge-1 cont_run-env
```

## Confirm with inspect

```
# docker inspect cont_run-env
```

## Adding host network during container run

```
# docker container run -itd --network host --name test_nginx nginx
```

## Check Port

```
# docker container port test_nginx
```

## Bridge network inspect details

```
# docker network inspect bridge
```

```
# docker network inspect my-bridge-1
```

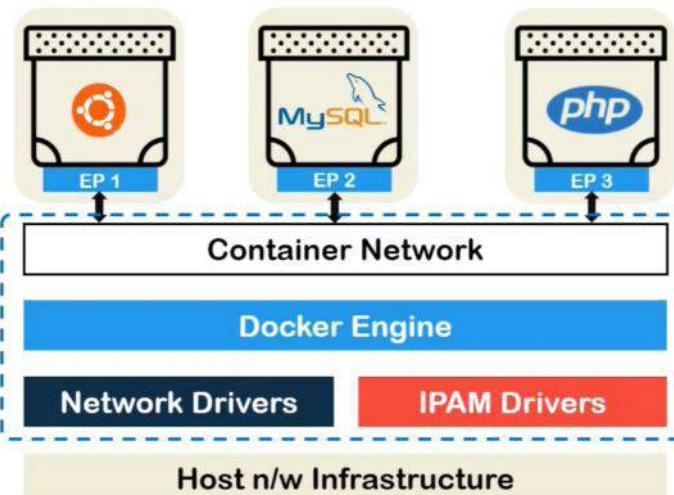
## Disconnect network from container

```
# docker network disconnect my-bridge-1 cont_run-env
```

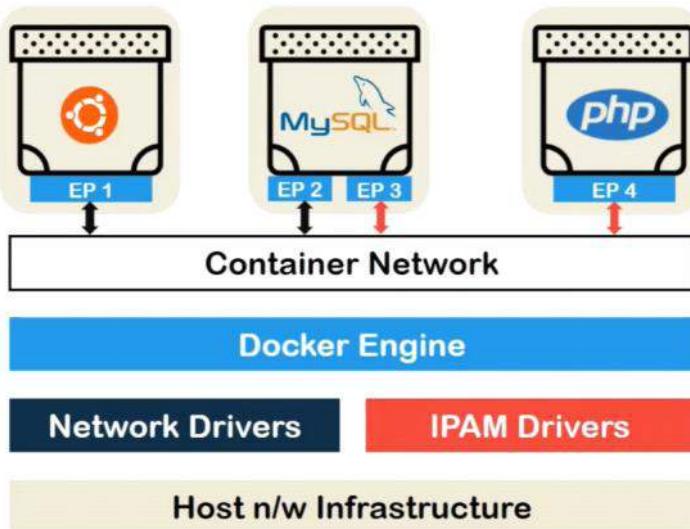
# Docker Network Drivers

Monday, October 25, 2021 8:48 PM

- ✓ Piece of software that enables networking of containers.
- ✓ Responsible for invoking a network inside the host or within the cluster.
- ✓ Native n/w drivers are shipped with Docker Engine.
- ✓ Remote n/w drivers are created and managed by 3<sup>rd</sup> party vendors or community.
- ✓ IP Address Management Drivers provide default subnets if not specified by admin.



Connected to multi host



# Swarm

Tuesday, September 7, 2021 1:08 AM

- It is possible to share Docker containers on different nodes with **Docker Swarm**.
- **Docker Swarm** is a tool that allows to create and manage a cluster of swarm nodes within the Docker platform.
- It consists of two types of nodes: a manager node and a worker node.

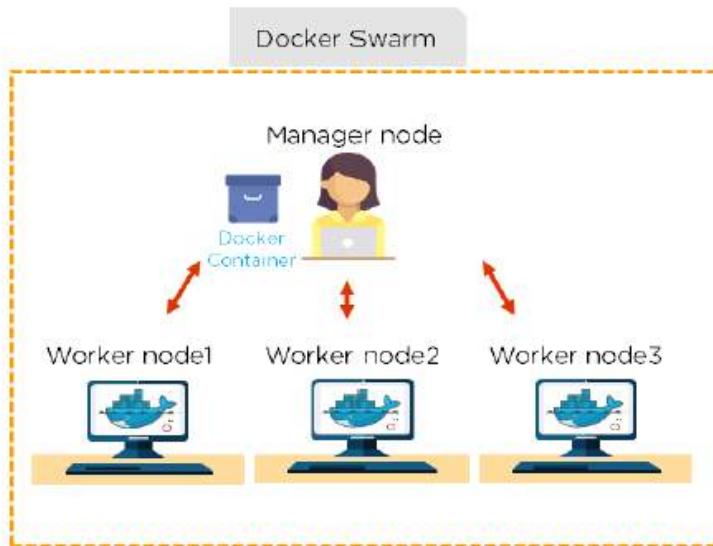
## Docker Swarm?

Docker Swarm is an **open-source container orchestration tool** that is integrated with the Docker engine and CLI.

If you want to use Docker Swarm, you should use the overlay network driver.

Using an overlay network enables the Swarm service by connecting multiple docker host daemons together.

Docker Swarm is native gathering for docker which helps you to a group of Docker hosts into a single and virtual docker host. It offers the standard docker application program interface.



## What are the commands used to create a **Docker swarm**?

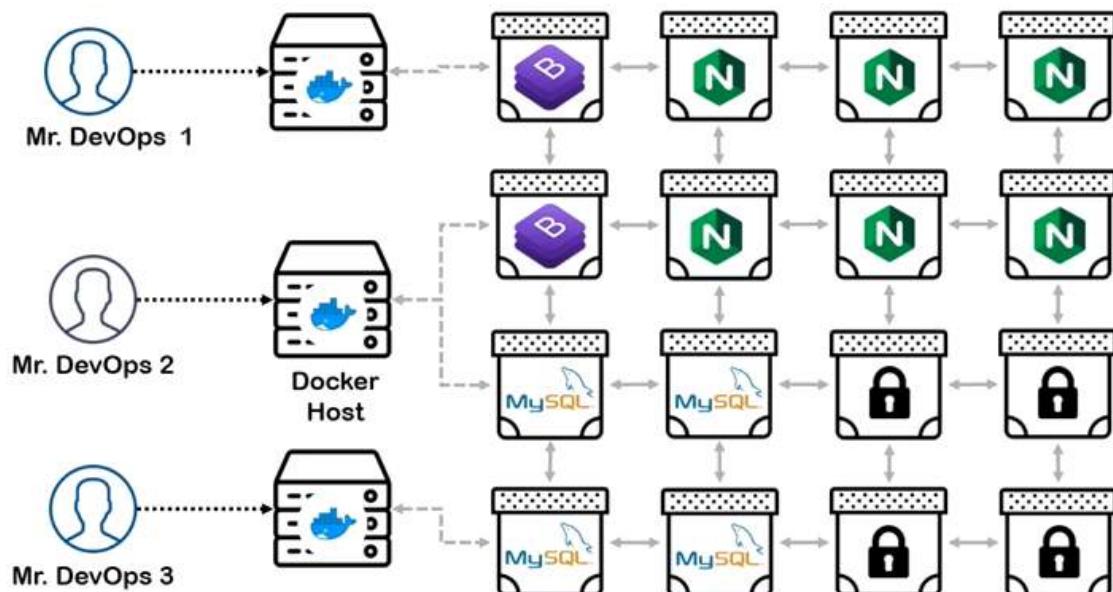
- Create a swarm on manager node.

```
Docker swarm init --advertise-addr <MANAGER-IP>
```

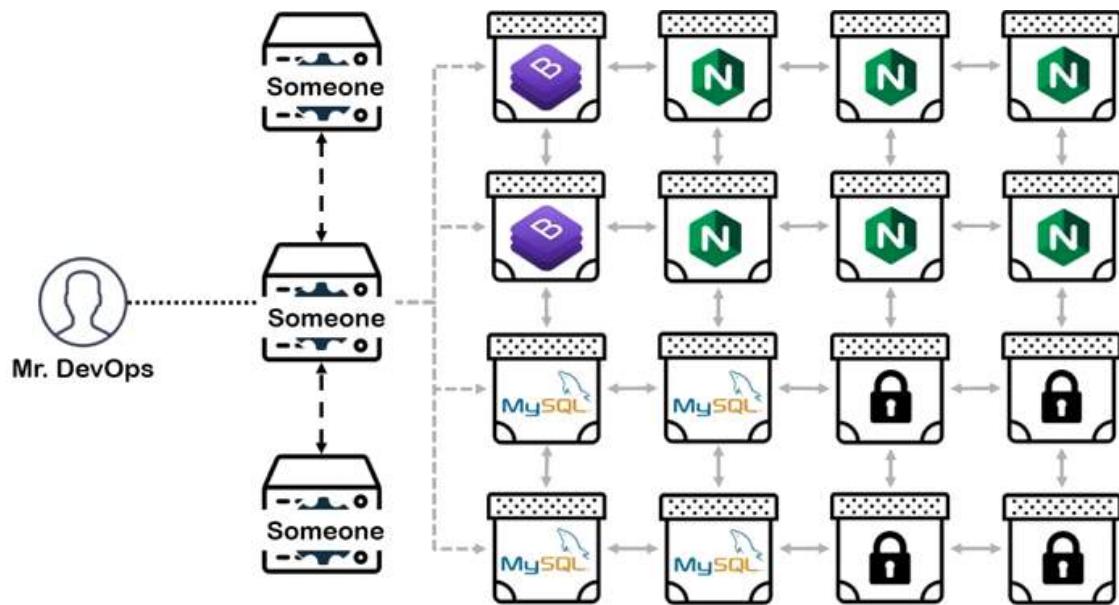
- Once you've created a swarm on your manager node, you can add worker nodes to your swarm.
- When a node is initialized as a manager, it immediately creates a token. In order to create a worker node, the following command (token) should be executed on the host machine of a worker node.

```
docker swarm join \ --token SWMTKN-1-49nj1cmql0jkz5s954yi3oex3nedyz0fb0xx14ie39trti4wxv-8vxv8rssmk743ojnwacrr2e7c \ 192.168.99.100:2377
```

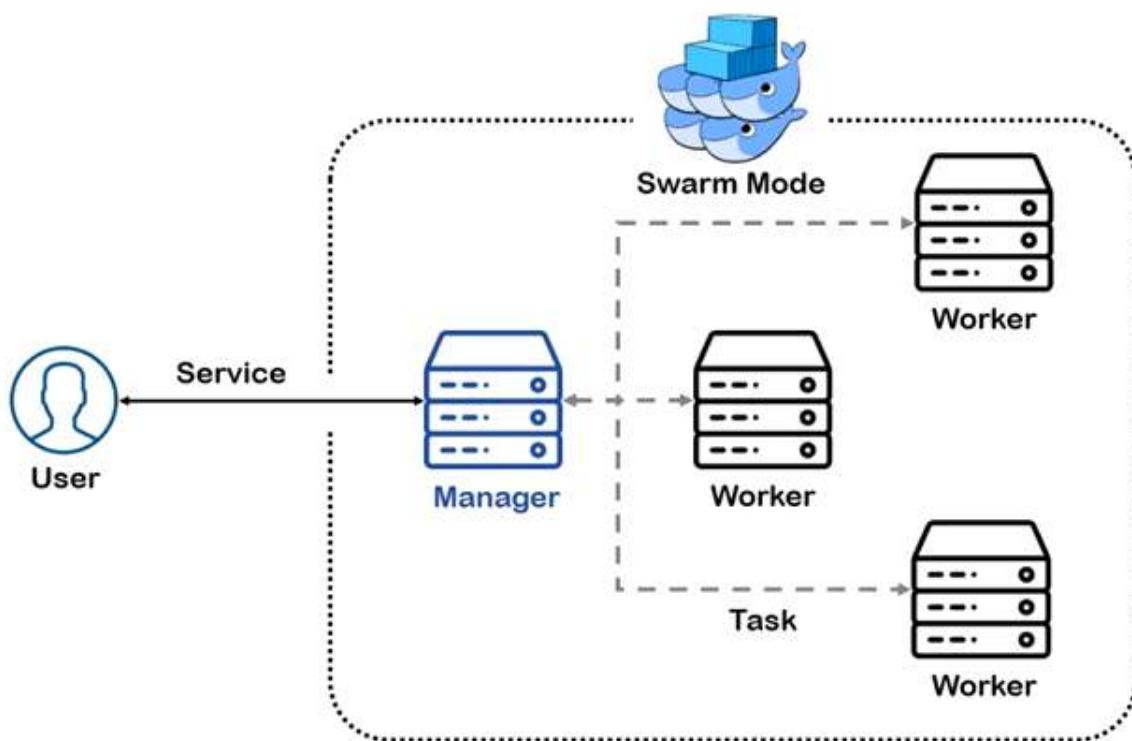
“ A Container Orchestrator is a tool used to provision, schedule and manage containers at large scale over one or more clusters of multiple hosts. ”



Cluster

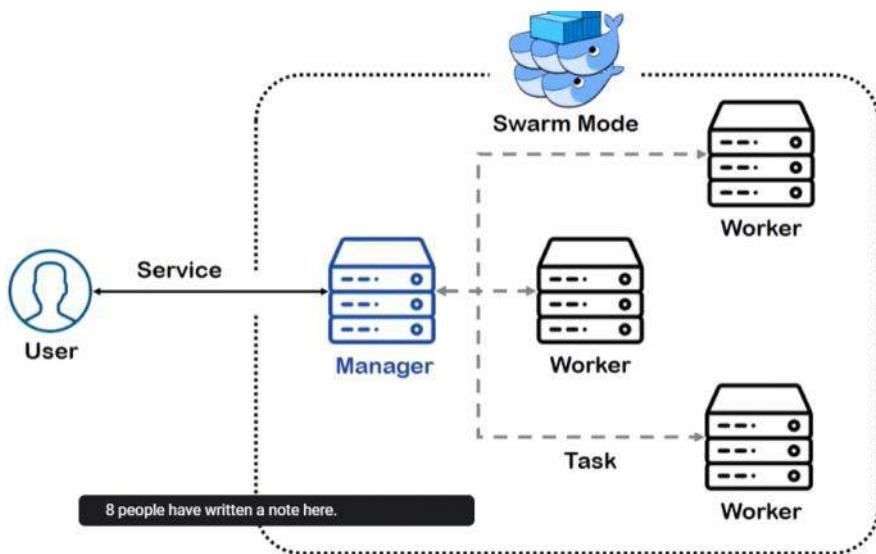


Docker Swarm



# Docker Swarm

Thursday, October 28, 2021 5:39 PM



## Manager is equipped with set of useful tools

- **HTTP API Endpoint** - Which makes it capable of serving our service request and creating objects out of those services.
- **Orchestrator** - It process tasks translated from services to workers
- **Allocator** - Which allocates internal cluster IPs to the workers and manager itself
- **Dispatcher** - Which decides which node will server which task and gives this information to the orchestrator
- **Scheduler** - The task provided by orchestrator are idle, They don't run soon as they get allocated. Scheduler signals workers to run the task which they have received, Also decide which task will run first.

# docker-machine

Khamis, 28 Oktober 2021 6:01 PTG

```
# nano /etc/apt/sources.list  
deb https://download.virtualbox.org/virtualbox/debian xenial contrib
```

Add GPG Key

<https://github.com/docker/machine/releases/>

## **Adding Manager and worker**

```
# docker-machine create --driver virtualbox manager  
# docker-machine create --driver virtualbox worker-1  
# docker-machine create --driver virtualbox worker-2
```

## **List the running docker machine**

```
# docker-machine ls
```

## **IP list**

```
# docker-machine ip manager
```

## **Docker machine Inspect**

```
# docker-machine inspect manager
```

## **SSH login to docker machine**

```
# docker-machine ssh manager
```

## **Initialize the Swarm on Manager machine**

```
# docker swarm init --advertise-addr 192.168.99.100  
Swarm initialized: current node (te0wzpgi64ubokpvak1aoyi7m) is now a manager.
```

## **Re-generate the token for worker nodes**

```
# docker swarm join-token worker
```

Login to Worker

```
# docker-machine ssh worker-2
```

## **And execute**

```
# docker swarm join --token SWMTKN-1-5qfzeqrdd6fy8vejv53z5x99t8r82fuvkczp2lkssgddu5mvd-34pahnv3oixzcg9fkhhigadsg0  
192.168.99.100:2377
```

This node joined a swarm as a worker.

## **Docker Inspect (Node ID, Hostname, Time stamp, Engine, Memory, CPU, Cert)**

```
# docker node inspect --pretty self
```

## **Create 3 Replica Web application container**

```
# docker service create --name web-server -p 8080:80 --replicas 3 nginx:latest
```

## **List Container**

```
# docker service ps web-server
```

## **Inspect Container**

```
# docker service inspect web-server
```

**Leave the swarm cluster from worker node**

```
# docker swarm leave
```

Node left the swarm.

**Delete the worker from Manager**

```
# docker node rm worker-2
```

**Scale the replica on manager**

```
# docker service scale web-server=6
```

**List the container**

```
# docker service
```

**Update the container**

```
# docker service update --image nginx:alpine web-server
```

**Inspect**

```
# docker node inspect --pretty web-server
```

**Remove web-server**

```
Docker service rm web-server
```

Docker GUI Repo

<https://github.com/docker/kitematic/releases>

Unzip Kitematic file

```
cd Downloads
```

```
sudo apt install unzip
```

```
sudo unzip Kitematic*.zip
```

```
sudo dpkg -i Kitematic*.deb
```

**If you get dependencies error then run the below command**

```
sudo apt install -f
```

```
sudo dpkg -i Kitematic*.deb
```

Run Kitematic on Ubuntu 20.04

```
kitematic
```

# Docker Engine

Tuesday, September 7, 2021 1:15 AM

## Docker Engine?

Docker daemon or Docker engine represents the server.

The docker daemon and the clients should be run on the same or remote host, which can communicate through command-line client binary and full RESTful API.

Three Main components

### Docker Client

It's machine or medium through which we as users interact with docker.

Two basic ways interaction are Docker CLI and Docker API



### Docker Host

Which actually performs the task for containerization. It runs docker daemon. Which listens and performs action asked by docker client. Docker daemon builds Dockerfile and turns it into docker image.

### Docker Registry

Stores docker images

Docker Client talks to docker daemon where it passes the request and receives the result

# Docker Hub

Tuesday, September 7, 2021 1:11 AM

## Docker hub?

Docker hub is a **cloud-based registry** that which helps you to link to code repositories. It allows you to build, test, store your image in Docker cloud. You can also deploy the image to your host with the help of Docker hub.

 malkiats / repo-nginx

## Search

```
# docker search python:3.6  
# docker search --filter "is-official=true" register
```

## Login

```
# docker login
```

## Tag image to public repo

```
# docker tag nginx:latest malkiats/repo-nginx:test-nginx
```

## Push the image to docker hub

```
# docker image push malkiats/repo-nginx:test-nginx
```

# Docker Images

Monday, October 25, 2021 7:17 PM

## Image details

```
# docker images inspect ubuntu:latest

# docker image inspect --format "{{.RepoTags}} : {{.RepoDigests}} " img-nginx

# docker image inspect --format "{{json .Config}}" img-nginx > inspect.txt

[root@docker dockertech]# docker image history ubuntu:16.04
IMAGE      CREATED     CREATED BY          SIZE    COMMENT
b6f507652425  7 weeks ago  /bin/sh -c #(nop) CMD ["/bin/bash"]    0B
<missing>  7 weeks ago  /bin/sh -c mkdir -p /run/systemd && echo 'do...  7B
<missing>  7 weeks ago  /bin/sh -c rm -rf /var/lib/apt/lists/*    0B
<missing>  7 weeks ago  /bin/sh -c set -xe && echo '#!/bin/sh' > /...  745B
<missing>  7 weeks ago  /bin/sh -c #(nop) ADD file:11b425d4c08e81a3e...  135MB
```

# Docker Image and Container

Tuesday, September 7, 2021 1:05 AM

**Explain the differences between Docker images and Docker containers.**

Docker Images	Docker Container
Docker images are templates of Docker containers	Containers are runtime instances of a Docker image
An image is built using a Dockerfile	Containers are created using Docker images
It is stored in a Docker repository or a Docker hub	They are stored in the Docker daemon
The image layer is a read-only filesystem	Every container layer is a read-write filesystem

A Docker image is a template of instructions, which is used to create containers.

Docker container is an executable package of an application and its dependencies together.

Docker registry is a service to host and distribute Docker images among users.

## **Docker image?**

The Docker image help to create Docker containers.

- You can create the Docker image with the build command.
- Due to this, it creates a container that starts when it begins to run.
- Every docker images are stored in the Docker registry.

# CNM

Tuesday, September 7, 2021 1:16 AM

## **CNM?**

CNM stands for Container Networking Model.

It is a standard or specification from Docker, Inc. that forms the basis of container networking in a Docker environment. This docker's approach provides container networking with support for multiple network drivers.

# Expose and Publish

Tuesday, September 7, 2021 1:02 AM

## What is the purpose of the expose and publish commands in Docker?

### Expose

- Expose is an instruction used in Dockerfile.
- It is used to expose ports within a Docker network.
- It is a documenting instruction used at the time of building an image and running a container.
- Expose is the command used in Docker.
- Example: Expose 8080

### Publish

- Publish is used in a Docker run command.
- It can be used outside a Docker environment.
- It is used to map a host port to a running container port.
- --publish or -p is the command used in Docker.
- Example: docker run -d -p 0.0.0.80:80

Now, let's have a look at the DevOps interview questions for continuous monitoring.

# Registry & Repository

Tuesday, September 7, 2021 1:03 AM

## Registry and a repository?

Registry	Repository
A Docker registry is an open-source server-side service used for hosting and distributing Docker images	The repository is a collection of multiple versions of Docker images

In a registry, a user can distinguish between Docker images with their tag names	It is stored in a Docker registry
Docker also has its own default registry called Docker Hub	It has two types: public and private repositories

### Registries

- Public Registry
- Private Registry

Docker's public registry is called Docker hub, which allows you to store images privately. In Docker hub, you can store millions of images.

### Docker Trusted Registry?

Docker Trusted Registry is the **enterprise-grade image storage toll** for Docker. You should install it after your firewall so that you can securely manage the Docker images you use in your applications.

# Scaling

Tuesday, September 7, 2021 1:14 AM

## **Scaling your Docker containers**

The Docker containers can be scaled to any level starting from a few hundred to even thousands or millions of containers.

The only condition for this is that **the containers need the memory and the OS at all times**, and there should not be a constraint when the Docker is getting scaled.

# Virtualization Hypervisor

Tuesday, September 7, 2021 1:13 AM

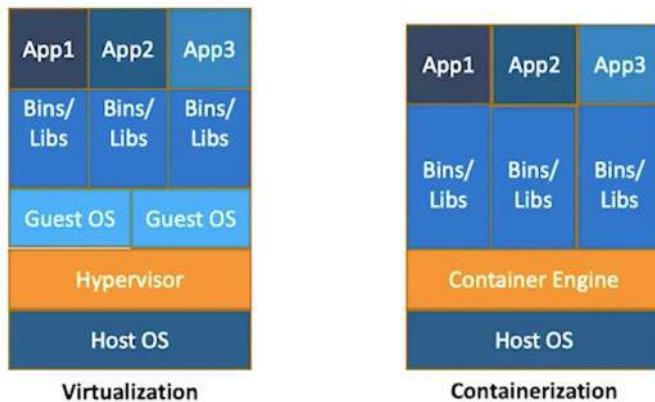
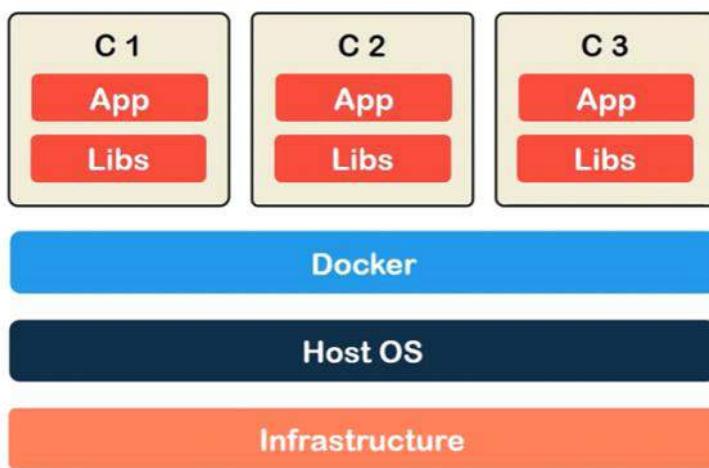
## Virtualization?

Virtualization is a method of logically dividing mainframes to allow multiple applications to run simultaneously.

However, this scenario changed when companies and open source communities were able to offer a method of handling privileged instructions. It allows multiple OS to run simultaneously on a single x86 based system.

## Hypervisor?

The hypervisor allows you to create a virtual environment in which the guest virtual machines operate. It controls the guest systems and checks if the resources are allocated to the guests as necessary.



## Docker More

Monday, August 30, 2021 7:35 PM

### **Memory-swap flag?**

Memory-swap is a modified flag that only has meaning if- **memory is also set.**

Swap allows the container to write express memory requirements to disk when the container has exhausted all the RAM which is available to it.

### **Monitor the docker in production environments?**

Docker states and Docker Events are used to monitoring docker in the production environment.

### **Run multiple containers using a single service?**

By using docker-compose, you can run multiple containers using a single service. All docker-compose files uses yaml language.

### **Volume mount types available in Docker?**

Bind mounts- It can be stored anywhere on the host system

### **Default logging driver under Docker?**

To configure the Docker daemon to default to a specific logging driver.

You need to set the value of log-driver to the name of the logging drive the daemon.jason.fie.

### **Docker Namespaces?**

The Namespace in Docker is a technique which **offers isolated workspaces called the Container.**

Namespaces also **offer a layer of isolation for the Docker containers.**

### **Components of Docker Architecture**

- Client
- Docker-Host
- Registry

### **Client?**

Docker provides **Command Line Interface tools to the client to interact with Docker daemon.**

### **Docker\_Host?**

It contains container, images, and Docker daemon.

It offers a complete environment to execute and run your application.

### **Multiple copies of Compose file on the same host?**

Compose uses the project name which allows you to create unique identifiers for all of a project's containers and other resources. To run multiple copies of a project, set a custom project name using the -a command-line option or using COMPOSE\_PROJECT\_NAME environment variable.

# Revert a commit

Monday, October 3, 2022 4:56 PM

This is call hotfix or quick fix

Whatever last commit was working fine so role back to that.

You can use Revert command

```
# git log  
# git revert id#####  
# git push origin master
```

# Branches

Tuesday, September 7, 2021 1:18 AM

## Branches Strategies

- Branching is a technique that makes a copy of the source code to create two versions that are developed separately. There are various forms of branching.
- By default when we create a repository (Example creating repo on GIT) it creates a branch called Master and all our commits are go and sit on the master branch.
- Apart from branches we can create our own branches and after creating our own branches we can remove the master branch
- It is not mandatory that you should maintain master branch it is just a default branch which comes while creating our repository

## Advantages of using branches.

- Branches helps us to avoid the problem when we deploy the code onto the system
- Branches is powerful SCM mechanism that allows developers to create separate line of development
- Branches makes developers to work parallel on separate lines of development.
- These lines of development are generally different product features.
- When development is complete on a branch, it is then merged into the main line of development.

## Feature Branch:

A feature branch is simply a separate branch in your Git repo that is used to implement a single feature in your project. Once the feature is complete, then you merged into master

## Release Branch:

Release branches contain production ready new features and bug fixes that come from stable develop branch. After finishing release branches, they get merged back into develop and master branches so as a result both of these branches will match each other eventually

## Hotfix Branch:

Maintenance or "hotfix" branches are used to quickly patch production releases.

Hotfix branches are a lot like release branches and feature branches except they're based on main instead of develop

### Creating a feature branch

When starting work on a new feature, branch off from the develop branch.

```
$ git checkout -b myfeature develop
Switched to a new branch "myfeature"
```

### Incorporating a finished feature on develop

Finished features may be merged into the develop branch to definitely add them to the upcoming release:

```
$ git checkout develop
Switched to branch 'develop'
$ git merge --no-ff myfeature
Updating ea1b82a..05e9557
(Summary of changes)
$ git branch -d myfeature
Deleted branch myfeature (was 05e9557).
$ git push origin develop
```

The --no-ff flag causes the merge to always create a new commit object, even if the merge could be performed with a fast-forward. This avoids losing information about the historical existence of a feature branch and groups together all commits that together added the feature. Compare:

# GIT use

Thursday, August 5, 2021 3:34 AM

## Configure User and Email

```
[root@server1 ~]# git config --global user.name "malkiats"  
[root@server1 ~]# git config --global user.email "msjanjua@live.co.uk"
```

## Create Folder

```
[root@server1 home]# cd GIT/
```

## Check config list

```
[root@server1 GIT]# git config --global --list  
user.name=malkiats  
user.email=msjanjua@live.co.uk
```

## Initialize empty Git repo

```
[root@server1 GIT]# git init git-jenkins  
Initialized empty Git repository in /home/GIT/git-jenkins/.git/
```

## Create folder

```
[root@server1 GIT]# cd git-jenkins/
```

## Copy or create file

```
[root@server1 GIT]# touch jenkinsfile
```

## Check GIT status

```
[root@server1 git-jenkins]# git status  
On branch master
```

No commits yet

Untracked files:

```
(use "git add <file>..." to include in what will be committed)  
jenkinsfile
```

nothing added to commit but untracked files present (use "git add" to track)

## Add new created file in Git repo

```
[root@server1 git-jenkins]# git add jenkinsfile
```

## Check status

```
[root@server1 git-jenkins]# git status  
On branch master
```

No commits yet

```
Changes to be committed:  
(use "git rm --cached <file>..." to unstage)  
new file: jenkinsfile
```

## Commit file with comment

```
[root@server1 git-jenkins]# git commit -m "Nothing to add"  
[master (root-commit) e577f64] Nothing to add  
1 file changed, 18 insertions(+)  
create mode 100644 jenkinsfile
```

## Create Public repository on your GIT Account

## Add remote GIT

```
[root@server1 git-jenkins]# git remote add origin https://github.com/malkiats/git-jenkins.git
```

## Push file to remote GIT repo

```
[root@server1 git-jenkins]# git push -u origin master
```

```
Username for 'https://github.com': malkiats
Password for 'https://malkiats@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 330 bytes | 330.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/malkiats/git-jenkins/pull/new/master
remote:
To https://github.com/malkiats/git-jenkins.git
 * [new branch]  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

ghp\_iyaiQssD7znTQ08MTvg91QhBwKkH7a0lVcXa

## Cloning to a specific folder

```
git clone <repo> <directory>
```

## Cloning a specific tag

```
git clone --branch <tag> <repo>
```

## git clone -branch

```
git clone -b new_feature git://remoterepository.git
```

## Git Alias

```
$ git config --global alias.co checkout
$ git config --global alias.br branch
$ git config --global alias.ci commit
$ git config --global alias.st status
```

Means

[alias]

- co = checkout
- br = branch
- ci = commit
- st = status

## Stashing your work

```
$ git status
```

On branch main

Changes to be committed:

**newfile:** style.css

```
Changes not staged for commit:
```

```
modified: index.html
```

```
$ git stash
```

```
Saved working directory and index state WIP on main: 5002d47 our new homepage  
HEAD is now at 5002d47 our new homepage
```

```
$ git status
```

```
On branch main
```

```
nothing to commit, working tree clean
```

## Re-applying your stashed changes

```
$ git status
```

```
On branch main
```

```
nothing to commit, working tree clean
```

```
$ git stash pop
```

```
On branch main
```

```
Changes to be committed:
```

```
newfile: style.css
```

```
Changes not staged for commit:
```

```
modified: index.html
```

```
Dropped refs/stash@{0} (32b3aa1d185dfe6d57b3c3cc3b32cbf3e380cc6a)
```

## git log

The `git log` command displays committed snapshots, Project History

## Tagging

```
git tag <tagname>
```

Tagging is generally used to capture a point in history that is used for a marked version release in GIT History

### git blame

This is used to examine specific points of a file's history and get context as to who the last author was that modified the line.

### git checkout

When you have found a commit reference to the point in history you want to visit, you can utilize the `git checkout` command to visit that commit. Git checkout is an easy way to "load" any of these saved snapshots onto your development machine.

### git clean

The `git clean` command operates on untracked files. Untracked files are files that have been created within your repo's working directory but have not yet been added to the repository's tracking index using the `git add` command.

### git revert

The `git revert` command can be considered an 'undo' type command

### git reset

The `git reset` command is a complex and versatile tool for undoing changes.

### git rm

The `git rm` command can be used to remove individual files or a collection of files. The primary function of `git rm` is to remove tracked files from the Git index

Git task	Notes	Git commands
<a href="#">Tell Git who you are</a>	Configure the author name and email address to be used with your commits. Note that Git strips some characters (for example trailing periods) from user.name.	git config --global user.name "Sam Smith" git config --global user.email sam@example.com
<a href="#">Create a new local repository</a>		git init
<a href="#">Check out a repository</a>	Create a working copy of a local repository:  For a remote server, use:	git clone /path/to/repository  git clone username@host:/path/to/repository
<a href="#">Add files</a>	Add one or more files to staging (index):	git add git add *
<a href="#">Commit</a>	Commit changes to head (but not yet to the remote repository):  Commit any files you've added with git add, and also commit any files you've changed since then:	git commit -m "Commit message"  git commit -a
<a href="#">Push</a>	Send changes to the main branch of your remote repository:	git push origin main
<a href="#">Status</a>	List the files you've changed and those you still need to add or commit:	git status
<a href="#">Connect to a remote repository</a>	If you haven't connected your local repository to a remote server, add the server to be able to push to it:  List all currently configured remote repositories:	git remote add origin  git remote -v
<a href="#">Branches</a>	Create a new branch and switch to it:  Switch from one branch to another:  List all the branches in your repo, and also tell you what branch you're currently in:  Delete the feature branch:  Push the branch to your remote repository, so others can use it:  Push all branches to your remote repository:  Delete a branch on your remote repository:	git checkout -b  git checkout  git branch  git branch -d  git push origin  git push --all origin  git push origin :  git merge
<a href="#">Update from the remote repository</a>	Fetch and merge changes on the remote server to your working directory:  To merge a different branch into your active branch:  View all the merge conflicts: View the conflicts against the base <a href="#">file:Preview</a> changes, before merging:  After you have manually resolved any conflicts, you mark the changed file:	git pull  git merge  git diff git diff --base git diff  git add
Tags	You can use tagging to mark a significant changeset, such as a release:  CommitId is the leading characters of the changeset ID, up to 10, but must be unique. Get the ID using:  Push all tags to remote repository:	git tag 1.0.0  git log  git push --tags origin
<a href="#">Undo local changes</a>	If you mess up, you can replace the changes in your working tree with the last content in head: Changes already added to the index, as well as new files, will be kept.  Instead, to drop all your local changes and commits, fetch the latest history from the server and point your local main branch at it, do this:	git checkout --  git fetch origin git reset --hard origin/main
Search	Search the working directory for foo():	git grep "foo()"

From <<https://www.atlassian.com/git/tutorials/svn-to-git-prepping-your-team-migration>>

# Rebase Reset Revert

Monday, October 18, 2021 1:58 PM

Rebase is another way to integrate changes from one branch to another. Rebase compresses all the changes into a single “patch.” Then it integrates the patch onto the target branch. Unlike merging, rebasing flattens the history because it transfers the completed work from one branch to another.

**git reset** is a command to "fix-uncommitted mistakes" and

**git revert** is a command to "fix-commited mistake". It means if we have made some error in some change and committed and pushed the same to git repo, then git revert is the solution.

# Git Merge

Wednesday, September 8, 2021 9:56 PM

## Git Merge Tutorial



```
$ git merge <branch_name> --no-ff -m "<commit message>"  
$ git checkout --ours -- file1.txt  
$ git checkout --theirs -- file2.txt  
$ git merge incomingBranch  
$ git merge --abort
```



# Version Control

Tuesday, September 7, 2021 1:19 AM

## Version Control?

- Know as source control
- Helps to manage changes to source code over time.
- Keeps track of every modification to the code
- If any mistake happen developer can turn back the clock and compare earlier versions of the code to help fix the mistake.
- Software developer continually writing new source code and changing existing source code.
- The code is organized in folder structure or file tree.

# Cheat Sheet

Tuesday, September 7, 2021 1:21 AM

## Create a Repository

From scratch -- Create a new local repository

```
$ git init [project name]
```

Download from an existing repository

```
$ git clone my_url
```

## Observe your Repository

List new or modified files not yet committed

```
$ git status
```

Show the changes to files not yet staged

```
$ git diff
```

Show the changes to staged files

```
$ git diff --cached
```

Show all staged and unstaged file changes

```
$ git diff HEAD
```

Show the changes between two commit ids

```
$ git diff commit1 commit2
```

List the change dates and authors for a file

```
$ git blame [file]
```

Show the file changes for a commit id and/or file

```
$ git show [commit]:[file]
```

Show full change history

```
$ git log
```

Show change history for file/directory including diffs

```
$ git log -p [file/directory]
```

## Working with Branches

List all local branches

```
$ git branch
```

List all branches, local and remote

```
$ git branch -av
```

Switch to a branch, my\_branch, and update working directory

```
$ git checkout my_branch
```

Create a new branch called new\_branch

```
$ git branch new_branch
```

Delete the branch called my\_branch

```
$ git branch -d my_branch
```

Merge branch\_a into branch\_b

```
$ git checkout branch_b
```

```
$ git merge branch_a
```

Tag the current commit

```
$ git tag my_tag
```

## Make a change

Stages the file, ready for commit

```
$ git add [file]
```

Stage all changed files, ready for commit

```
$ git add .
```

Commit all staged files to versioned history

```
$ git commit -m "commit message"
```

Commit all your tracked files to versioned history

```
$git commit -am "commit message"
```

Unstages file, keeping the file changes

```
$ git reset [file]
```

Revert everything to the last commit

```
$ git reset --hard
```

## Synchronize

Get the latest changes from origin (no merge)

```
$ git fetch
```

Fetch the latest changes from origin and merge

```
$ git pull
```

Fetch the latest changes from origin and rebase

```
$ git pull --rebase
```

Push local changes to the origin

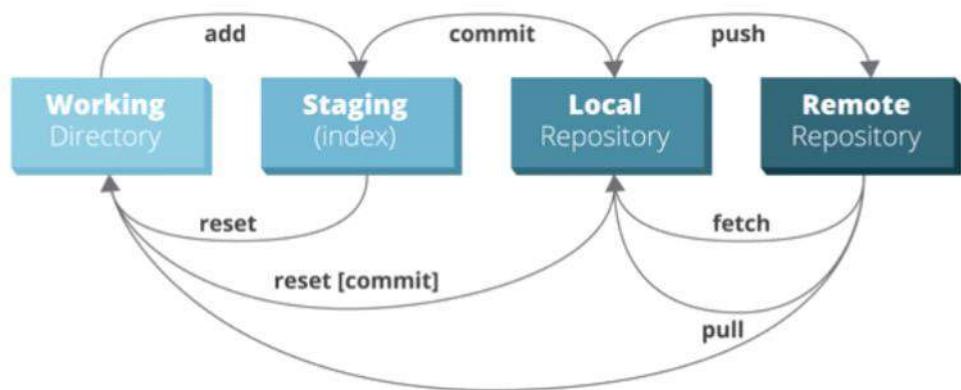
```
$ git push
```

## Finally!

When in doubt, use git help

```
$ git command --help
```

Or visit <https://training.github.com/> for official GitHub training.



## Fork

- Creates a new codebase and update to the fork repop
- Not sync with original remote repo

# WebHook

Wednesday, October 13, 2021 7:09 PM

basic steps of implementing [CI/CD](#) is integrating your SCM (Source Control Management) tool with your CI tool. This saves you time and keeps your project updated all the time.

- Schedule your build
- Pull your code and data files from your GitHub repository to your [Jenkins](#) machine
- Automatically trigger each build on the Jenkins server, after each Commit on your Git repository

**Step 1:** go to your GitHub repository and click on '**Settings**'.

**Step 2:** Click on **Webhooks** and then click on '**Add webhook**'.

**Step 3:** In the '**Payload URL**' field, paste your Jenkins environment URL. At the end of this URL add `/github-webhook/`. In the '**Content type**' select: '`application/json`' and leave the '**Secret**' field empty.

**Step 4:** In the page '**Which events would you like to trigger this webhook?**' choose '*Let me select individual events.*' Then, check '**Pull Requests**' and '**Pushes**'. At the end of this option, make sure that the '**Active**' option is checked and click on '**Add webhook**'.

**Step 5:** In Jenkins, click on '**New Item**' to create a new project.

**Step 6:** Give your project a name, then choose '**Freestyle project**' and finally, click on '**OK**'.

**Step 7:** Click on the '**Source Code Management**' tab.

**Step 8:** Click on **Git** and paste your GitHub repository URL in the '**Repository URL**' field.

**Step 9:** Click on the '**Build Triggers**' tab and then on the '**GitHub hook trigger for GITScm polling**'. Or, choose the trigger of your choice.

## Triggering the Jenkins Job to Run with Every Code Commit

**Step 10:** Click on the '**Build**' tab, then click on '**Add build step**' and choose '**Execute shell**'.

# Authentication less

Monday, September 13, 2021 6:52 PM

## .git config

All the information about our repositories versions is stored in .git directory. It's a hidden folder. There is a config file in it that allows us to configure the settings. But, it's not recommended in general.

We can clone a private repository by adding our username and password in the repository URL as follows.

```
git clone https://<strong>username:password</strong>@github.com/<strong>
```

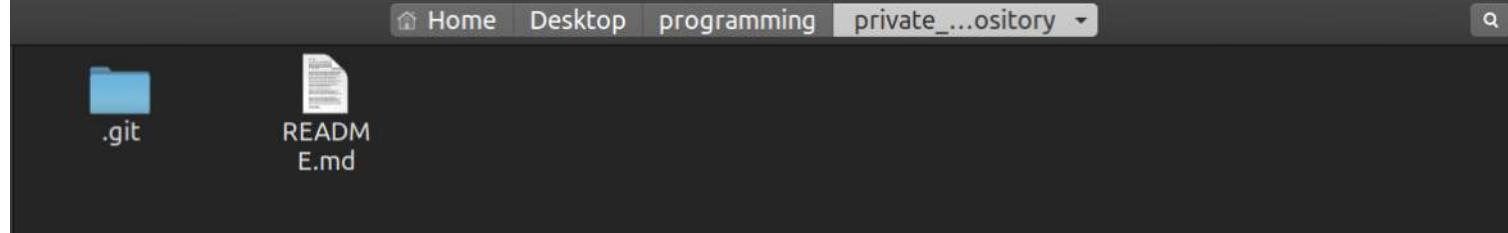
```
<strong>/<strong>/repository_name</strong>.git
```

Copy

Update the username, password, and repository\_name with appropriate details. Since we have given our credentials in the URL, it won't ask for authentication as we have seen before.

So, we are going to follow the above method of authentication and update our repository configuration accordingly. Let's see the steps to get rid of repetitive authentication by updating the URL.

- Open the .git folder in your cloned repository.



.git Folder

- You'll find a file with the name config. Open it using any text editor of your choice.
- There will be a line with our repository link as follows.

```
[remote "origin"]
```

```
url = https://github.com/hafeezulkareem/private_repository.git
```

Repository Link in config

- Update the URL by adding your username and password, as seen above.

```
[remote "origin"]
```

```
url = https://username:password@github.com/hafeezulkareem/private_repository.git
```

Repository URL Update

Now, again update something in the repository, commit and push them.

Do you observe anything?

It shouldn't have asked for your GitHub credentials this time. So, we have solved our problem by updating our repository setting.

You might have noticed that it's not secure. As we are exposing our credentials. And this method won't work in case your GitHub password contains @ character.

So, there are some critical disadvantages of using this method. Hence let's ignore it and move to the next method.

From <<https://geekflare.com/github-setup-passwordless-auth/>>

# Basic Branching and Merging

Thursday, September 23, 2021 1:34 PM

## 3.2 Git Branching -

### Basic Branching and Merging

Let's go through a simple example of branching and merging with a workflow that you might use in the real world. You'll follow these steps:

1. Do some work on a website.
2. Create a branch for a new user story you're working on.
3. Do some work in that branch.

At this stage, you'll receive a call that another issue is critical and you need a hotfix. You'll do the following:

1. Switch to your production branch.
2. Create a branch to add the hotfix.
3. After it's tested, merge the hotfix branch, and push to production.
4. Switch back to your original user story and continue working.

### Basic Branching

First, let's say you're working on your project and have a couple of commits already on the master branch.

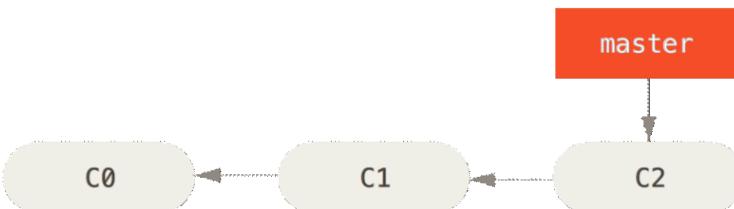


Figure 18. A simple commit history

You've decided that you're going to work on issue #53 in whatever issue-tracking system your company uses. To create a new branch and switch to it at the same time, you can run the `git checkout` command with the `-b` switch:

```
$ git checkout -b iss53
Switched to a new branch "iss53"
```

This is shorthand for:

```
$ git branch iss53
$ git checkout iss53
```

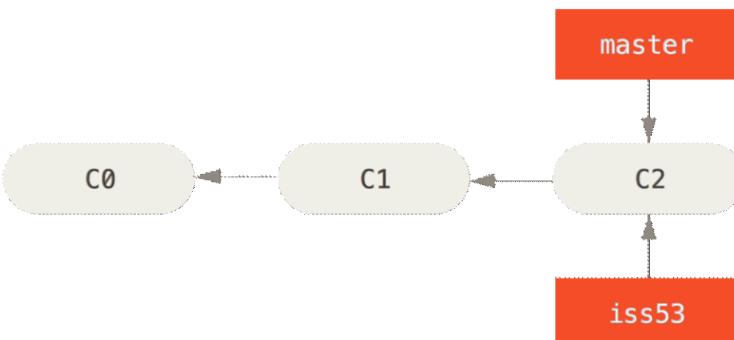


Figure 19. Creating a new branch pointer

You work on your website and do some commits. Doing so moves the `iss53` branch forward, because you have it checked out (that is, your `HEAD` is pointing to it):

```
$ vim index.html
$ git commit -a -m 'Create new footer [issue 53]'
```

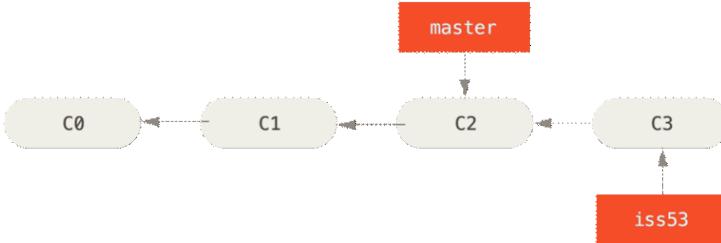


Figure 20. The iss53 branch has moved forward with your work

Now you get the call that there is an issue with the website, and you need to fix it immediately. With Git, you don't have to deploy your fix along with the iss53 changes you've made, and you don't have to put a lot of effort into reverting those changes before you can work on applying your fix to what is in production. All you have to do is switch back to your master branch.

However, before you do that, note that if your working directory or staging area has uncommitted changes that conflict with the branch you're checking out, Git won't let you switch branches. It's best to have a clean working state when you switch branches. There are ways to get around this (namely, stashing and commit amending) that we'll cover later on, in [Stashing and Cleaning](#). For now, let's assume you've committed all your changes, so you can switch back to your master branch:

```
$ git checkout master
Switched to branch 'master'
```

At this point, your project working directory is exactly the way it was before you started working on issue #53, and you can concentrate on your hotfix. This is an important point to remember: when you switch branches, Git resets your working directory to look like it did the last time you committed on that branch. It adds, removes, and modifies files automatically to make sure your working copy is what the branch looked like on your last commit to it.

Next, you have a hotfix to make. Let's create a hotfix branch on which to work until it's completed:

```
$ git checkout -b hotfix
Switched to a new branch 'hotfix'
$ vim index.html
$ git commit -a -m 'Fix broken email address'
[hotfix 1fb7853] Fix broken email address
1 file changed, 2 insertions(+)
```

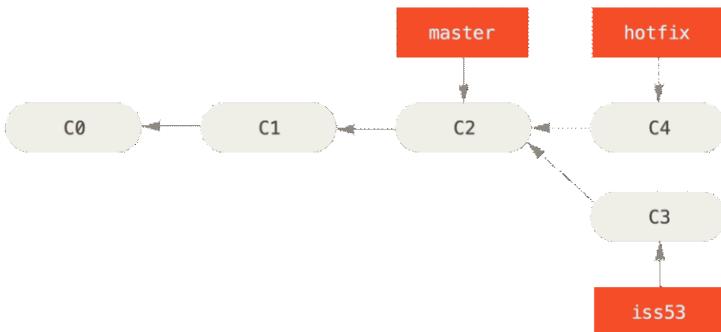


Figure 21. Hotfix branch based on master

You can run your tests, make sure the hotfix is what you want, and finally merge the hotfix branch back into your master branch to deploy to production. You do this with the git merge command:

```
$ git checkout master
$ git merge hotfix
Updating f42c576..3a0874c
Fast-forward
index.html | 2 ++
1 file changed, 2 insertions(+)
```

You'll notice the phrase "fast-forward" in that merge. Because the commit C4 pointed to by the branch hotfix you merged in was directly ahead of the commit C2 you're on, Git simply moves the pointer forward. To phrase that another way, when you try to merge one commit with a commit that can be reached by following the first commit's history, Git simplifies things by moving the pointer forward because there is no divergent work to merge together — this is called a "fast-forward."

Your change is now in the snapshot of the commit pointed to by the master branch, and you can deploy the fix.

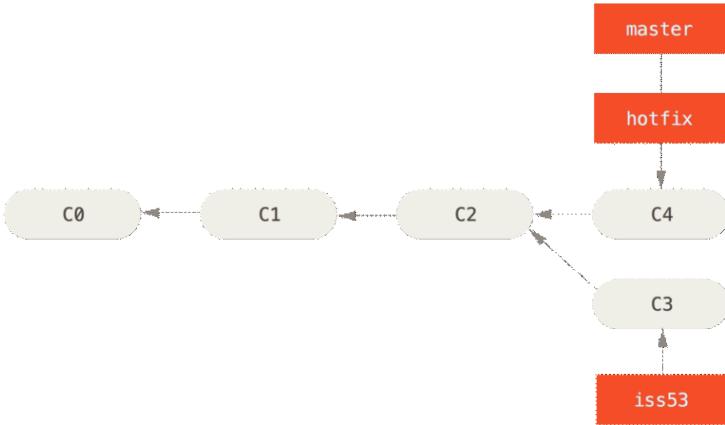


Figure 22. master is fast-forwarded to hotfix

After your super-important fix is deployed, you're ready to switch back to the work you were doing before you were interrupted. However, first you'll delete the hotfix branch, because you no longer need it—the master branch points at the same place. You can delete it with the `-d` option to `git branch`:

```
$ git branch -d hotfix
Deleted branch hotfix (3a0874c).
```

Now you can switch back to your work-in-progress branch on issue #53 and continue working on it.

```
$ git checkout iss53
Switched to branch "iss53"
$ vim index.html
$ git commit -a -m 'Finish the new footer [issue 53]'
[iss53 ad82d7a] Finish the new footer [issue 53]
1 file changed, 1 insertion(+)
```

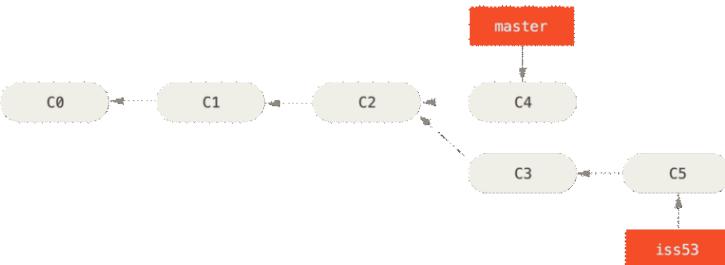


Figure 23. Work continues on iss53

It's worth noting here that the work you did in your hotfix branch is not contained in the files in your iss53 branch. If you need to pull it in, you can merge your master branch into your iss53 branch by running `git merge master`, or you can wait to integrate those changes until you decide to pull the iss53 branch back into master later.

## Basic Merging

Suppose you've decided that your issue #53 work is complete and ready to be merged into your master branch. In order to do that, you'll merge your iss53 branch into master, much like you merged your hotfix branch earlier. All you have to do is check out the branch you wish to merge into and then run the `git merge` command:

```
$ git checkout master
Switched to branch 'master'
$ git merge iss53
Merge made by the 'recursive' strategy.
index.html | 1 +
1 file changed, 1 insertion(+)
```

This looks a bit different than the hotfix merge you did earlier. In this case, your development history has diverged from some older point. Because the commit on the branch you're on isn't a direct ancestor of the branch you're merging in, Git has to do some work. In this case, Git does a simple three-way merge, using the two snapshots pointed to by the branch tips and the common ancestor of the two.

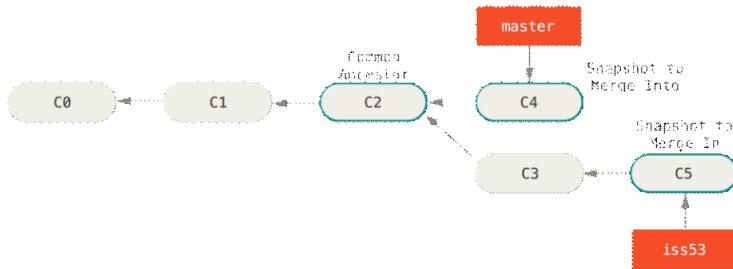


Figure 24. Three snapshots used in a typical merge

Instead of just moving the branch pointer forward, Git creates a new snapshot that results from this three-way merge and automatically creates a new commit that points to it. This is referred to as a merge commit, and is special in that it has more than one parent.

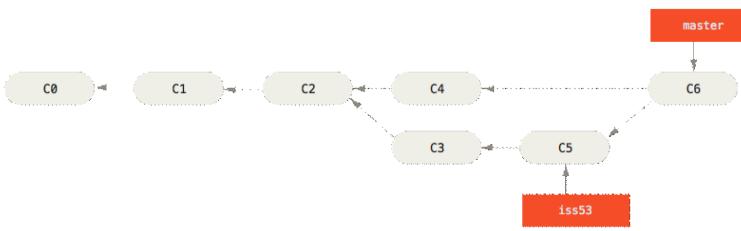


Figure 25. A merge commit

Now that your work is merged in, you have no further need for the iss53 branch. You can close the issue in your issue-tracking system, and delete the branch:

```
$ git branch -d iss53
```

### Basic Merge Conflicts

Occasionally, this process doesn't go smoothly. If you changed the same part of the same file differently in the two branches you're merging, Git won't be able to merge them cleanly. If your fix for issue #53 modified the same part of a file as the hotfix branch, you'll get a merge conflict that looks something like this:

```
$ git merge iss53
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
```

Git hasn't automatically created a new merge commit. It has paused the process while you resolve the conflict. If you want to see which files are unmerged at any point after a merge conflict, you can run `git status`:

```
$ git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")

Unmerged paths:
  (use "git add <file>..." to mark resolution)

both modified:    index.html

no changes added to commit (use "git add" and/or "git commit -a")
```

Anything that has merge conflicts and hasn't been resolved is listed as unmerged. Git adds standard conflict-resolution markers to the files that have conflicts, so you can open them manually and resolve those conflicts. Your file contains a section that looks something like this:

```
<<<<< HEAD:index.html
<div id="footer">contact : email.support@github.com</div>
=====
<div id="footer">
please contact us at support@github.com
</div>
>>>>> iss53:index.html
```

This means the version in HEAD (your master branch, because that was what you had checked out when you ran your merge command) is the top part of that block (everything above the =====), while the version in your iss53 branch looks like everything in the bottom part. In order to resolve the conflict, you have to either choose one side or the other or merge the contents yourself. For instance, you might resolve this conflict by replacing the entire block with this:

```
<div id="footer">  
please contact us at email.support@github.com  
</div>
```

This resolution has a little of each section, and the <<<<<, =====, and >>>>> lines have been completely removed. After you've resolved each of these sections in each conflicted file, run git add on each file to mark it as resolved. Staging the file marks it as resolved in Git.

If you want to use a graphical tool to resolve these issues, you can run git mergetool, which fires up an appropriate visual merge tool and walks you through the conflicts:

```
$ git mergetool
```

This message is displayed because 'merge.tool' is not configured.

See 'git mergetool --tool-help' or 'git help config' for more details.

'git mergetool' will now attempt to use one of the following tools:

```
opendiff kdiff3 tkdiff xxdiff meld tortoisemerge gvimdiff diffuse diffmerge ecmerge p4merge araxis bc3 codecompare vimdiff emerge
```

Merging:

```
index.html
```

Normal merge conflict for 'index.html':

```
{local}: modified file
```

```
{remote}: modified file
```

Hit return to start merge resolution tool (opendiff):

If you want to use a merge tool other than the default (Git chose opendiff in this case because the command was run on a Mac), you can see all the supported tools listed at the top after "one of the following tools." Just type the name of the tool you'd rather use.

Note	If you need more advanced tools for resolving tricky merge conflicts, we cover more on merging in <a href="#">Advanced Merging</a> .
------	--

After you exit the merge tool, Git asks you if the merge was successful. If you tell the script that it was, it stages the file to mark it as resolved for you. You can run git status again to verify that all conflicts have been resolved:

```
$ git status
```

On branch master

All conflicts fixed but you are still merging.

(use "git commit" to conclude merge)

Changes to be committed:

```
modified: index.html
```

If you're happy with that, and you verify that everything that had conflicts has been staged, you can type git commit to finalize the merge commit. The commit message by default looks something like this:

```
Merge branch 'iss53'
```

```
Conflicts:  
index.html
```

```
#  
# It looks like you may be committing a merge.  
# If this is not correct, please remove the file  
# .git/MERGE_HEAD  
# and try again.
```

```
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.
```

```
# On branch master
```

```
# All conflicts fixed but you are still merging.
```

```
#
```

```
# Changes to be committed:
```

```
#     modified: index.html
```

```
#
```

If you think it would be helpful to others looking at this merge in the future, you can modify this commit message with details about how you resolved the merge and explain why you did the changes you made if these are not obvious.

From <<https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging>>

# Merge Conflict

Tuesday, 27 December 2022 11:48 PM

when Git is unable to automatically resolve differences in code between two commits.

To prevent such conflicts, developers work in separate isolated branches.

The Git merge command combines separate branches and resolves any conflicting edits.

## Git Commands to Resolve Conflicts

### 1. git log --merge

The git log --merge command helps to produce the list of commits that are causing the conflict

### 2. git diff

The git diff command helps to identify the differences between the states repositories or files

### 3. git checkout

The git checkout command is used to undo the changes made to the file, or for changing branches

### 4. git reset --mixed

The git reset --mixed command is used to undo changes to the working directory and staging area

### 5. git merge --abort

The git merge --abort command helps in exiting the merge process and returning back to the state before the merging began

### 6. git reset

The git reset command is used at the time of merge conflict to reset the conflicted files to their original state

# Git Stash

Tuesday, 14 February 2023 9:51 PM

- Developer working with current branch wants to switch to another branch.
- Doesn't want to commit changes on unfinished work current branch.
- Solution is Git stash
- **Takes your modified tracked files**
- Saves them on stack of unfinished changes
- You can re-apply them any time

# Fetch & Pull

Tuesday, 14 February 2023 9:58 PM

## Fetch

- Only downloads new data from remote repo
- Doesn't integrate any new data into your working files
- Can run anytime to update the remote-tracking branches
- `# git fetch origin`
- `# git fetch --all`

## Pull

- Pull updates the current HEAD branch with latest changes from remote repo
- Integrate any new data into your working files
- Tries to merge remote changes with your local ones
- `# git pull origin master`

# Terraform & AWS CLI Installation

Thursday, September 23, 2021 1:54 PM

## Step-01: Introduction

- Install Terraform CLI
- Install AWS CLI
- Install VS Code Editor
- Install HashiCorp Terraform plugin for VS Code

### Terraform?

- Terraform is open-source communication as a system software tool created by HashiCorp.
- that is used for building the infrastructure as code.
- It can direct existing and accepted service providers as well as follow convention in-house solutions.

ID: AKIA3QBI2PBU37ZKYI6Z

KEY: gVv0R4QAd+tKz2Cetb0kBBKd6HmOKT4L9FL9KaP/

```
[root@centos7 home]# aws configure
AWS Access Key ID [None]: AKIA4R37AJJQGSJZDDY3
AWS Secret Access Key [None]: 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
Default region name [None]: ap-southeast-1
Default output format [None]: json
```

```
[root@centos7 .aws]# pwd
/root/.aws
```

```
[root@centos7 ~]# cat .aws/credentials
[default]
aws_access_key_id = AKIA4R37AJJQGSJZDDY3
aws_secret_access_key = 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
```

### ## Step-02: MACOS: Terraform Install

- [Download Terraform MAC](<https://www.terraform.io/downloads.html>)
- [Install CLI](<https://learn.hashicorp.com/tutorials/terraform/install-cli>)
- Unzip the package

```
## Step-02: MACOS: Terraform Install
# Download Terraform MAC
curl -O https://releases.hashicorp.com/terraform/0.14.3/terraform_0.14.3_darwin_amd64.zip
# Extract Terraform binary
unzip terraform_0.14.3_darwin_amd64.zip
# Copy terraform binary to /usr/local/bin
mv terraform /usr/local/bin
# Verify Version
terraform version
```



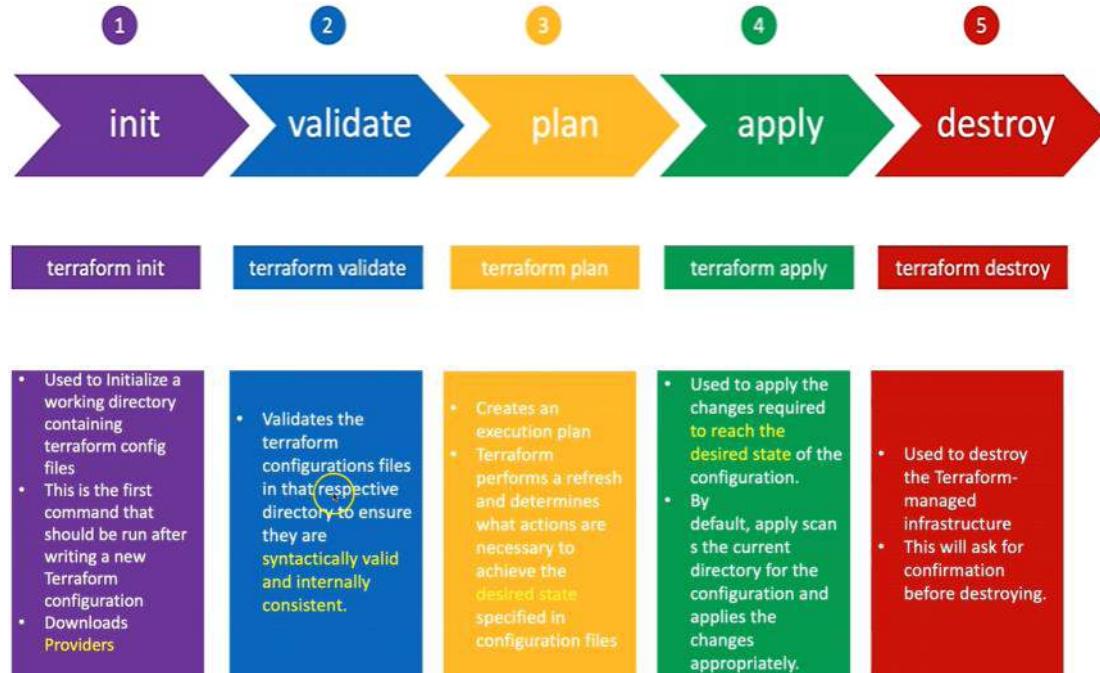
# Provisioning & Command

Tuesday, September 7, 2021 1:41 AM

## Provisioning

- Provisioners are used for executing scripts or shell commands on a local or remote machine as part of resource creation/deletion.
- They are similar to “EC2 instance user data” scripts that only run once on the creation and if it fails terraform marks it tainted.

## Terraform Workflow



# terraform apply -auto-approve

- **Terraform fmt** – it is used to rewrite configuration files in a canonical styles and format
- **Terraform providers** – it gives information of providers working in the current configuration.

List of resources built

# terraform state list

Details output on resource

# terraform state show resource\_name

Main commands:

```
init      Prepare your working directory for other commands
validate  Check whether the configuration is valid
plan     Show changes required by the current configuration
apply    Create or update infrastructure
destroy   Destroy previously-created infrastructure
```

All other commands:

```
console   Try Terraform expressions at an interactive command prompt
fmt       Reformat your configuration in the standard style
force-unlock Release a stuck lock on the current workspace
get      Install or upgrade remote Terraform modules
graph    Generate a Graphviz graph of the steps in an operation
```

```
import  Associate existing infrastructure with a Terraform resource
login   Obtain and save credentials for a remote host
logout  Remove locally-stored credentials for a remote host
output   Show output values from your root module
providers Show the providers required for this configuration
refresh  Update the state to match remote systems
show    Show the current state or a saved plan
state   Advanced state management
taint   Mark a resource instance as not fully functional
test    Experimental support for module integration testing
untaint Remove the 'tainted' state from a resource instance
version Show the current Terraform version
workspace Workspace management
```

# Files

Saturday, 28 May 2022 1:03 PM

.terraform - terraform init - download cloud provider packages

Terraform.tfstate

- state of terraform
- Create any resource with any cloud provider
- Keep track of everything we created

# Provider

Tuesday, September 7, 2021 1:40 AM

## Terraform provider?

- Terraform relies on plugins called "providers" to interact with cloud providers, SaaS providers, and other APIs.
- Terraform configurations declare which providers they require so that Terraform can install and use them.
- Each provider adds a set of resource types and/or data sources that Terraform can manage.
- Every resource type is implemented by a provider; without providers, Terraform can't manage any kind of infrastructure.
- Terraform supports a large number of cloud providers.
- Most providers configure a specific infrastructure platform (either cloud or self-hosted). Providers can also offer local utilities for tasks like generating random numbers for unique resource names.

## Arguments & Attributes

Tuesday, September 7, 2021 1:42 AM

### Arguments & Attributes:

**Arguments** References are input for our resource

**Attribute** reference other outputs, which we are going to take from that resource and then hold and then use those in another resources are another child.

Modules are somewhere like in another separate project configuration files and then how you are going to call them.

## State File

Tuesday, September 7, 2021 1:43 AM

### State File

- Terraform stores information about infrastructure in a state file. This state file keeps track of resources created by your configuration and maps them to real-world resources.
- It's underlying database for that terraformed action, whatever is performing.
- So whenever you execute terraform apply and your create infra is created and once the execution of terraform apply is completed immediately it will create Terraform.tfstate file in your local working directory.
- What is present on cloud, your real infrastructure information is maintained in that data form that state file is.
- Important file for your Infra, so it can be stored on remotely, which works better in team environment.

**Desired state:** terraform configuration files, whatever I have defined in terraform configuration file.

**Current State:** Real world results, whatever the EC2 instance you have created in your cloud env.

## Resource Behavior



## Terraform State

# State File Locking

Tuesday, September 7, 2021 1:44 AM

## State File Locking?

**Answer:** State file locking is a mechanism in terraform where operation on a specific state file is blocked to avoid conflicts between multiple users performing the same operation. Once the lock from one user is released, then only any other user can operate on that state file after taking a lock on it. This helps in preventing any corruption of the state file. It is a backend operation, so the acquiring of lock on a state file in backend. If it takes more time than expected to acquire a lock on the state file, you will get a status message as an output.

# Statefile storage

Wednesday, May 11, 2022 2:37 PM

**Terraform will lock your state for all operations that could write state.** This prevents others from acquiring the lock and potentially corrupting your state. State locking happens automatically on all operations that could write state.

You can run `apply` with the `-lock-timeout=<TIME>` parameter to tell Terraform to wait up to `TIME` for a lock to be released (e.g., `-lock-timeout=10m` will wait for 10 minutes).

Usage: `terraform force-unlock`

**Shared storage for state files:** To be able to use Terraform to update your infrastructure, each of your team members needs access to the same Terraform state files. That means you need to store those files in a shared location.

## IAC & Core

Tuesday, September 7, 2021 1:43 AM

**IaC** is a short form to the term “Infrastructure as Code”. IaC refers to a scheme whereby developers can run and provision the computer data center’s mechanically instead of getting into a physical process. Terraform, for example, is a case tool of IaC.

### Describe the working of **Terraform core**?

**Answer :** The terraform core looks at the configuration monitoring and creates analysis and evaluation based on the configuration. It keeps track and compare the versions (current and previous) and then display the output through the terminal.

Terraform core mainly takes two input:

**Terraform Configuration** – It keeps track of the infrastructure detail

**Terraform state** – It keeps track of the infrastructure status.

# Module

Thursday, September 23, 2021 12:19 PM

A module is a **container for multiple resources that are used together**

we use module clause for modules instead of resource clause.

In modules, we only specify a name, which is used elsewhere in the configuration to reference module outputs.

1. Input variables should be defined in variables.tf.
2. Resources and Data sources should be defined in main.tf.
3. Output values should be defined in outputs.tf

## Modules in Terraform?

**Answer:** A module in Terraform is a **jug for numerous resources that are used jointly**. The root module is required for every Terraform that includes resources mentioned in the .tf files.

# Policy / Modules / Recover / Plugins

Tuesday, September 7, 2021 1:45 AM

## Policies

**Answer:** You cannot insert policies to the open-source description of Terraform Enterprise. The equal also goes for the Enterprise Pro version. The finest version of Terraform Enterprise only could contact the lookout policies.

## Recover from a failed apply in Terraform?

**Answer:** You can put your configuration in version control and commit before each change, and then you can use your version control system's features to revert to an older configuration if needed. You always need to make sure that you recommit the previous version code for it to be the new version in the version control system.

## Plugins?

The authority “Terraform init” helps Terraform interpret configuration files in the operational directory. Then, Terraform finds out the essential plugins and searches for installed plugins in diverse locations. In addition, Terraform also downloads extra plugins at times. Then, it decides the plugin versions to use and writes a security device file for ensuring that Terraform will employ the identical plugin versions.

## Remote / Tainted

Tuesday, September 7, 2021 1:46 AM

### Remote Backend in Terraform?

**Answer:** The remote backend in terraform is used to store the state of terraform and can also run operations in terraform cloud. Remote backend multiple terraform commands such as init, plan, apply, destroy (terraform version >= v0.11.12), get, output, providers, state (sub-commands: list, mv, pull, push, rm, show) , taint, untaint, validate and many more. It can work with a single remote terraform cloud workspace or even multiple workspaces. For running remote operations like terraform plan or terraform apply, you can use terraform cloud's run environment.

### Tainted Resource?

**Answer:** Tainted resources are those resources that are forced to be destroyed and recreated on the next apply command. When you mark a resource as tainted, nothing changes on infrastructure but the state file is updated with this information(destroy and create). After marking a resource as tainted, terraform plan out will show that the resource will get destroyed and recreated, and when the next apply happens the changes will get implemented.

# Terraform cmd

Monday, 10 April 2023 7:07 PM

## Terraform CLI tricks

```
# terraform -install-autocomplete #Setup tab autocomplete, requires logging back in
```

## Format and Validate Terraform code

```
# terraform fmt #format code per HCL canonical standard  
# terraform validate #validate code for syntax  
# terraform validate -backend=false #validate code skip backend validation
```

## Initialize your Terraform working directory

```
❑ terraform init #initialize directory, pull down providers  
❑ terraform init -get-plugins=false #initialize directory, do not download plugins  
❑ terraform init -verify-plugins=false #initialize directory, do not verify plugins for Hashicorp signature
```

## Plan, Deploy and Cleanup Infrastructure

1. terraform apply --auto-approve #apply changes without being prompted to enter "yes"
2. terraform destroy --auto-approve #destroy/cleanup deployment without being prompted for "yes"
3. terraform plan -out plan.out #output the deployment plan to plan.out
4. terraform apply plan.out #use the plan.out plan file to deploy infrastructure
5. terraform plan -destroy #outputs a destroy plan
6. terraform apply -target=aws\_instance.my\_ec2 #only apply/deploy changes to the targeted resource
7. terraform apply -var my\_region\_variable=us-east-1 #pass a variable via command-line while applying a configuration
8. terraform apply -lock=true #lock the state file so it can't be modified by any other Terraform apply or modification action(possible only where backend allows locking)
9. terraform apply refresh=false # do not reconcile state file with real-world resources(helpful with large complex deployments for saving deployment time)
10. terraform apply --parallelism=5 #number of simultaneous resource operations
11. terraform refresh #reconcile the state in Terraform state file with real-world resources
12. terraform providers #get information about providers used in current configuration

## Terraform Workspaces

1. terraform workspace new mynewworkspace #create a new workspace
2. terraform workspace select default #change to the selected workspace
3. terraform workspace list #list out all workspaces
4. terraform workspace delete mynewworkspace # Delete the workspace

## Terraform State Manipulation

1. terraform state show aws\_instance.my\_ec2 #show details stored in Terraform state for the resource
2. terraform state pull > terraform.tfstate #download and output terraform state to a file
3. terraform state mv aws\_iam\_role.my\_ssm\_role module.custom\_module #move a resource tracked via state to different module
4. terraform state replace-provider hashicorp/awsregistry.custom.com/aws #replace an existing provider with another
5. terraform state list #list out all the resources tracked via the current state file
6. terraform state rm aws\_instance.myinstace #unmanage a resource, delete it from Terraform state file

## Terraform Import And Outputs

1. terraform import aws\_instance.new\_ec2\_instance iabcd1234 #import EC2 instance with id i-abcd1234 into the Terraform resource named "new\_ec2\_instance" of type "aws\_instance"
2. terraform import 'aws\_instance.new\_ec2\_instance[0]' iabcd1234 #same as above, imports a real-world resource int an instance of Terraform resource
3. terraform output #list all outputs as stated in code
4. terraform output instance\_public\_ip # list out a specific declared output
5. terraform output -json #list all outputs in JSON format

## Terraform Miscellaneous commands

1. terraform version #display Terraform binary version, also warns if version is old
2. terraform get -update=true #download and update modules in the "root" module.

## Terraform Console(Test out Terraform interpolations)

1. echo 'join("",["foo","bar"])' | terraform console #echo an expression into terraform console and see its expected result as output
2. echo '1 + 5' | terraform console #Terraform console also has an interactive CLI just enter "terraform console"
3. echo "aws\_instance.my\_ec2.public\_ip" | terraform console #display the Public IP against the "my\_ec2" Terraform resource as seen in the Terraform state file

## Terraform Graph(Dependency Graphing)

1. terraform graph | dot -Tpng > graph.png #produce a PNG diagrams showing relationship and dependencies between Terraform resource in your configuration/code

## Terraform Taint/Untaint(mark/unmark resource for recreation - > delete and then recreate)

1. terraform taint aws\_instance.my\_ec2 #taints resource to be recreated on next apply
2. terraform untaint aws\_instance.my\_ec2 #Remove taint from a resource
3. terraform force-unlock LOCK\_ID #forcefully unlock a locked state file, LOCK\_ID provided when locking the State file beforehand

## Terraform Cloud

1. `terraform login` #obtain and save API token for Terraform cloud
2. `terraform logout` #Log out of Terraform Cloud, defaults to hostname app.terraform.io

## Terraform More

Saturday, August 28, 2021 12:16 AM

### Question 2: What are the reasons for choosing Terraform for DevOps?

**Answer:** Below are the reasons for choosing Terraform for DevOps:

- It can do complete orchestration and not just configuration management (like Ansible and Puppet).
- Has amazing support of almost all the popular cloud providers like AWS, Azure, GCP, DigitalOcean etc.
- Easily manages the configuration of an immutable (dynamic) infrastructure.
- Provide immutable infrastructure where configuration changes smoothly.
- Works on HCL (HashiCorp configuration language), which is very easy to learn and understand.
- Easily portable from one provider to another.
- Easy Installation.

### Question 5: What are the ways to lock Terraform module versions?

**Answer:** You can use the terraform module registry as a source and provide the attribute as 'version' in the module in a terraform configuration file. If you are using the GitHub repository as a source, then you need to specify the branch, version and query string with '? ref'.

### Question 6: What do you mean by Terraform cloud?

**Answer:** Terraform Cloud is an application that helps teams use Terraform together. It manages Terraform runs in a consistent and reliable environment, and includes easy access to shared state and secret data, access controls for approving changes to infrastructure, a private registry for sharing Terraform modules, detailed policy controls for governing the contents of Terraform configurations, and more.

### Question 7: Define null resource in Terraform?

**Answer:** The null resource implements the average resource lifecycle but takes no extra action. The trigger argument permits specifying a subjective set of values that, when misrepresented will source the reserve to be replaced.

The primary use-case for the null resource is as a do-nothing container for arbitrary actions taken by a provisioner.

### Question 8: Can Terraform be used for on-prem infrastructure?

**Answer:** Yes, Terraform can be utilized for on-prem infrastructure. There are a lot of obtainable providers. You can decide any one of them which suits you most excellent. Many also build client Terraform providers for themselves; all wanted is just an API.

### Question 9 : What does the following command do?

**Answer:**

- **Terraform -version** – to check the installed version of terraform
- **Terraform destroys** – to destroy the managed infrastructure of terraform.

### Question 10 : List all the Terraform supported versions

**Answer:**

- GitHub.com
- GitLab.com
- GitHub Enterprise
- GitLab CE and EE
- Bitbucket Cloud and Server
- Azure DevOps Server and Services

## 3. Advanced Level: Terraform Interview Question

**Question 2: What do you mean by Terragrunt, list some of its use cases?**

**Answer:** Terragrunt is a thin wrapper that provides extra tools for keeping your configurations DRY, working with multiple Terraform modules, and managing remote state.

Use cases:

- Keep your Terraform code DRY
- Keep your remote state configuration DRY
- Keep your CLI flags DRY
- Execute Terraform commands on multiple modules at once
- Work with multiple AWS accounts

**Question 3: What steps should be followed for making an object of one module to be available for the other module at a high level?**

**Answer:** Following are the steps that should be followed for making an object of one module to be available for the other module at a high level:

1. First, an output variable to be defined in a resource configuration. Till you do not declare resource configuration details, the scope of local and to a module.
2. Now, you have to declare the output variable of module\_A to be used in other module's configuration. A brand new and latest key name should be created by you and the value should be kept equivalent to the module\_A's output variable.
3. Now, for module\_B you have to create a file variable.tf. Establish an input variable inside this file having exactly the same name as was in the key defined by you in module\_B. In a module, this particular variable enables the resource's dynamic configuration. For making this variable available to some other module also, replicate the process. This is because the particular variable established here have its scope restricted to module\_B.

**Question 7 : How callbacks on Azure are utilised with the help of terraform?**

**Answer :** Azure Event Hub is used to perform Azure callbacks with the terraform. It helps in achieving functionality like sending a callback to the system as well as to other events. Terraform AzureRM already include this functionality to ease the process.

**Question 8 : How to prevent Error Duplicate Resource**

**Answer :** It can be done in three ways depending on the situation and the requirement

- 1) By deleting the resource so that terraform code stops managing them.
- 2) By discarding resource from the APIs
- 3) Importing action will also help to eliminate resource

**Question 4: Name some major competitors of Terraform?**

**Answer:** Some of them are:

- Packer
- Cloud Foundry
- Ansible
- Kubernetes

**Question 7: Name some major features of Terraform?**

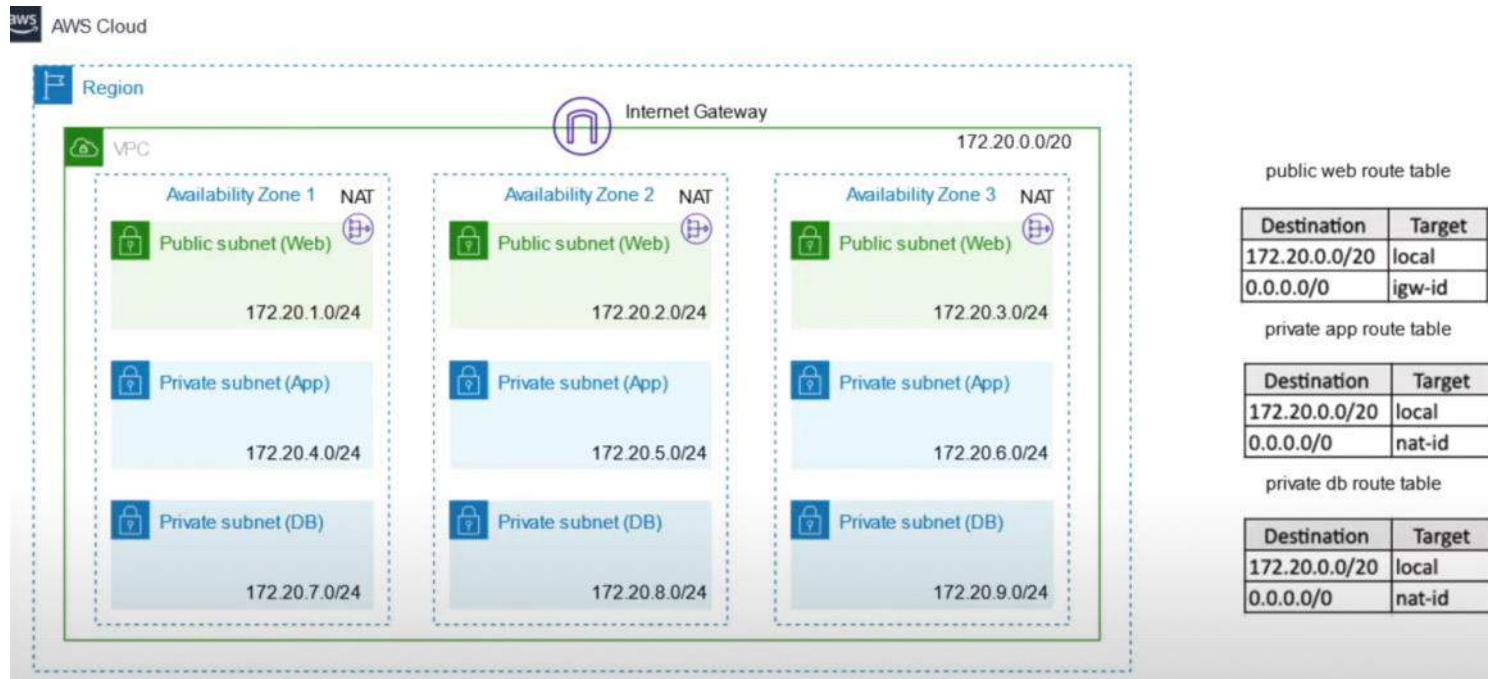
**Answer:** Some of them are:

- Execution Plan
- Change Automation
- Resource Graph
- Infrastructure as code

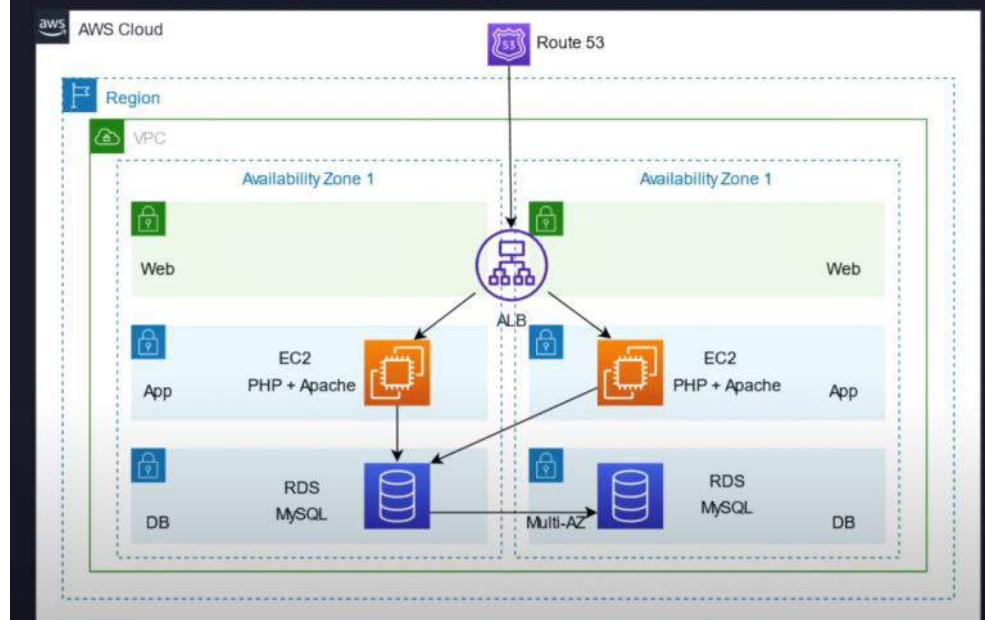


## 3 tier Arch AWS

Thursday, May 12, 2022 9:47 AM



## Three tier app architecture



- Create VPC
- Create corresponding subnets
- Create route tables & do subnet associations
- Create Internet GW (we need to establish the connectivity) & attached to VPC - allows communication between resources in your VPC and the internet.
- Create NAT GW in public subnet- is in each Availability Zone that **enable EC2 instances in private subnets (App and Data) to access the internet.**
- Make changes in route table so that connectivity to the internet will be established from your internet gateway for the public subnet & NAT gw for private subnets
- Create jump server & App server

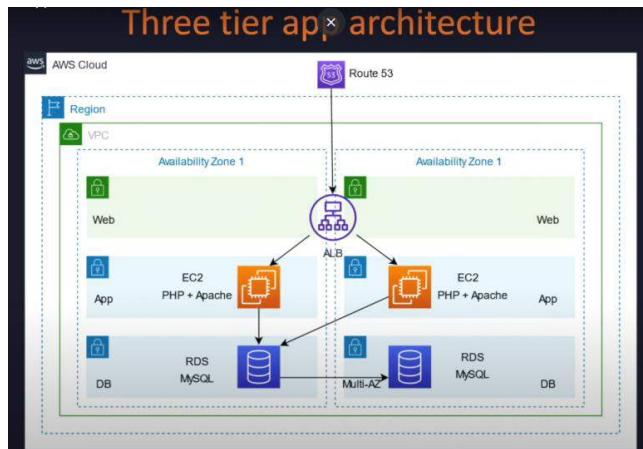
Element	Brief description
Virtual Private Cloud (VPC)	A logically isolated virtual network in the AWS cloud. You define a VPC's IP address space from a range you select.

<i>Subnet</i>	A segment of a VPC's IP address range where you can place groups of isolated resources.
<i>Internet Gateway</i>	The Amazon VPC side of a connection to the public Internet.
<i>NAT Gateway</i>	A highly available, managed Network Address Translation (NAT) service for your resources in a private subnet to access the Internet.
<i>Hardware VPN Connection</i>	A hardware-based VPN connection between your Amazon VPC and your datacenter, home network, or co-location facility.
<i>Virtual Private Gateway</i>	The Amazon VPC side of a VPN connection. The Customer gateway is the customer side of a VPN connection.
<i>Peering Connection</i>	A peering connection enables you to route traffic via private IP addresses between two peered VPCs
<i>VPC Endpoint</i>	Enables Amazon S3 access from within your VPC without using an Internet gateway or NAT, and allows you to control the access using VPC endpoint policies.

## 3 Tier Architecture

Saturday, October 9, 2021 1:02 AM

- Create VPC - make it secure, private env
- Create corresponding subnets (private for App and DB) (Public for Web)
- Create route tables & do subnet associations (helping you traffic being routed through your VPC)
- Create Internet GW (we need to establish the connectivity) & attached to VPC - allows communication between resources in your VPC and the internet.
- Create NAT GW in public subnet: is in each Availability Zone that enable EC2 instances in private subnets (App and Data) to access the internet.
- Make changes in route table so that connectivity to the internet will be established from your internet gateway for the public subnet & NAT gw for private subnets
- Create jump server & App server



**Private Network - VPC** - To make it secure, Private isolate environment

**Web Server (VM) - EC2 / EBS** - using web server hosting on ec2 with attached EBS - front end stuff is taken care by web server

**IP** - User can access from outside

**App Server (VM) - EC2 / EBS** - business logic, suppose it's any social web application, connecting with different people, making new connection etc

**Relational Database - RDS** - You want to extend is then required some kind of Database, like MySQL or oracle whatever you prefer.

Considering your app is doing good, there is more traction from the users and your webserver and app server becomes a bottleneck. Not able to handle the load

**Scaling - Auto Scaling for EC2** -

vertical (Increase the capacity of machine components)

horizontal ( adding more machines). If there's load increasing on these ec2 they can scale horizontally automatically.

Now multiple servers and IPs are there,

**Load Balancer - ELB** - you need Intelligent entity that distribute the load to the servers that's load balancing. Which can distributes the incoming traffic to multiple backend EC2 machines

**DNS - Route53** - Map your DNS domain name to load balancer IP

Data is growing, Connection growing so RDS cannot really serve this kind of data storage.

**NoSQL Database (Mongo DB) / DynamoDB** - you need scalable databases and also for connection information and all it makes sense to rather going to NoSQL databases. Some part stored in RDS

Your RDS could be a performance bottleneck. Maybe there is read heavy operations happening on RDS, for that typically you will bring in one more component

**DB Cache - ElasticCache** - (redis and mem chached engines) where you query frequently accessed data so that your application servers don't hit the DB and all request serve from DB Cache

Getting more data like million of picture or videos daily. EBS on Ec2 are not really capable of extending on the fly and have size limitation, so this media never stored typically on these web servers

**External Storage - S3** - unlimited kind of storage, Not a block storage but file storage. You can go on dumping the data and it is accessible over the internet

**Content Filter - Rekognition** - filter the content that are objectionable from the media files

**Click Stream Analysis - Kinesis** - Continue watching what activity you are doing, like what product or what post you are liking, based on that it gives you suggestions.

**Storage for Click Stream - S3** - what data captured will be stored to external storage

**Spark/Hadoop - EMR** - to do some data operations like aggregation, sort the data and find meaning out of that data.

**Data Processing - Glue** - ETL transactions from your dynamoDB tables like maybe you want to do what all friends are there, friend's friends, What activity doing. So year end you want all this data to be extracted & converted into different format data cataloging then further do some data processing using EMR so you need this glue service.

**Data Warehousing - RedShift** - To do data analytics in the end of the year.

**Business Intelligence tool** - Quicksight or Athena - to query the data and generate the report

**Video Convert - Lambda** - Users accessing media files using Mobile so you need converter. Serverless service, just write that code specify how to convert a video. Execution happening into your S3 so new video comes lambda gets triggered, it will convert your video.

## Mobile - External Storage S3

**Content Delivery Network (Cache) - CloudFront / Edge Location** - Caches these videos and pictures to the nearest caching devices from where the user is accessing data - low latency, better experience  
CloudFront stores or caches your data in edge locations

Notification Service - When users send some request to add connection or message or Email

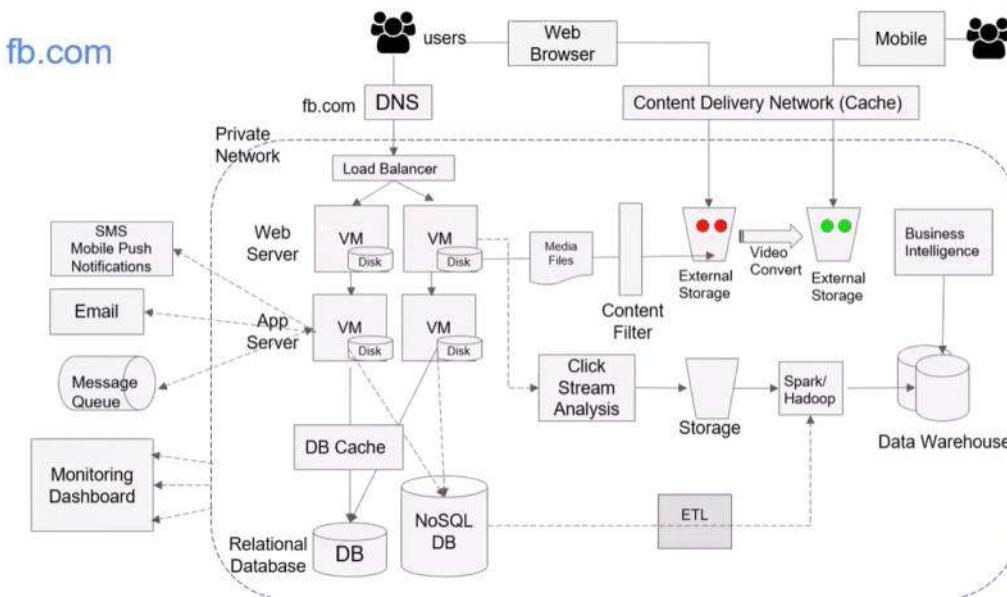
SMS Mobile push notification - SNS (Simple Notification Service)

Email - SES (Simple Email Service)

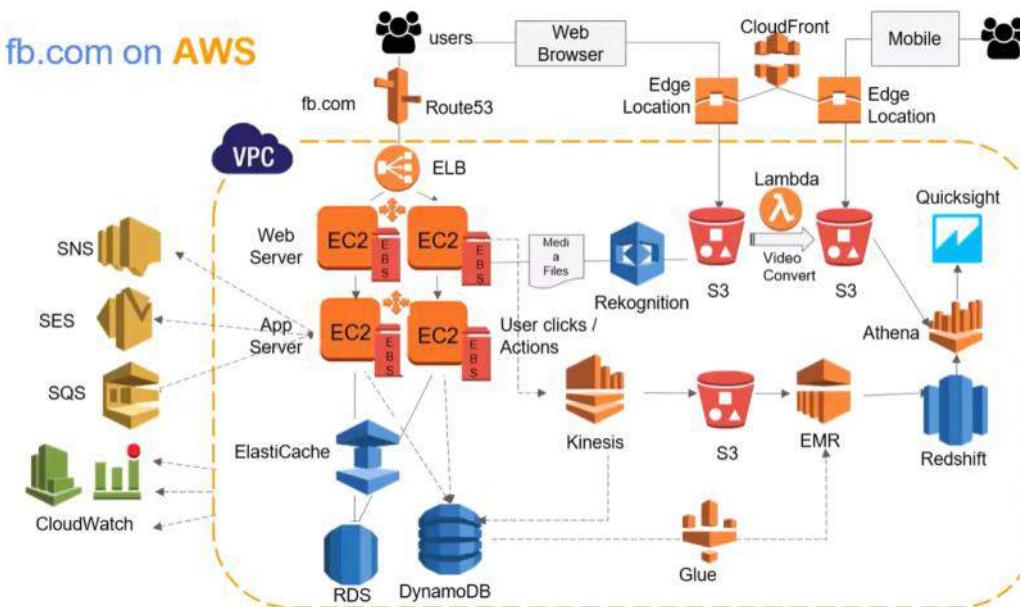
Message Queue - SQS (Simple Queue Service)

Monitoring Dashboard - CloudWatch - Monitoring continuously, Set alarm, Alerts, Take some action, Do some autoscaling

## On-Premises



## On AWS



## AWS Application Services

**REST API - API Gateway** - It exposes all their services through API Calls so that different third-party application can integrate with these applications for that they need REST API services  
In Amazon you can have managed API gateways where it takes care of scaling, throttling, everything so you can just write a code for your APIs,

**Cognito** - Also mobile usage increasing most of your web users, you need to manage their identities like when you develop an application your users must sign up your application. That means you need to manage your user pools, their access and everything.

# AWS Application Services



## Security Service

**IAM - Identity Access Management** - managing all access in your AWS users, What access they have, what service they can use, so all access and authentication and authorization is managed using amazon's IAM.

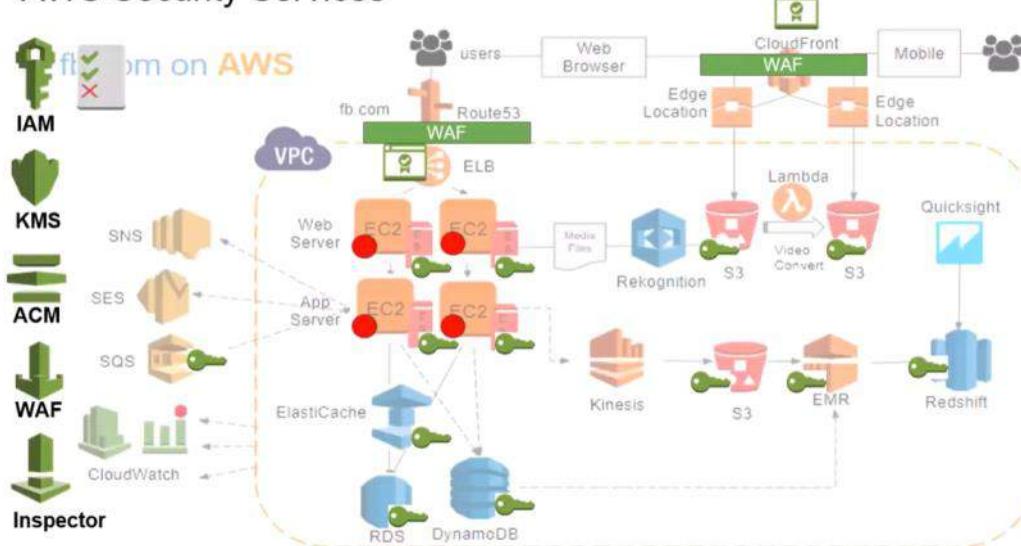
**KMS - Key Management Service** - Encrypt Data which in various storage locations like EBS, S3, EMR, RedShift etc. it manages all the encryption keys for you. You don't need to have your own secure location where you can store your keys and do the encryption.

**ACM - Amazon certificate manager** - Application will be accessed over HTTPS which is SSL enabled connection because if users are doing transaction and they don't want to lose that communication and for this you need digital certificates. That you deploy on Load balancer or CloudFront so that your communication is secure.

**WAF - Web application firewall** - that take care of any attacks. It can prevent like cross-site scripting sql injection, DDoS attacks. WAF can protect your application from that Deploy on CloudFront, Load Balancer or API Gateway.

**AWS Inspector** - Machines need to be patched properly, Free from Vulnerabilities or CVE. Put agent inside your machine and it scan your machine for any Vulnerabilities and gives you report.

## AWS Security Services



## AWS Development and DevOps Services (IaaS)

**CloudFormation** - JSON YAML Template and create your infrastructure from scratch, maybe within 30min. Template will be written by DevOps people and same time you will have Developers and QA.

**CodeCommit** - Everyone required some kind of code repository like GIT so AWS have CodeCommit Service.

**CodeBuild** - Take the source code and build artifacts. Artifacts are like your exec or binaries, actual your application is executable basically.

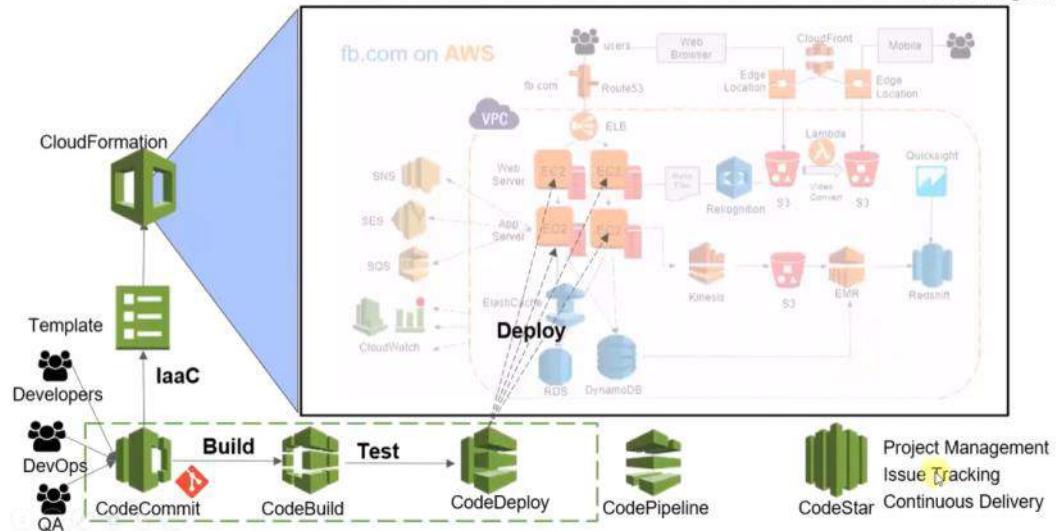
**CodeDeploy** - Whatever you produce you have to deploy it. Put your application on EC2 where your app is running, you required deployment

**CI/CD** - This is your pipeline. Automatically deployment

**CodeStar** - Project Management tool like JIRA, Issue Tracking, Continuous Delivery, integrate all these things with project management tool

# AWS Development and DevOps Services

AWS Region



# Landing Zone

Tuesday, 25 October 2022 5:58 PM

A landing zone is a **well-architected**,

**multi-account AWS environment you can deploy workloads and applications.**

It provides a baseline to get started with multi-account architecture, identity and access management, governance, data security, network design, and logging.

# AWS Console

## access

Wednesday, October 5, 2022 12:50 PM

[Putty - Resources](#)

[AWS CLI in Linux](#)

[AWS CLI for Windows](#)

[AWS CLI for Windows CMD](#)

[AWS SDK - Software development kit](#)

[Eclipse](#)

# AMI vs Instance

Thursday, October 6, 2022 10:14 PM

AMI - Amazon machine image - template of OS

Many different type of Instance can be launched from AMI

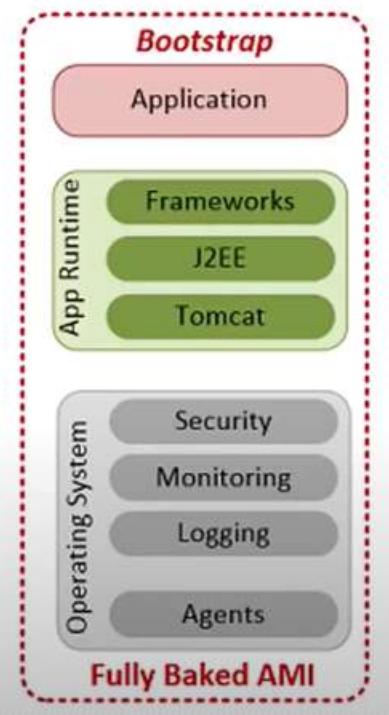
Instance as hardware machine where you will install AMI

# AMI design

Wednesday, October 5, 2022 3:00 PM

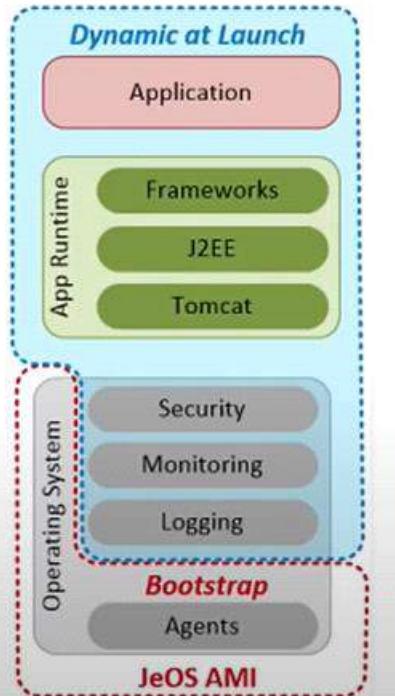
## Fully Baked AMI

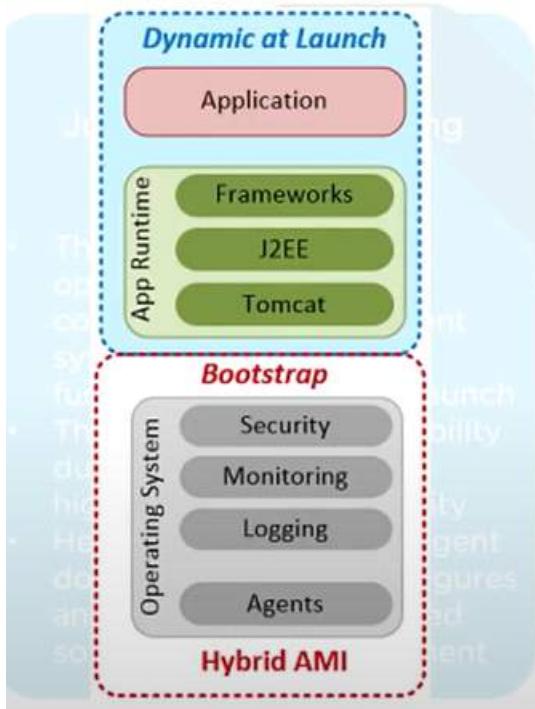
- These AMIs are the simplest to deploy and provide the fastest launch times
- This is best suited for small AWS deployments as it can be expensive and cumbersome to setup



## Just Enough Operating System AMI

- This has a minimal operating system that is fully functional system at its launch
- They offer the most flexibility during deployment and highest levels of portability
- Here, the configuration agent downloads, installs and configures all the required software during deployment



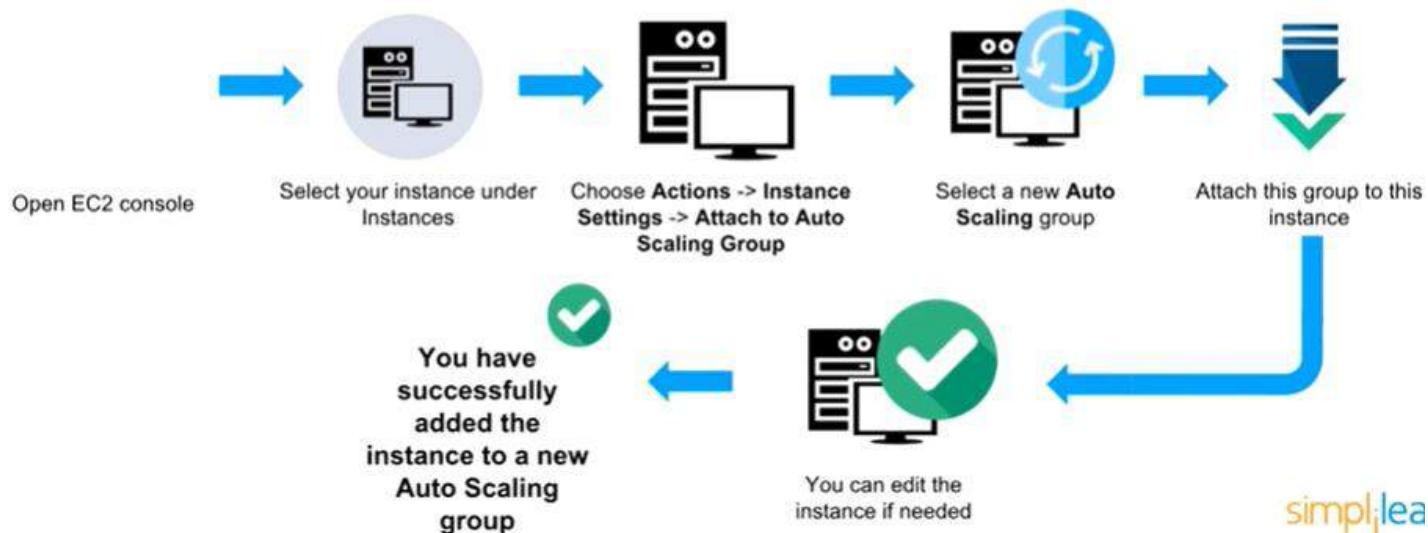


## Hybrid AMI

- Hybrid AMIs fall in between the fully baked and JeOS options
- These AMIs have a partially baked generic infrastructure on top of which you can install required software based on your requirement
- Frameworks, J2EE and Tomcat run during runtime and help to create role specific AMIs

# Add an existing instance to new Auto Scale

Wednesday, October 5, 2022 4:41 PM



# CloudFormation

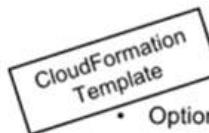
Wednesday, October 5, 2022 12:37 PM

- Create or use existing CloudFormation template using JSON/YAML format
- Save the code template locally or in S3 bucket as repo
- From CloudFormation call the template from S3 bucket and create stack
- AWS CF constructs and configures your stack resources that you have specified in your template.



AWS CloudFormation

AWS CloudFormation templates are JSON or YAML-formatted text files that are comprised of five types of elements:



- Optional list of template parameters
- An optional list of output values
- An optional list of data tables
- List of AWS resources and their configuration values
- A template file format version number

This feature simplifies system administration and layered solutions built on top of AWS CloudFormation

This is useful in cases when you accidentally exceed your limit of Elastic IP addresses or don't have access to an EC2 AMI

By default, the "automatic rollback on error" feature is enabled



This will delete all AWS resources that AWS CloudFormation created till the point where an error occurred

# CloudFormation vs Elastic Beanstalk

Wednesday, October 5, 2022 4:54 PM



AWS CloudFormation



AWS Elastic Beanstalk

- AWS CloudFormation helps you describe and provision all the infrastructure resources in your cloud environment
- It supports the infrastructure needs of many different types of applications such as existing enterprise applications, legacy applications, applications

- AWS Elastic Beanstalk provides an environment to easily deploy and run applications in the cloud
- It is combined with developer tools and helps you manage the lifecycle of your applications

# Cost Explorer

Wednesday, October 5, 2022 12:46 PM

## 1. Top Free Tier Services Table

- This is a dashboard of the Billing and Cost Management console
- This table shows the free tier usage limit for your top five most-used free tier services

Top Free Tier Services by Usage			<a href="#">View all</a>
Service	Free Tier usage limit	Month-to-date usage	
AmazonS3	2,000 Put Requests of Amazon S3	100.00% (2,000.00/2,000 Requests)	
AmazonS3	5 GB of Amazon S3 standard storage	60.00% (3/5 GB-Mo)	

## 2. Cost Explorer

- This allows you to view and analyze costs
- You can view costs for the last 13 months
- You can also get cost forecast for the coming 3 months

## 3. AWS Budgets

- Here, you can plan your service usage, service costs and instance reservations
- You can view the following:
  - Is your current plan meeting your budget?
  - Usage details

## 4. Cost Allocation Tags

- You can assign a label to every AWS resource
- Each tag has a *key* and a *value*
- You can organize your resources and cost allocation tags to keep a track of your AWS costs

# CloudTrail Monitoring

Thursday, October 6, 2022 10:30 PM

AWS CloudTrail **enables**

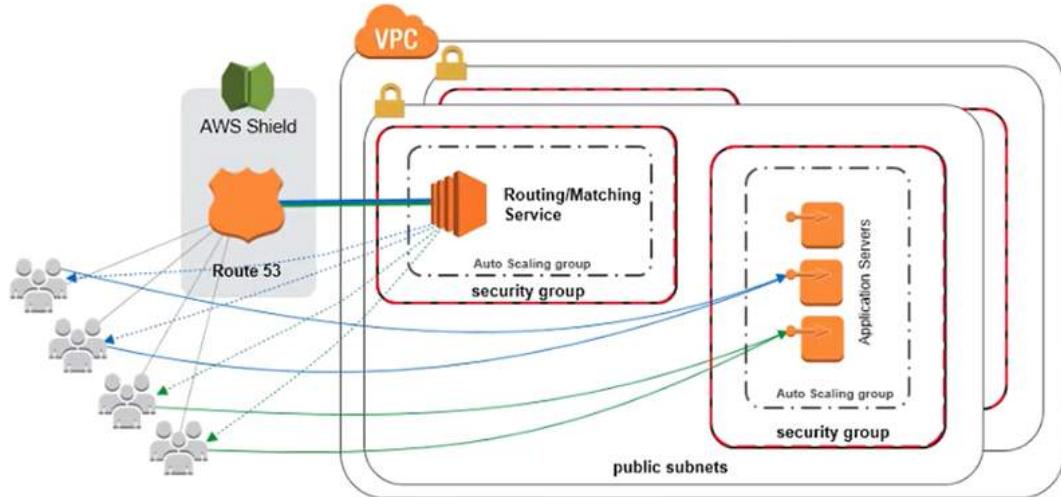
**auditing, security monitoring, and operational troubleshooting by tracking user activity and API usage.**

CloudTrail logs, continuously monitors, and retains account activity related to actions across your AWS infrastructure, giving you control over storage, analysis, and remediation actions.

# DDoS attack

Wednesday, October 5, 2022 1:07 PM

We can minimize DDoS attacks using the below architecture where a TCP or UDP based application



A DDoS attack is an attempt to make a website or an application unavailable to other genuine end users. This is achieved by hackers using various methods that completely consume a network and its resources

We can minimize DDoS attacks using the following services:



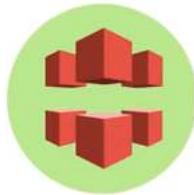
AWS Shield



AWS WAF



Amazon Route53



CloudFront



ELB



VPC

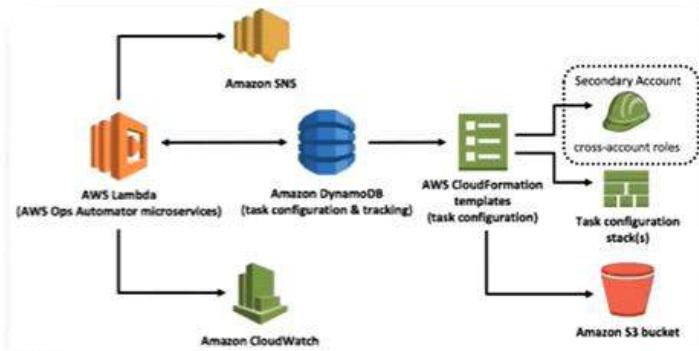
# Delete old snapshot

Wednesday, October 5, 2022 5:07 PM

- As per best practices, you will take snapshots of EBS volumes on Amazon S3
- You can use AWS Ops Automator to automatically handle all snapshots
- It allows you to **create, copy and delete** Amazon EBS snapshots

## How to deploy?

- You can deploy AWS Ops Automator using AWS CloudFormation
- This Automator will use the CloudFormation template to automatically handle all EBS snapshots



© simplilearn. All rights reserved.

simplilearn

## DNS Route 53

Wednesday, October 5, 2022 5:15 PM

21

What is the difference between Latency Based Routing and Geo DNS?

### Geo Based Routing



Geo DNS bases routing decisions on the geographic location of the requests

If you have compliance, localization requirements, or other use cases that require stable routing from a specific geography to a specific endpoint, we recommend using Geo DNS

Latency Based Routing utilizes latency measurements between viewer networks and AWS datacenters. These measurements are used to determine which endpoint to direct users toward

If your goal is to minimize end-user latency, we recommend using Latency Based Routing

22

What is the difference between a Domain and a Hosted Zone?

### Domain

www.simplilearn.com

Domain is a collection of data describing a self-contained administrative and technical unit on the internet

www.simplilearn.com is a domain and is a general DNS concept

### Hosted Zone



Hosted zone is a container that holds information about how you want to route traffic on the internet for a specific domain

All resource record sets within a hosted zone must have the hosted zone's domain name as a suffix.

E.g. lms.simplilearn.com

# Data transfer huge

Wednesday, October 5, 2022 4:53 PM



- AWS Snowball is a data transport solution for moving high volumes of data into and out of a specified AWS region (50TB & 82TB versions)

- AWS Snowball Edge adds additional computing functions apart from providing a data transport solution like AWS Snowball (Up to 100 TB)

exabyte-scale migration service that allows you to transfer data up to 100 PB (100,000 TB)

# EC2 Backup

Wednesday, October 5, 2022 4:58 PM

To automate the EC2 backup, you will need to write a script to automate the below steps by using AWS' API

Below is the step by step process which should be followed in the script:



**Amazon Elastic Block Storage**

1. Get the list of instances
2. Connect to AWS through API to list the Amazon EBS volumes that are attached locally to the instance
3. List the snapshots of each volume
4. Assign a retention period to the snapshot
5. Create a snapshot of each volume
6. Delete the snapshot if it is older than the retention period

# EC2 Recovery with CloudWatch

Wednesday, October 5, 2022 2:57 PM



- You can create an Alarm using Amazon CloudWatch
- In this Alarm, go to Define Alarm -> Actions tab
- Select the “Recover this instance” option

**Create Alarm**

1. Select Metric [2. Define Alarm](#)

Provide the details and threshold for your alarm. Use the graph on the right to help set the appropriate threshold.

Name: PR-01-W-AR  
Description: Auto Recovery on StatusCheckFailed\_System

Whenever: StatusCheckFailed\_System  
is:  $\geq 1$   
for: 2 consecutive period(s)

**Actions**

Define what actions are taken when your alarm changes state.

EC2 Action [Delete](#)

Whenever this alarm: State is ALARM  
 Recover this instance [?](#)  
 Stop this instance [?](#)  
 Terminate this instance [?](#)

This will auto recover your EC2 instance (i-384127cb). You can only recover certain EC2 instance types. Please see documentation.

Namespace: AWS/EC2  
InstanceId: i-384127cb  
InstanceName: Prod-01-Web  
Metric Name: StatusCheckFailed\_Sy

Period: 1 Minute [:](#)  
Statistic: Minimum [:](#)

[Cancel](#) [Back](#) [Next](#) **Create Alarm**

# EC2 to S3 connection

Tuesday, 25 October 2022 12:01 PM

- Create IAM role that grant access to S3
- Attached IAM instance profile to the instance
- Validate permission on your S3 bucket, network connectivity from the EC2 to S3 and check access S3

## Resolution

### Create an IAM instance profile that grants access to Amazon S3

1. Open the [IAM console](#).
2. Choose Roles, and then choose Create role.
3. Select AWS Service, and then choose EC2 under Use Case.

Note: Creating an IAM role from the console with EC2 selected as the trusted entity automatically creates an IAM instance profile with the same name as the role name. However, if the role is created using the AWS Command Line Interface (AWS CLI) or from the API, then an instance profile isn't automatically created. For more information, refer to [I created an IAM role, but the role doesn't appear in the dropdown list when I launch an instance. What do I do?](#)

4. Select Next: Permissions.
5. Create a custom policy that provides the minimum required permissions to access your S3 bucket. For instructions on creating custom policies, see [Writing IAM policies: how to grant access to an Amazon S3 bucket](#) and [Identity and access management in Amazon S3](#).

Note: Creating a policy with the minimum required permissions is a security best practice. However, to allow EC2 access to all your Amazon S3 buckets, use the AmazonS3ReadOnlyAccess or AmazonS3FullAccess managed IAM policy.

6. Select Next: Tags, and then select Next: Review.
7. Enter a Role name, and then select Create role.

### Attach the IAM instance profile to the EC2 instance

1. Open the [Amazon EC2 console](#).
2. Choose Instances.
3. Select the instance that you want to attach the IAM role to.
4. Choose the Actions tab, choose Security, and then choose Modify IAM role.
5. Select the IAM role that you just created, and then choose Save. The IAM role is assigned to your EC2 instance.

### Validate permissions on your S3 bucket

1. Open the [Amazon S3 console](#).
2. Select the S3 bucket that you want to verify the policy for.
3. Choose Permissions.
4. Choose Bucket Policy.
5. Search for statements with Effect: Deny.
6. In your bucket policy, edit or remove any Effect: Deny statements that are denying the IAM instance profile access to your bucket. For instructions on editing policies, see [Editing IAM policies](#).

### Validate network connectivity from the EC2 instance to Amazon S3

For your EC2 instance to connect to S3 endpoints, the instance must be one of the following:

- EC2 instance with a public IP address and a route table entry with the default route pointing to an Internet Gateway
- Private EC2 instance with a default route through a [NAT gateway](#)
- Private EC2 instance with connectivity to Amazon S3 using a [gateway VPC endpoint](#)

To troubleshoot connectivity between a private EC2 instance and an S3 bucket, see [Why can't I connect to an S3 bucket using a gateway VPC endpoint?](#)

## Validate access to S3 buckets

1. [Install the AWS CLI on your EC2 instance.](#)

Note: If you receive errors when running AWS CLI commands, [make sure that you're using the most recent version of the AWS CLI](#).

2. Verify access to your S3 buckets by running the following command. Replace DOC-EXAMPLE-BUCKET with the name of your S3 bucket.

```
aws s3 ls s3://DOC-EXAMPLE-BUCKET
```

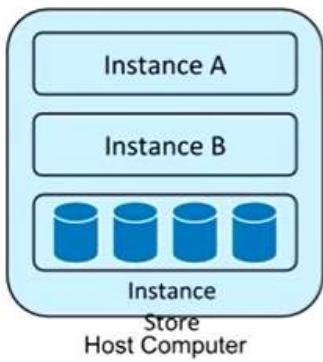
Note: S3 objects encrypted with an AWS Key Management Service (AWS KMS) key must have kms: Decrypt permissions granted in the following:

- The IAM role attached to the instance.
- The KMS key policy.

If these permissions aren't granted, then you can't copy or download the S3 objects. For more information, see [Do I need to specify the AWS KMS key when I download a KMS-encrypted object from Amazon S3?](#)

# EBS vs Instance Store

Wednesday, October 5, 2022 5:04 PM

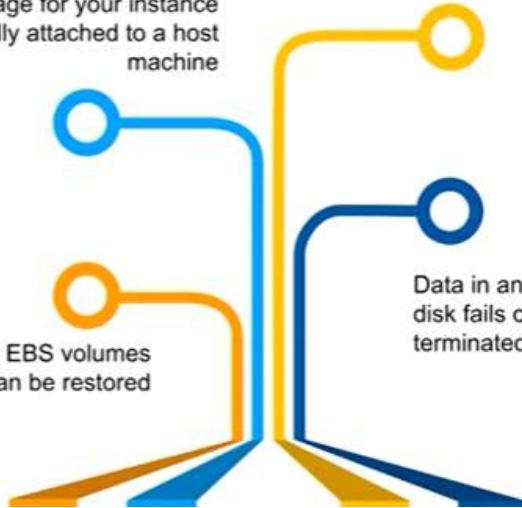


Instance store provides temporary block-level storage for your instance that is physically attached to a host machine

You cannot detach an instance store from one instance and attach it to another

Instances in EBS volumes can be restored

Data in an instance store is lost if the disk fails or the instance is stopped or terminated



# Elasticache

Thursday, October 6, 2022 10:34 PM

Which can be access faster. Repeatable web data can be cache

## Why do we use Elasticache? And in what cases?



- Using this you can deploy, run, scale, in-memory data stores
- Improve the performance of your app
- When frequent reads are required for similar data, we can use ElastiCache
- Preferable for Gaming, Ad-Tech, Financial Services, Healthcare and IoT apps

## FSX & EFS

Thursday, October 6, 2022 9:36 PM

FSX is shared drive service where you get high IO (Input/Output).

If Application you are not using that configure with higher IO then can use EFS

# IAM

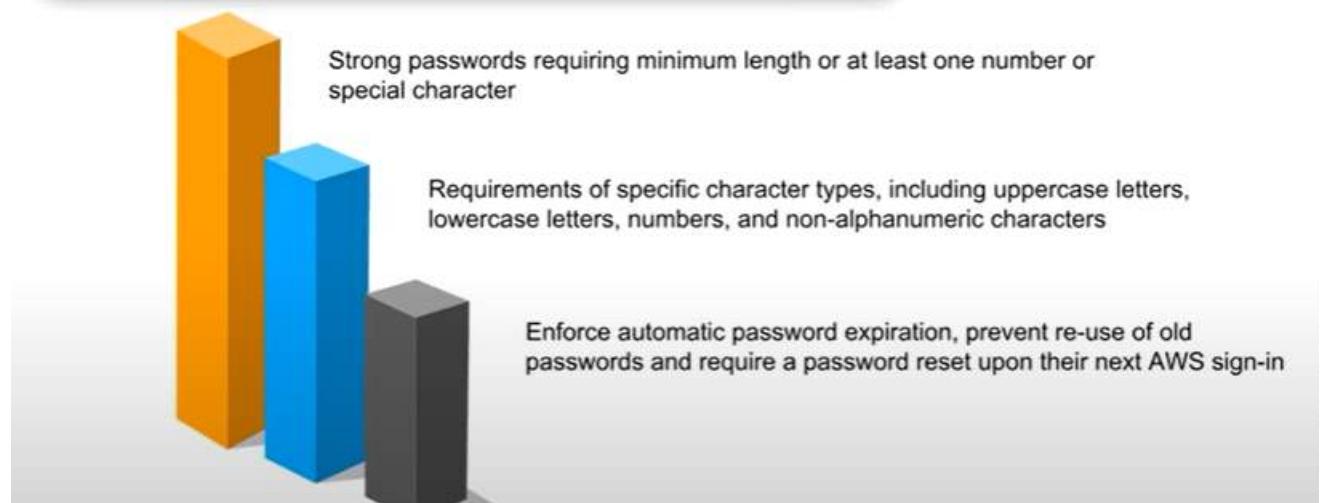
Wednesday, October 5, 2022 5:12 PM

## Identity and Access Management

Using AWS IAM you can do the following:



Using AWS IAM, you can set the following policies for your users' passwords:



# IAM Role - IAM User

Wednesday, October 5, 2022 5:13 PM

## IAM Role

An IAM role is an IAM entity that defines a set of permissions for making AWS service requests

Trusted entities, such as IAM users, applications, or an AWS service (e.g. EC2) assume roles

An IAM user has permanent long-term credentials and is used to directly interact with AWS services

Here, the IAM user has full access to all AWS IAM functionalities

# IAM Policy

Wednesday, October 5, 2022 5:14 PM

The following policy is used to grant access to add, update, and delete objects from a specific folder 'example\_folder' in a specific bucket 'example\_bucket'

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3:PutObject",  
                "s3:GetObject",  
                "s3:GetObjectVersion",  
                "s3:DeleteObject",  
                "s3:DeleteObjectVersion"  
            ],  
            "Resource": "arn:aws:s3:::example_bucket/example_folder/*"  
        }  
    ]  
}
```

A **policy summary** lists the **access level**, **resources** and **conditions** for each service defined in a policy

Service	Access level	Resource	Request condition
<b>Allow (10 of 94 services)</b>			
CloudFormation	Full, List Limited: Read, Write	All resources	None
CloudWatch Logs	Full access	Multiple	None
EC2	Full, List Limited: Read	All resources	None
Elastic Beanstalk	Full access	All resources	elasticbeanstalk:InApplication = arn:aws:elasticbeanstalk:*.111122223333:application/Bank-Devl

# Key Recovery

Wednesday, October 5, 2022 3:04 PM



Follow the below steps to recover or login to an EC2 instance to which you have lost the key:

- Step 1:** Verify that the EC2Config service is running
- Step 2:** Detach the root volume from the instance
- Step 3:** Attach the volume to a temporary instance
- Step 4:** Modify the configuration file
- Step 5:** Restart the original instance

# Load Balancer

Wednesday, October 5, 2022 5:08 PM

## Elastic Load Balancing

### 42. What are the different types of load balancers in AWS?

There are three types of load balancers that are supported by Elastic Load Balancing:

1. Application Load Balancer
2. Network Load Balancer
3. Classic Load Balancer

### 43. What are the different uses of the various load balancers in AWS Elastic Load Balancing?

#### *Application Load Balancer*

Used if you need flexible application management and TLS termination.

#### *Network Load Balancer*

Used if you require extreme performance and static IPs for your applications.

#### *Classic Load Balancer*

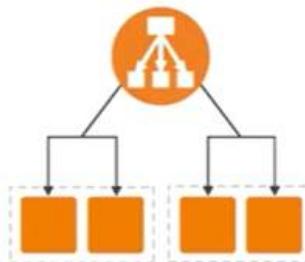
Used if your application is built within the EC2 Classic network

There are 3 types of load balancers which are supported by Elastic Load Balancing:

## 1 Application Load Balancer

### 1. Application Load Balancer

- An application load balancer makes routing decisions at the application layer (HTTP/HTTPS)
- It supports path based routing
- It can route requests to one or more ports on each container



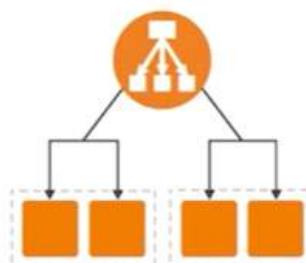
inflearn All rights reserved

There are 3 types of load balancers which are supported by Elastic Load Balancing:

## 2 Network Load Balancer

### 2. Network Load Balancer

- A Network Load Balancer makes router decisions at the transport level
- It handles millions of requests per second
- After the load balancer receives a connection, it selects a target group for the default rule using a flow hash routing algorithm



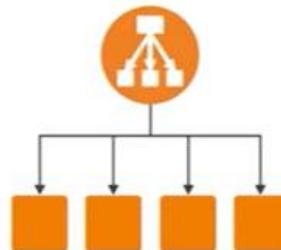
inflearn All rights reserved

There are 3 types of load balancers which are supported by Elastic Load Balancing:

### 3 Classic Load Balancer

#### 3. Classic Load Balancer

- A Classic Load Balancer makes routing decisions either on the transport layer (TCP/SSL) or the application layer (HTTP/HTTPS)
- It requires a fixed relationship between the load balancer port and the container port



SI

### 1 Application Load Balancer

If you need flexible application management and TLS termination

### 2 Network Load Balancer

If you require extreme performance and static IPs for your applications

3

Classic Load  
Balancer

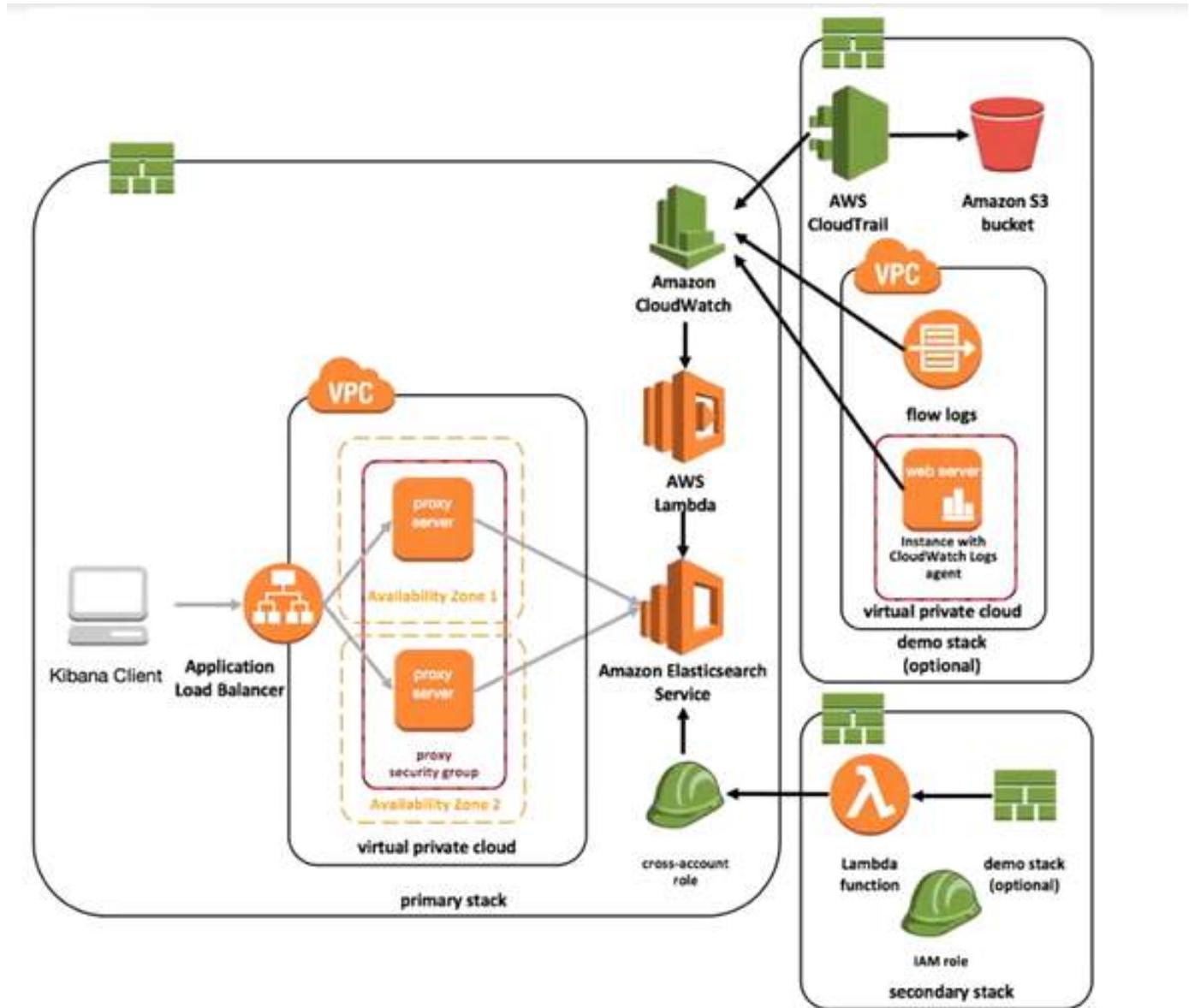
If your application is built  
within the EC2 Classic  
network

# Log Management

Wednesday, October 5, 2022 12:52 PM

- Help org to track relationship between operational, security and change management service
- Help to understand infra

AWS CloudWatch Logs > S3 > ElasticSearch to visualize them > Use Kinesis to move Data from S3 to Elastic



1. Deploy Amazon elastic search cluster along with two AZ of VPC network
2. Two Instance with proxy serve as an additional layer of security to restrict access to Amazon ES dashboard
3. A custom Lambda function is used to load the data from CloudWatch to an ES domain
4. Only those user requests from approved IP address will be allowed access to the kibana UI using customer-defined credentials

# Lambda

Tuesday, 25 October 2022 11:50 AM

Serverless service that lets you run code without thinking about servers.

- Function name
- Execution role
- Existing role
- Lambda function code

# Monitor website metrics in real time

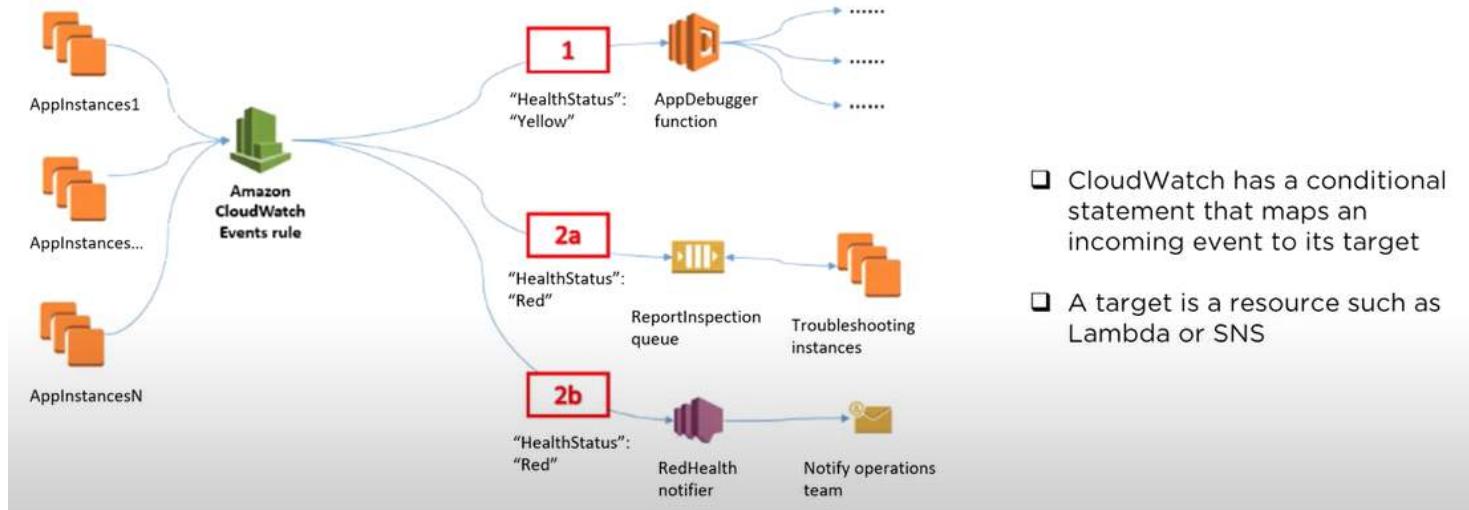
Wednesday, October 5, 2022 1:14 PM



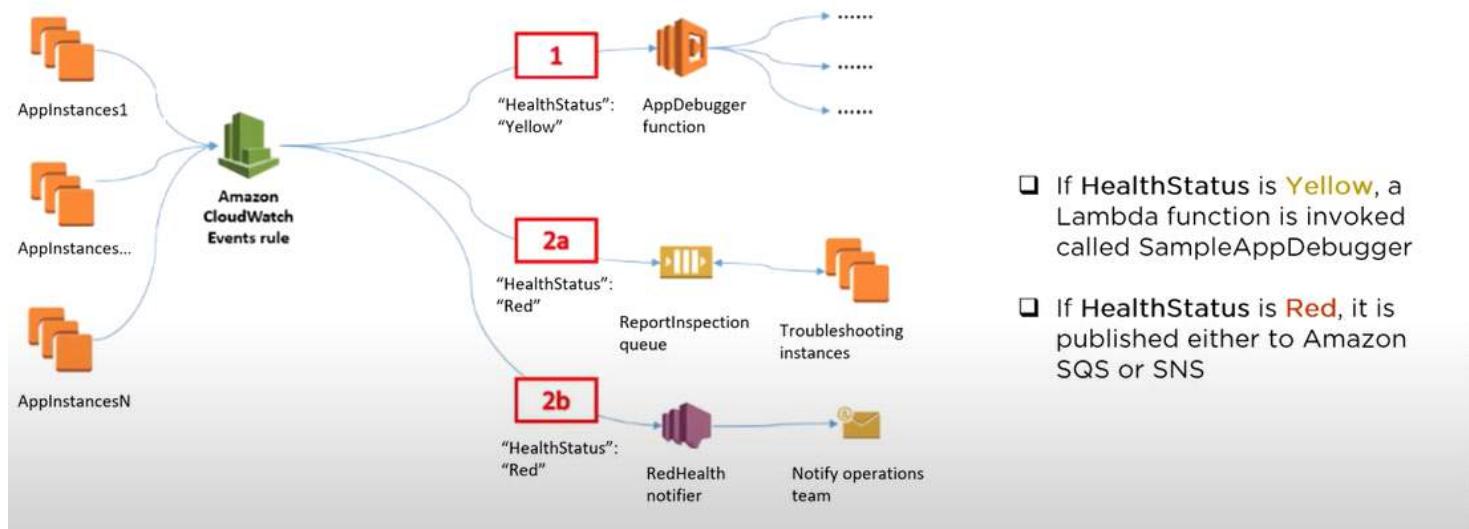
AWS CloudWatch

- ❑ CloudWatch events helps us to monitor application status of various AWS services and custom events

- ❑ Using CloudWatch we can monitor:
  1. State changes in Amazon EC2
  2. Auto-scaling lifecycle events
  3. Scheduled events
  4. AWS API calls
  5. Console sign-in events



- ❑ CloudWatch has a conditional statement that maps an incoming event to its target
- ❑ A target is a resource such as Lambda or SNS

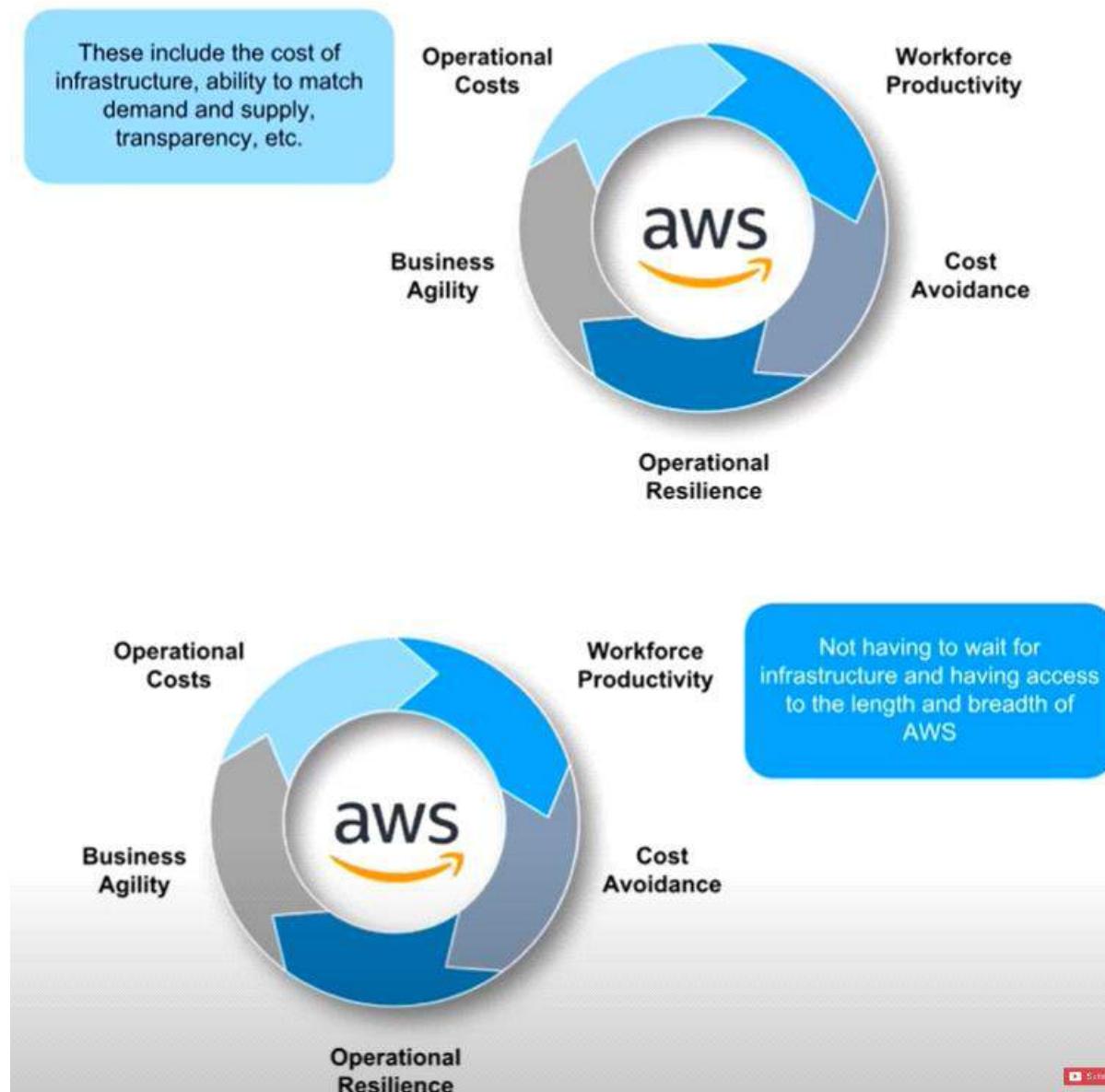


- ❑ If **HealthStatus** is **Yellow**, a Lambda function is invoked called **SampleAppDebugger**
- ❑ If **HealthStatus** is **Red**, it is published either to Amazon SQS or SNS

# Migration to AWS

Wednesday, October 5, 2022 4:47 PM

You will consider the following:





# Multi Choice

Wednesday, October 5, 2022 4:25 PM

- Database Service for single digit millisecond latency
  - Amazon RDS (Relational database Service - SQL- Paas) Good for Banking
  - Amazon Neptune (Graph database)
  - Amazon Snowball (It's storage)
  - Amazon DynamoDB (Key value store DB)
- Real time Monitoring
  - Amazon Firewall Manager
  - Amazon GuardDuty (Threads Monitoring Only)
  - Amazon CloudWatch
  - Amazon EBS
- Mobile Platform add user sign up, sign in and access control to your web
  - AWS Shield (DDOS protection)
  - AWS Macie (Security service Machine Learning)
  - AWS Inspector (Improving the security in App)
  - Amazon Cognito (Admin for control access mobile and web app)

A customer wants to capture all client connection information from his load balancer at an interval of 5 minutes, which of the following options should he choose for his application?



- a) Enable AWS CloudTrail for the load balancer
- b) CloudTrail is enabled globally
- c) Install the Amazon CloudWatch Logs agent on the load balancer.
- d) Enable Amazon CloudWatch metrics on the load balancer.

# Not region specific AWS Services

Wednesday, October 5, 2022 1:19 PM



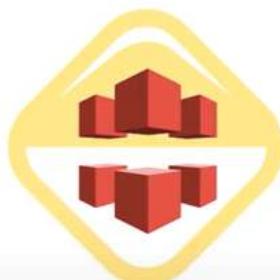
IAM



Route 53



Web Application Firewall



CloudFront



IAM

IAM Users, Groups, Roles & Accounts can be used globally across all regions

Route 53

Web Application Firewall



Route 53

All Route53 services are offered at AWS edge locations and are global

Web Application Firewall

CloudFront



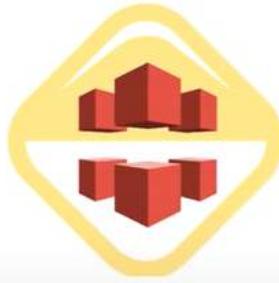
Web Application Firewall

Web Application Firewall which protects web applications from common web exploits are offered at AWS edge locations and are global

IAM

Route 53

CloudFront is the global Content Delivery Network (CDN) service which is offered at AWS edge locations



CloudFront

Route 53

Web Application Firewall

# NAT Gateway NAT Instance

Wednesday, October 5, 2022 1:22 PM

## Network Address Translation

The following are the key differences between NAT Gateway and NAT Instance:

Feature	NAT Gateway	NAT Instance
Availability	High	High
Bandwidth	Up to 45 Gbps	Depends on instance bandwidth
Maintenance	Managed by AWS	Managed by you
Performance	Very Good	Average
Cost	Number of gateways, duration and amount of usage	Number of instances, duration, amount and type of usage
Size and load	Uniform	As per your need
Security Groups	Cannot be assigned	Can be assigned

# RDS Horizontal Vertical Scaling

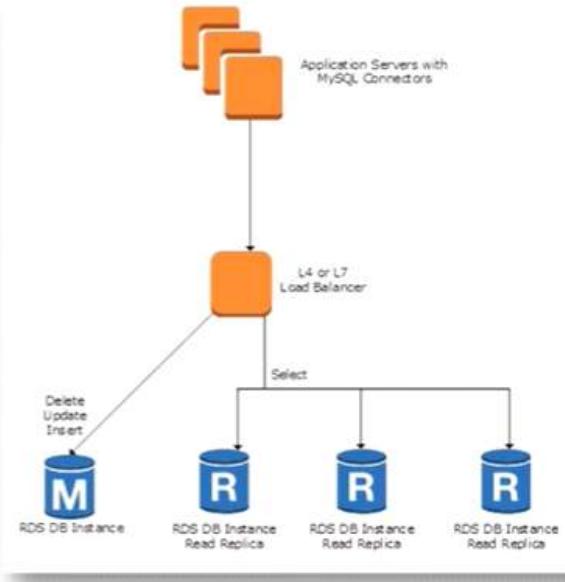
Wednesday, October 5, 2022 5:19 PM

27

Which type of scaling would you recommend for RDS and why?

## Horizontal Scaling

- You can also increase the performance of a read-heavy database by using read replicas to horizontally scale your RDS databases
- RDS MySQL, PostgreSQL and MariaDB can have up to 5 read replicas
- Amazon Aurora can have up to 15 read replicas

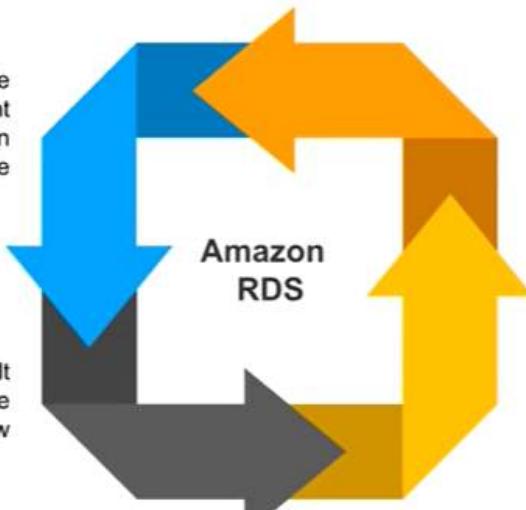


If you are looking to increase your storage and processing capacity, you can opt for **vertical scaling**

If you are looking at increasing performance of read-heavy database, you can opt for **horizontal scaling**

Your DB instance will still be available during these events though you might observe a minimal effect on performance

By default, a 30 minute default value is assigned as maintenance window



RDS maintenance window lets you decide when DB instance modifications, database engine version upgrades, and software patching have to occur

Automatic scheduling is done only for patches that are security and durability related

# RDS Cluster Master goes down.

Thursday, October 6, 2022 9:40 PM

Blue Green deployment not configured.

If master down I wont be able to write anymore on cluster  
But read operation still will be valid bcoz I will have multiple read replicas

Promote read replica to Master

# RTO RPO Disaster Recovery

Wednesday, October 5, 2022 4:51 PM

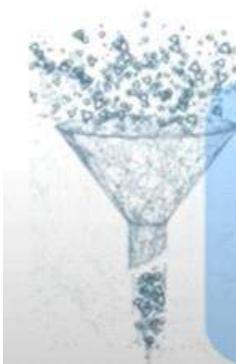
Both RTO and RPO are essentials in AWS Disaster Recovery

## Recovery Time Objective

RTO stands for Recovery Time Objective and it is the maximum time your company is willing to wait for the recovery to finish in case of an outage



RPO is Recovery Point Objective which is the maximum amount of data loss your company is willing to accept as measured in time



# Share EBS with Multi EC2

Monday, October 3, 2022 4:56 PM

Yes can share with Multi EC2 which are running on Nitro Hypervisor : type C5 Instance  
All Instance must be in same AZ.

Create EBS

```
Volume type: io2
Same AZ
Multi-Attach: Enable
lsblk
File -s /dev/nvme1n1
```

# Scalability vs Elasticity

Thursday, October 6, 2022 10:19 PM

- Scalability is the ability of a system to handle the increased load on its current hardware and software resources
- Elasticity is the ability of a system to increase the workload by increasing the hardware/software resources dynamically

# Security logging capabilities

Wednesday, October 5, 2022 1:04 PM

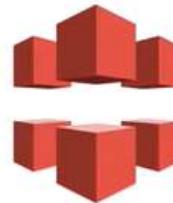
- ❑ Every service in AWS provides metrics or log files to provide insight on how that service is operating
- ❑ The following provide the AWS service-specific log recommendations:



AWS CloudTrail



AWS Config



AWS CloudFront



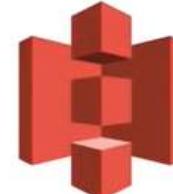
AWS Redshift



AWS RDS



AWS VPC

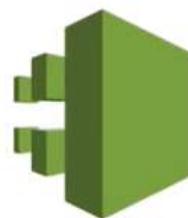


S3



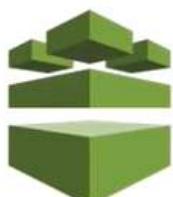
S3

All rights reserved



AWS CloudTrail

- ❑ AWS CloudTrail provides a history of AWS API calls for every account
- ❑ You can perform security analysis, resource change tracking and compliance auditing of your AWS environment
- ❑ It delivers log files to a designated S3 bucket every 5 minutes
- ❑ It can be configured to send notifications via AWS SNS when new logs are delivered



AWS Config

- ❑ AWS Config provides an AWS inventory which includes configuration history, configuration change notification and relationships between AWS resources
- ❑ It provides a timeline of resource configuration change for specific services
- ❑ It records the cumulative changes if many changes are made within a short period of time
- ❑ It can also be configured to send notifications via AWS SNS when new logs are delivered

# Service in particular region

Wednesday, October 5, 2022 1:11 PM

- ❑ As of now, not all services are available in all regions. This is because of the high infrastructure and maintenance costs
- ❑ Here is a short snippet of the available regions for various services

## Region Table

Last updated: August 06, 2018

Americas	Europe / Middle East / Africa	Asia Pacific					
Services Offered:	Northern Virginia	Ohio	Oregon	Northern California	Montreal	São Paulo	GovCloud
Alexa for Business	✓						
Amazon API Gateway	✓	✓	✓	✓	✓	✓	✓
Amazon AppStream 2.0	✓		✓				
Amazon Athena	✓	✓	✓				
Amazon Aurora - MySQL-compatible	✓	✓	✓	✓	✓		✓
Amazon Aurora - PostgreSQL-compatible	✓	✓	✓	✓	✓	✓	

Artifact ID: Windows

# Stateless and Stateful

Thursday, October 6, 2022 9:48 PM

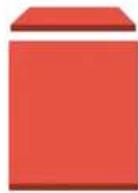
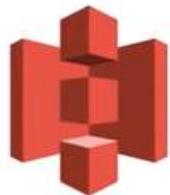
Stateful: Where server remembers whatever user executed the task let's user running 10 jobs so 5 servers are there and they know with each other whether the job is done by other server or not.

Stateless: where servers are not aware what has happened and what is going to happen.

So make sure application is stateful

# S3 vs EBS

Wednesday, October 5, 2022 3:07 PM



Feature	AWS S3	AWS EBS
Paradigm	Object Store	Filesystem
Performance	Fast	Superfast
Redundancy	Across data centers	Within a data center
Security	Using public or private key	Can be used only with EC2

# Stop Termination Instance

Wednesday, October 5, 2022 2:47 PM

## Stopping an instance

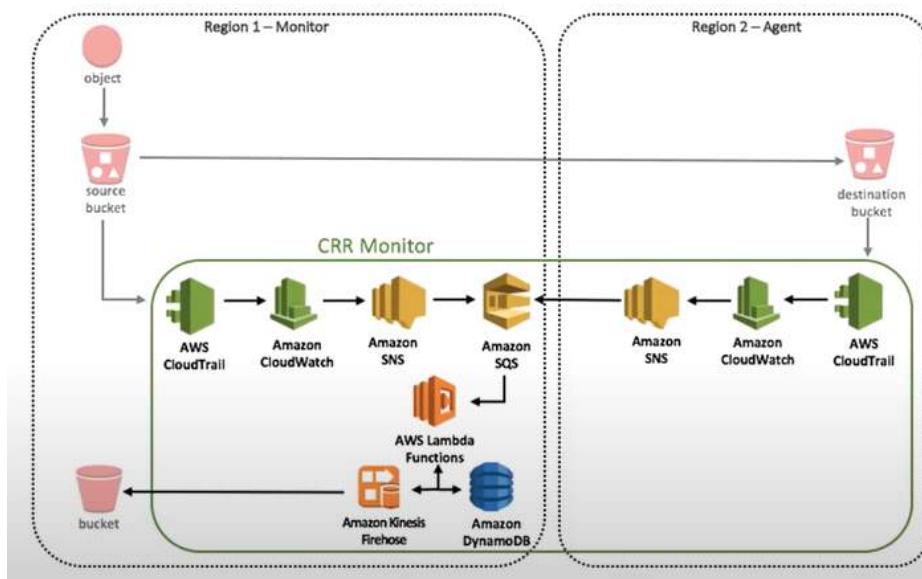
When you stop an instance, it performs a normal shutdown on the instance and moves to a stopped state

## Terminating an instance

Here, the instance is moved to a shutdown state and its attached EBS volumes are deleted unless you have set `deleteOnTermination` to 'False'

# S3 cross region replication check

Wednesday, October 5, 2022 3:15 PM



Cross-Region Replication Monitor (CRR Monitor) application is used to monitor the replication status of your Amazon S3 objects

# S3 bucket User access

Wednesday, October 5, 2022 3:13 PM



AWS S3 Bucket

We will follow the following four steps to allow access to a certain bucket:

**Step 1:** Categorize your instances

**Step 2:** Define how authorized users can (or can't) manage specific servers

**Step 3:** Lock down your tags

**Step 4:** Attach your policies to IAM users

# Types of EC2 based on costs

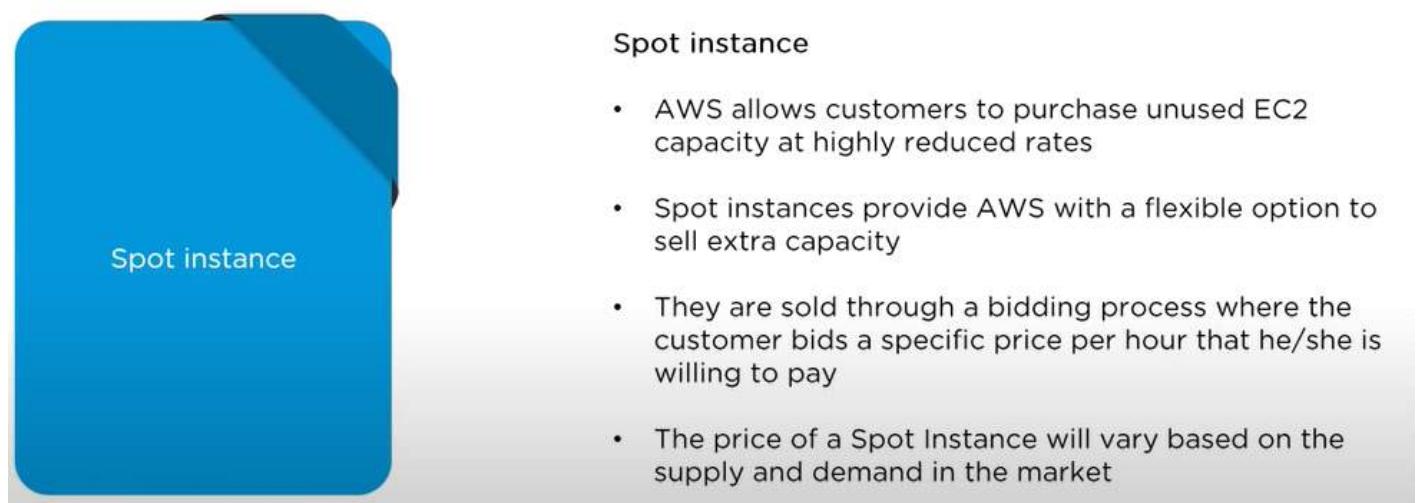
Wednesday, October 5, 2022 2:48 PM

- There are three types of Amazon EC2 instances based on costs:



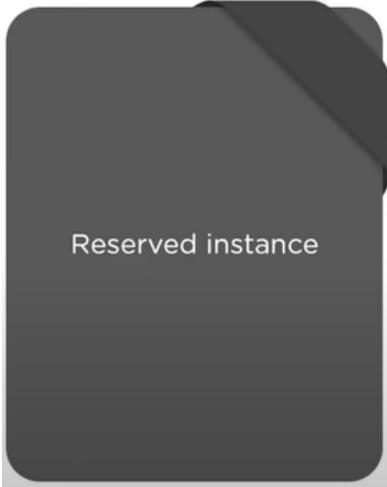
## On-demand instance

- These are EC2 instances that are purchased at a fixed rate per hour
- These are used for applications with short-term irregular workloads that cannot be interrupted
- These are best suited for development and testing of applications



## Spot instance

- AWS allows customers to purchase unused EC2 capacity at highly reduced rates
- Spot instances provide AWS with a flexible option to sell extra capacity
- They are sold through a bidding process where the customer bids a specific price per hour that he/she is willing to pay
- The price of a Spot Instance will vary based on the supply and demand in the market



## Reserved instance

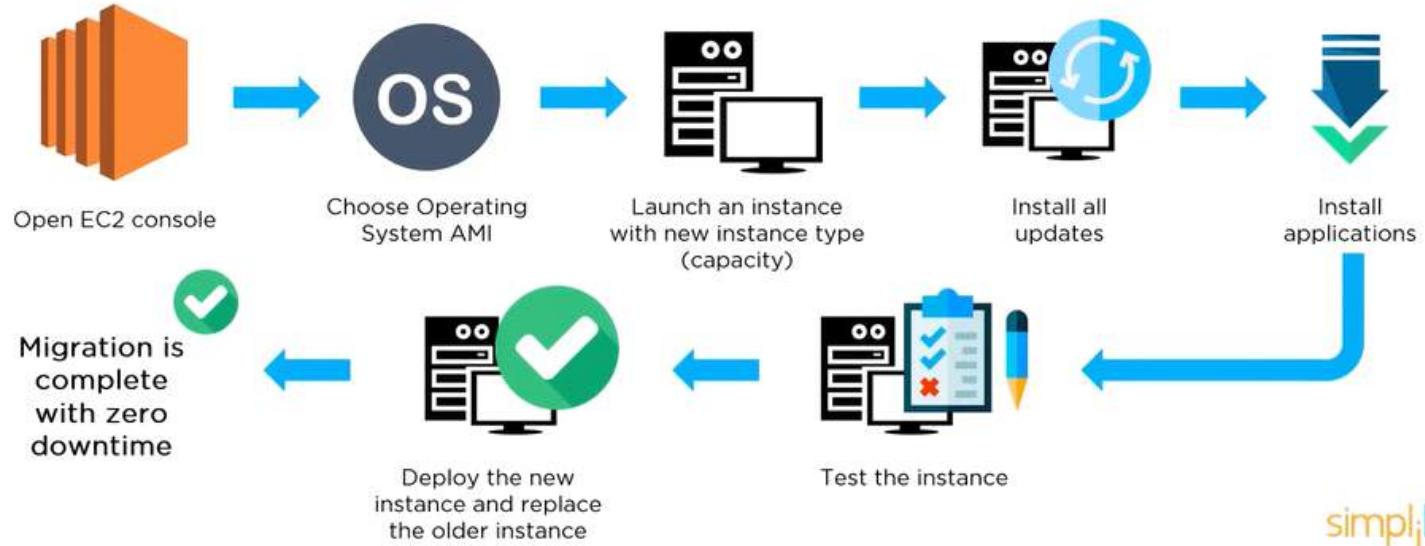
### Reserved instance

- Reserved instances are majorly used for short-term and they provide cost savings for companies
- While purchasing Reserved instances, users can opt for no upfront payment, partial payment or full payment upfront
- Reserved instances are available in three types: **light, medium and heavy**

# Upgrade or downgrade with Zero downtime

Wednesday, October 5, 2022 12:42 PM

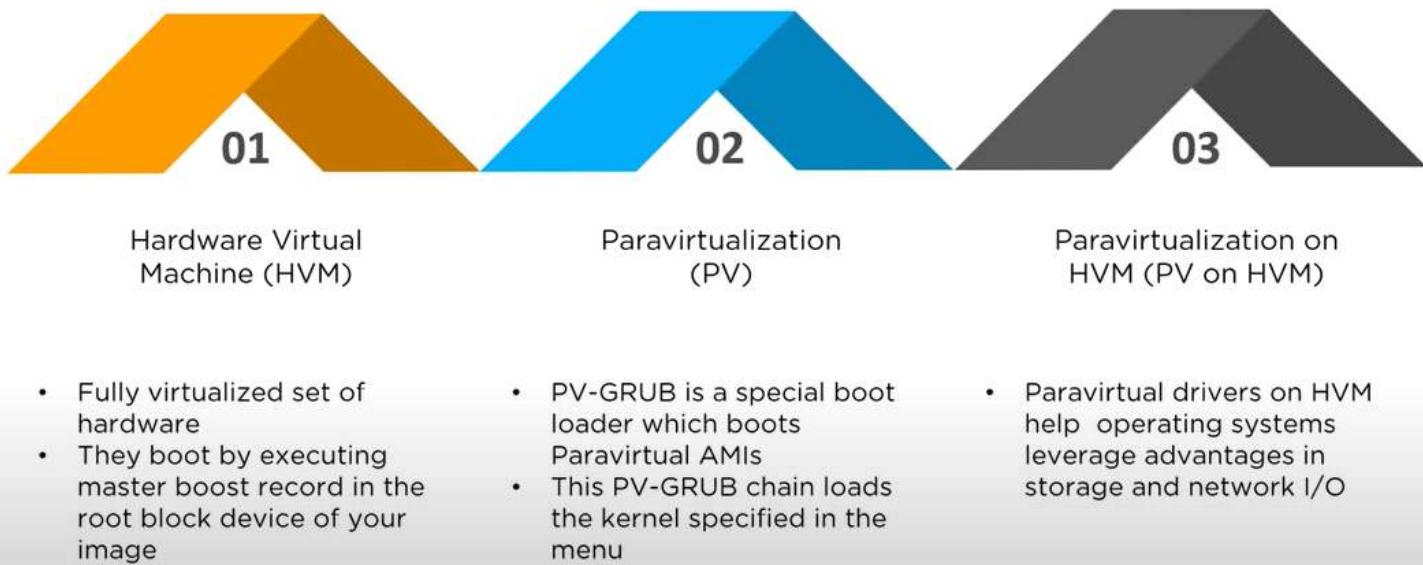
We can upgrade or downgrade a system with near zero downtime using the following steps of migration:



simplilearn

# Virtualization in AWS

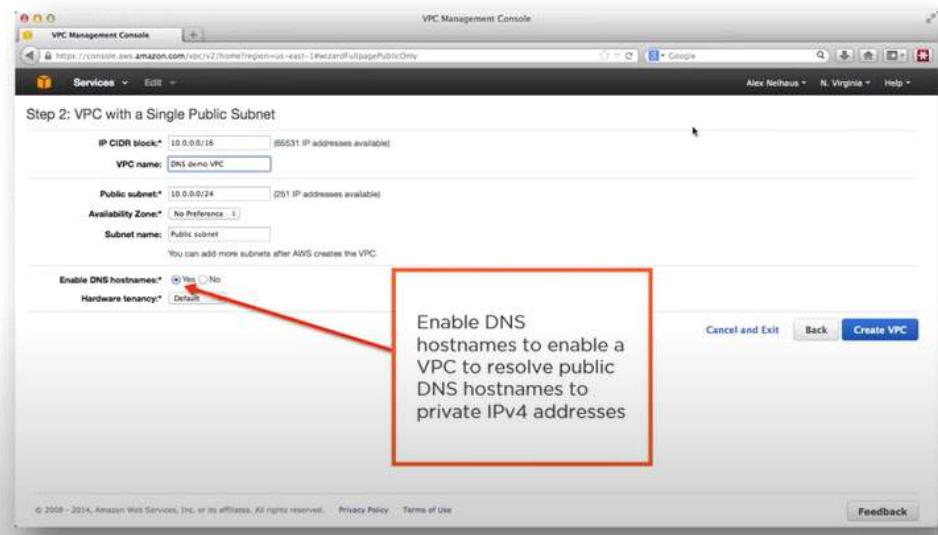
Wednesday, October 5, 2022 1:17 PM



# VPC not resolving the server through DNS

Wednesday, October 5, 2022 3:18 PM

To enable a VPC to resolve public IPv4 DNS hostnames to private IPv4 addresses when queried from instances in the peer VPC, you must modify the peering connection

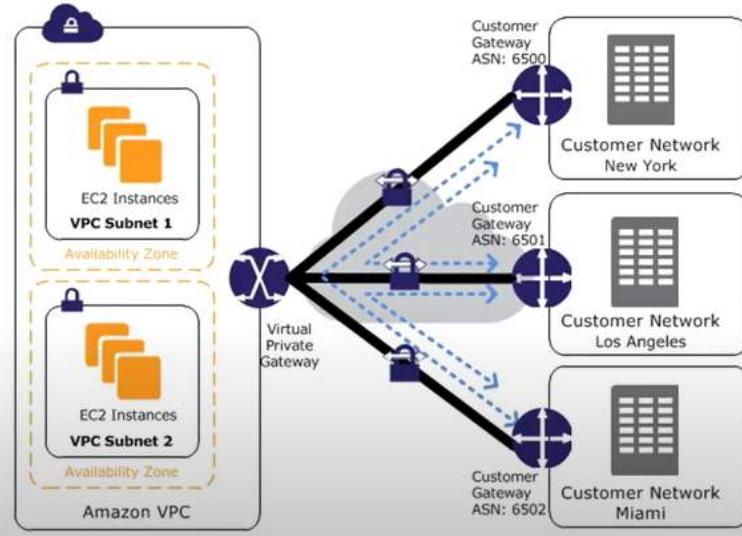


# VPC multiple sites connection

Wednesday, October 5, 2022 3:19 PM

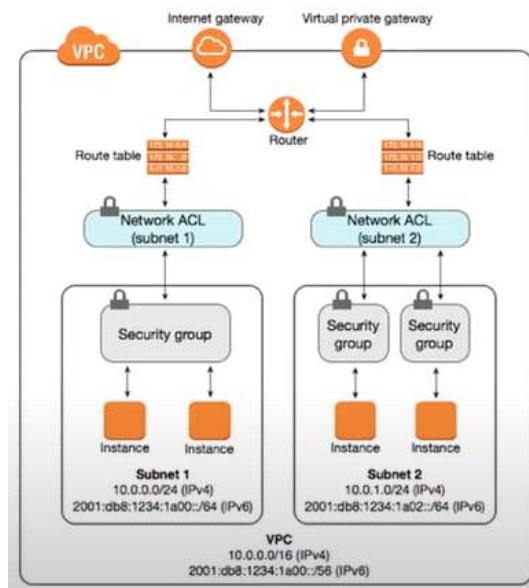
If you have multiple VPN connections, you can provide secure communication between sites using the AWS VPN CloudHub

You can connect multiple sites to a VPC as shown in this diagram



# VPC Security / ACL (NACL)

Wednesday, October 5, 2022 3:23 PM



- ❑ **Security groups** — Act as a firewall for associated Amazon EC2 instances, controlling both inbound and outbound traffic at the instance level
- ❑ **Network access control lists (ACLs)** — Act as a firewall for associated subnets, controlling both inbound and outbound traffic at the subnet level
- ❑ **Flow logs** — Capture information about the IP traffic going to and from network interfaces in your VPC

# VPC Monitor

Wednesday, October 5, 2022 3:27 PM

You can monitor Amazon VPC using the following:

- CloudWatch and CloudWatch Logs
- VPC Flow Logs



A screenshot of the AWS CloudWatch Logs interface. The left sidebar shows navigation links: Dashboard, Alarms, CloudWatch Metrics, Logs (selected), Metrics, CloudWatch Metrics, EBS, and EC2. The main area shows a hierarchical tree: Log Groups > Streams for MyFirstFlowLog > Events for eni-30076669-all. A table titled "Event Data" displays log entries. The first few entries are as follows:

Event ID	Source	Destination	Action	Timestamp
# 2	493062987015	eni-30076669	107.170.242.27	172.31.8.238 123 123 17 1 76 1433804902 1433807038 ACCEPT OK
# 2	493062987015	eni-30076669	172.31.8.238	107.170.242.27 123 123 17 1 76 1433804982 1433807038 ACCEPT OK
# 2	493062987015	eni-30076669	79.33.7.58	172.31.8.238 54517 23 6 3 180 1433807174 1433807218 REJECT OK
# 2	493062987015	eni-30076669	71.6.105.131	172.31.8.238 15214 21379 6 1 40 1433807224 1433807278 REJECT OK
# 2	493062987015	eni-30076669	172.31.8.238	108.41.54.35 123 123 17 1 76 1433807281 1433807338 ACCEPT OK
# 2	493062987015	eni-30076669	108.41.54.35	172.31.8.238 123 123 17 1 76 1433807281 1433807338 ACCEPT OK
# 2	493062987015	eni-30076669	172.31.8.238	23.226.142.216 123 123 17 1 76 1433807350 1433807399 ACCEPT OK
# 2	493062987015	eni-30076669	188.191.1.215	172.31.8.238 40082 24 6 1 40 1433807393 1433807398 REJECT OK
# 2	493062987015	eni-30076669	23.226.142.216	172.31.8.238 123 123 17 1 76 1433807350 1433807398 ACCEPT OK
# 2	493062987015	eni-30076669	50.114.38.147	172.31.8.238 123 123 17 1 76 1433807411 1433807458 ACCEPT OK
# 2	493062987015	eni-30076669	172.31.8.238	50.116.38.147 123 123 17 1 76 1433807411 1433807458 ACCEPT OK
# 2	493062987015	eni-30076669	107.170.242.27	172.31.8.238 123 123 17 1 76 1433807329 1433807579 ACCEPT OK
# 2	493062987015	eni-30076669	172.31.8.238	107.170.242.27 123 123 17 1 76 1433807529 1433807579 ACCEPT OK
# 2	493062987015	eni-30076669	118.211.0.90	172.31.8.238 46372 0000 6 1 40 1433807648 1433807699 REJECT OK
# 2	493062987015	eni-30076669	220.122.32.157	172.31.8.238 58394 22 6 3 180 1433807094 1433807999 REJECT OK
# 2	493062987015	eni-30076669	199.217.117.85	172.31.8.238 5064 5060 17 1 443 1433807944 1433807999 REJECT OK
# 2	493062987015	eni-30076669	107.170.242.27	172.31.8.238 123 123 17 1 76 1433808069 1433808119 ACCEPT OK
# 2	493062987015	eni-30076669	172.31.8.238	107.170.242.27 123 123 17 1 76 1433808069 1433808119 ACCEPT OK

# WAF - Web application Firewall

Wednesday, October 5, 2022 5:11 PM



sir

Some of the characteristics you can mention in AWS WAF are:



## VPC - Overview

Tuesday, October 19, 2021 6:37 PM

# What Is a VPC?

Think of a VPC as a virtual data center in the cloud.

- ✓ Logically isolated part of AWS Cloud where you can define your own network.
- ✓ Complete control of virtual network, including your own IP address range, subnets, route tables, and network gateways.

## Fully Customizable Network

You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances in each subnet.

### Web

Public-facing subnet.

### Application

Private subnet. Can only speak to web tier and database tier.

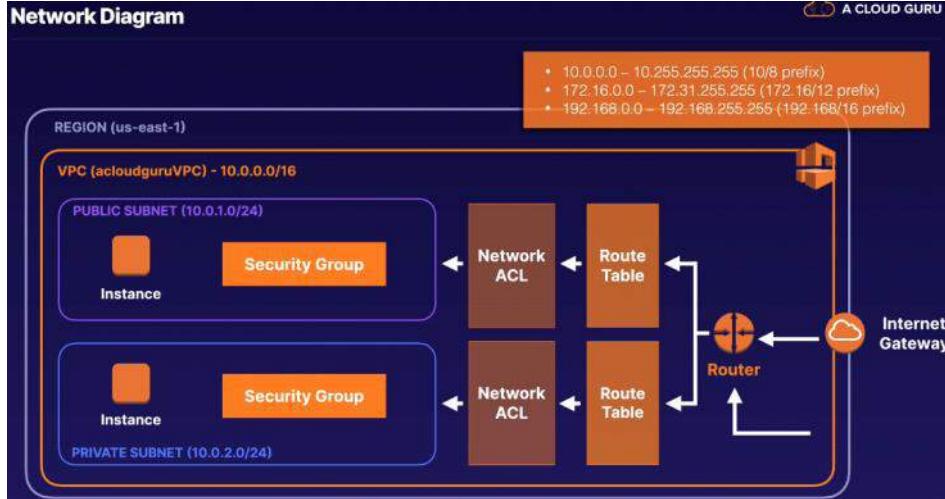
### Database

Private subnet. Can only speak to application tier.

Additionally, you can create a **hardware Virtual Private Network (VPN)** connection between your corporate data center and your VPC and leverage the AWS Cloud as an extension of your corporate data center.



## Network Diagram



## What can we do with a VPC?

**Launch Instances**  
Launch instances into a subnet of your choosing.

**Internet Gateway**  
Create internet gateway and attach it to our VPC.

### BONUS TIP

You can use network access control lists (**NACLs**) to block specific IP addresses.

**Custom IP Addresses**  
Assign custom IP address ranges in each subnet.

**More Control**  
Much better security control over your AWS resources.

**Route Tables**  
Configure route tables between subnets.

**Access Control Lists**  
Subnet network access control lists.

## Default VPC vs. Custom VPC

### Default

- Default VPC is user friendly.
- All subnets in default VPC have a route out to the internet.
- Each EC2 instance has both a public and private IP address.

### Custom

- Fully customizable.
- Takes time to set up.

## VPC Exam Tips

- ✓ Think of a VPC as a logical data center in AWS.
- ✓ Consists of internet gateways (or virtual private gateways), route tables, network access control lists, subnets, and security groups.
- ✓ 1 subnet is always in 1 Availability Zone.

# Building VPC Lab

Tuesday, October 19, 2021 10:14 PM

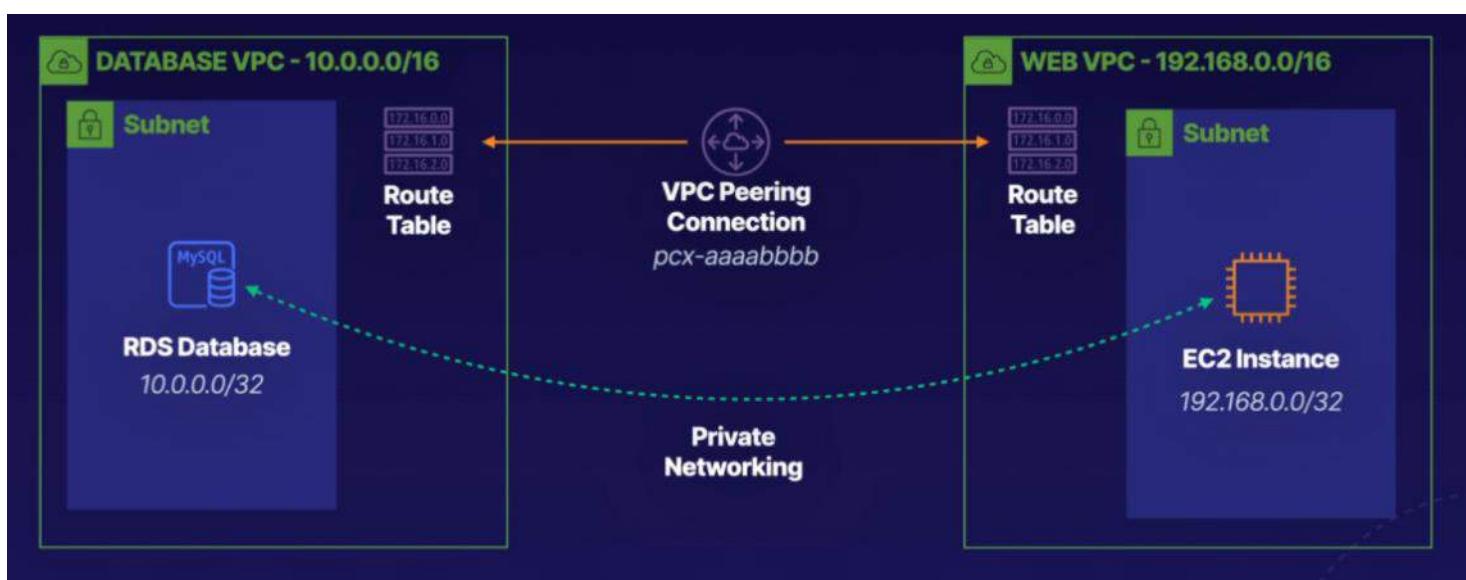
## Build Solutions across VPCs with Peering

### Introduction

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses. In this lab, we create a new VPC for our WordPress blog to run from. We then create a VPC peering connection between the new VPC and an existing database VPC. By the end of this lab, the user will understand how to create a new VPC from scratch, attach internet gateways, edit routing tables, and peer multiple VPCs together.

### Solution

Log in to the AWS Management Console using the credentials provided on the lab instructions page. Make sure you're in the N. Virginia (**us-east-1**) region throughout the lab.



Create Web\_VPC Subnets and Attach a New Internet Gateway

### Create a VPC

1. Navigate to VPC.
2. Under *Resources by Region*, click **VPCs**.
3. Click **Create VPC**.
4. Set the following values:
  - **Name tag:** **Web\_VPC**
  - **IPv4 CIDR block:** **192.168.0.0/16**
5. Leave the rest as their defaults and click **Create VPC**.

### Create a Subnet

3. On the left menu under *VIRTUAL PRIVATE CLOUD*, select **Subnets**.
4. Click **Create subnet**.
5. Select the newly created **Web\_VPC**.
6. Under *Subnet settings*, set the following values:
  - **Subnet name:** **WebPublic**
  - **Availability Zone:** **us-east-1a**
  - **IPv4 CIDR (Classless inter-domain routing) block:** **192.168.0.0/24**

7. Click **Create subnet**.

### Create an Internet Gateway

9. On the left menu, select **Internet Gateways**.
10. Click **Create internet gateway**.
11. In *Name tag*, enter "WebIG".
12. Click **Create internet gateway**.
13. In the green notification at the top of the page, click **Attach to a VPC**.
14. In *Available VPCs*, select the **Web\_VPC** and click **Attach internet gateway**.
15. On the left menu, select **Route Tables**.
16. Select the **Web\_VPC**.
17. Underneath, select the *Routes* tab and click **Edit routes**.
18. Click **Add route**.
19. Set the following values:
  - Destination*: **0.0.0.0/0**
  - Target*: **Internet Gateway > WebIG**
20. Click **Save changes**.

### Create a Peering Connection

1. On the left menu, select **Peering Connections**.
2. Click **Create peering connection**.
3. Set the following values:
  - Name*: **DBtoWeb**
  - VPC (Requester)*: **DB\_VPC**
  - VPC (Acceptor)*: **Web\_VPC**
4. Click **Create peering connection**.
5. At the top of the page, click **Actions > Accept Request**.
6. Click **Accept request**.
7. On the left menu, select **Route Tables**.
8. Select the **Web\_VPC**.
9. Underneath, select the *Routes* tab and click **Edit routes**.
10. Click **Add route**.
11. Set the following values:
  - Destination*: **10.0.0.0/16**
  - Target*: **Peering Connection > DBtoWeb**
12. Click **Save changes**.
13. Go back to *Route Tables* and select the **DB\_VPC** instance with a *Main* column value of Yes.
14. Under the *Routes* tab, click **Edit routes**.
15. Click **Add route**.
16. Set the following values:
  - Destination*: **192.168.0.0/16**
  - Target*: **Peering Connection > DBtoWeb**
17. Click **Save changes**.

### Create an EC2 Instance and Configure WordPress

1. In a new browser tab, navigate to EC2.
2. Click **Launch instance > Launch instance**.
3. Scroll down to *Ubuntu Server 20.04 LTS* and click **Select**.

4. Select **t3.micro** as the instance type.
5. Click **Next: Configure Instance Details**.
6. Set the following values:
  - **Network: Web\_VPC**
  - **Auto-assign Public IP: Enable**

7. At the bottom under *User data*, paste in the following bootstrap script:

```
#!/bin/bash
sudo apt update -y
sudo apt install php-curl php-gd php-mbstring php-xml php-xmlrpc php-soap php-intl php-zip perl
mysql-server apache2 libapache2-mod-php php-mysql -y
wget https://github.com/ACloudGuru-Resources/course-aws-certified-solutions-architect-
associate/raw/main/lab/5/wordpress.tar.gz
tar zxvf wordpress.tar.gz
cd wordpress
wget https://raw.githubusercontent.com/ACloudGuru-Resources/course-aws-certified-solutions-
architect-associate/main/lab/5/000-default.conf
cp wp-config-sample.php wp-config.php
perl -pi -e "s/database_name_here/wordpress/g" wp-config.php
perl -pi -e "s/username_here/wordpress/g" wp-config.php
perl -pi -e "s/password_here/wordpress/g" wp-config.php
perl -i -pe' BEGIN {
    @chars = ("a" .. "z", "A" .. "Z", 0 .. 9);
    push @chars, split //, "!@#$%^&*()-_ []{}<>~\`+=,.:/?|";
    sub salt { join "", map $chars[ rand @chars ], 1 .. 64 }
}
s/put your unique phrase here/salt()/ge
' wp-config.php
mkdir wp-content/uploads
chmod 775 wp-content/uploads
mv 000-default.conf /etc/apache2/sites-enabled/
mv /wordpress /var/www/
apache2ctl restart
```

8. Click **Review and Launch**.
9. Scroll down to *Security Groups* and click **Edit security groups**.
10. Click **Add Rule**.
11. Select **HTTP** and click **Review and Launch**.
12. Click **Launch**.
13. On the dropdown, select **Proceed without a key pair**.
14. Select the checkbox acknowledging that you will not be able to connect to this instance unless you already know the password built into this AMI.
15. Click **Launch Instances** and then click **View Instances**.
   
**Note:** It may take a few minutes for the new instance to launch.
16. In a new browser tab, navigate to RDS.
17. Select the provisioned RDS instance.
18. Under *Connectivity & security*, copy the RDS endpoint for later use.
19. Navigate back to EC2.

20. Select the new instance and click **Connect**.
21. Click **Connect**.
22. To confirm WordPress installed correctly, view the configuration files:  

```
cd /var/www/wordpress  
ls
```
23. To configure WordPress, open **wp-config.php**:  

```
sudo nano wp-config.php
```
24. Scroll down to **/\*\* MySQL hostname \*/** and replace **localhost** with the RDS endpoint previously copied.
25. To save, press **Ctrl+X**, and then type **Y** and press **Enter**.

### Modify the RDS Security Groups to Allow Connections from the Web\_VPC VPC

1. Navigate to RDS.
2. In *Connectivity & security*, click the active link under *VPC security groups*.
3. Select the *Inbound rules* tab and click **Edit inbound rules**.
4. Click **Add rule**.
5. Under *Type*, type and select **MySQL/Aurora**.
6. Under *Source*, type and select **192.168.0.0/16**.
7. Click **Save rules**.
8. Return to the terminal.
9. At the bottom of the terminal window, copy the public IP address of our server.
10. Open a new browser tab and paste the public IP address in the address bar. You should now see the WordPress installation page.
11. Set the the following values :
  - *Site Title*: **A Blog Guru**
  - *Username*: **guru**
  - *Your Email*: Your email address
12. Click **Install WordPress**.
13. Reload the public IP address in the address bar to view our newly created WordPress blog.

### Conclusion

Congratulations — you've completed this hands-on lab!

### Troubleshooting

If the website isn't loading the way you'd expected at the end of this guide, here are some tips to help troubleshoot:

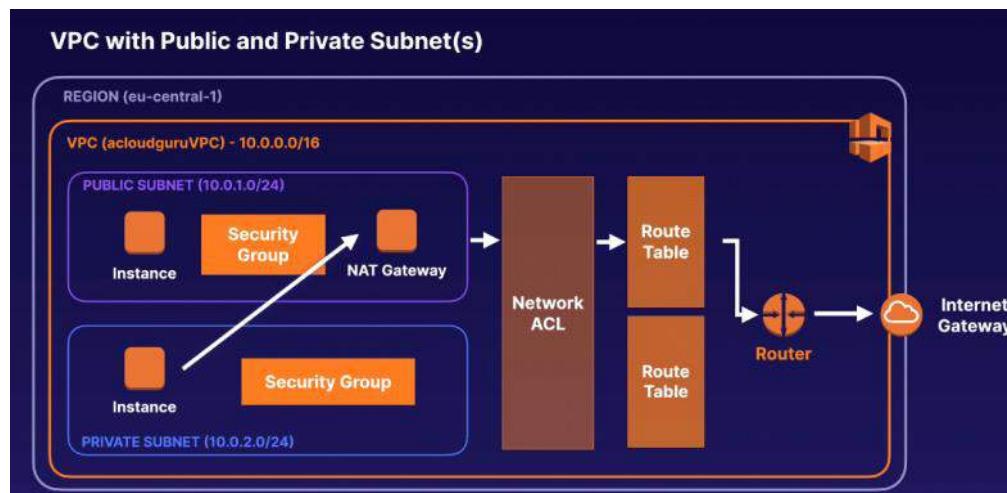
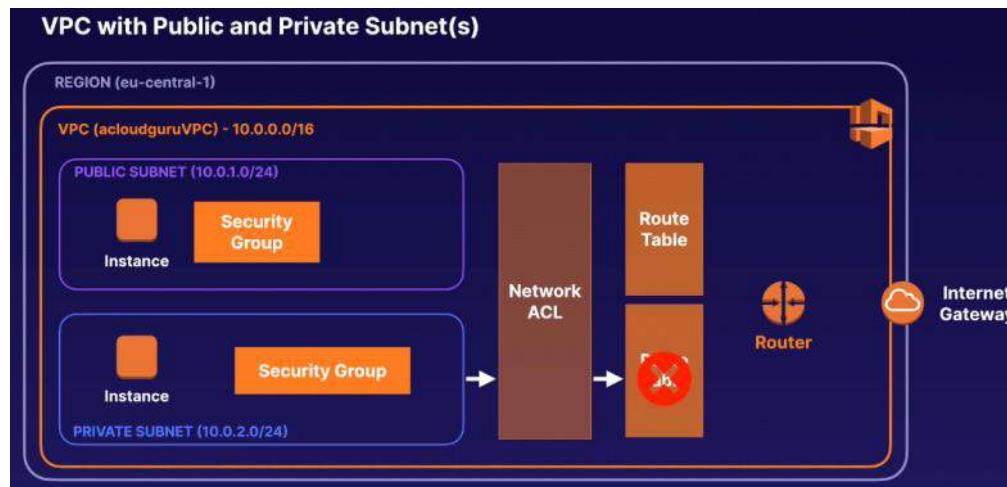
- Check the status of the lab objectives - are any not yet completed?
- Has everything we've setup successfully become ready to use? Check things like the VPC Peering Connection, which requires you to specifically accept the connection request.
- Does the database error page load after a minute or so of waiting, or does no page load at all? This gives a hint to whether the issue may be with the peering, or the security groups.

# NAT - Using NAT Gateways for Internet Access

Tuesday, October 19, 2021 11:23 PM



- ✓ Redundant inside the Availability Zone
- ✓ Starts at 5 Gbps and scales currently to 45 Gbps
- ✓ No need to patch
- ✓ Not associated with security groups
- ✓ Automatically assigned a public IP address



# Security Groups -Protecting Your Resources

Tuesday, October 19, 2021 11:35 PM

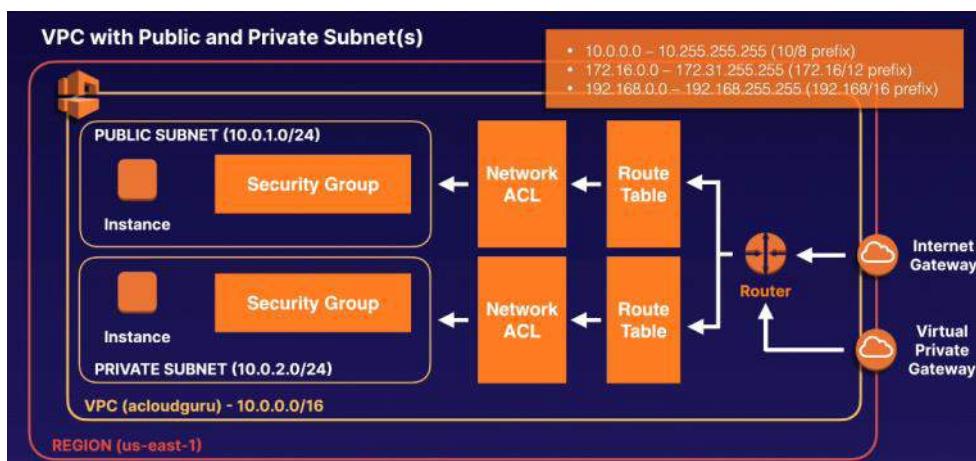
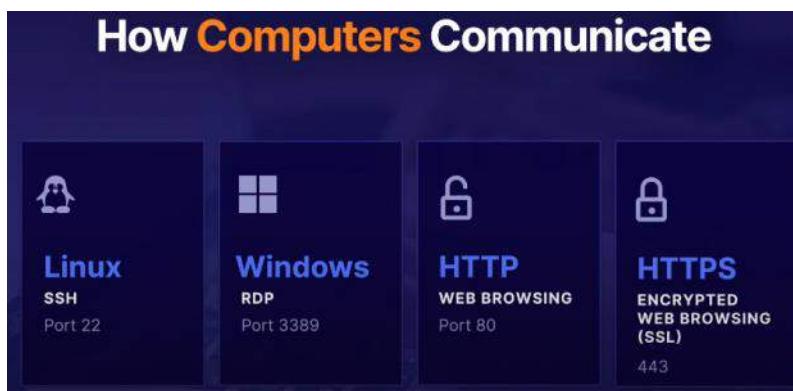


## Security Groups

Security groups are **virtual firewalls** for an EC2 instance. By default, everything is blocked.

TO LET EVERYTHING IN: 0.0.0.0/0

In order to communicate to your EC2 instances via SSH, RDP, or HTTP, you will need to open up the correct ports.



## Security Groups

Security groups are stateful — if you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules.

Responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules.

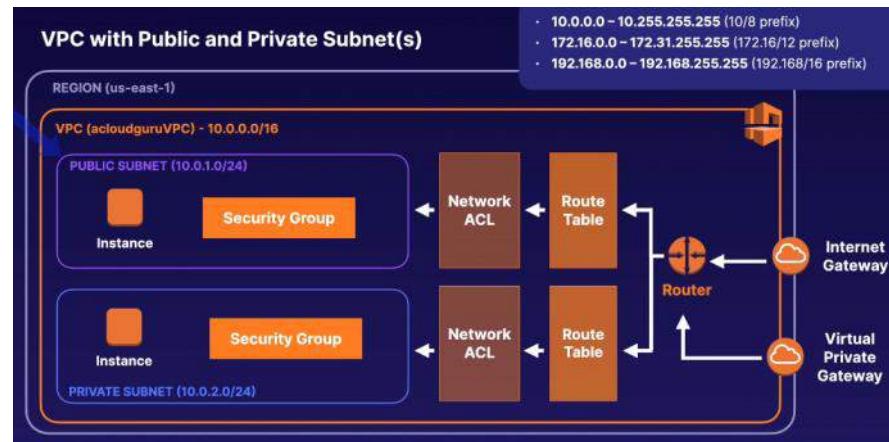
## ACLs - Controlling Subnet Traffic with Network ACLs

Tuesday, October 19, 2021 11:38 PM

# Network ACLs

## The first line of defense

- ✓ A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.
- ✓ You might set up network ACLs with rules similar to your security groups in order to add another layer of security to your VPC.



## Overview of Network ACLs

- ✓ **Default Network ACLs:** Your VPC automatically comes with a default network ACL, and by default it allows all outbound and inbound traffic.
- ✓ **Custom Network ACLs:** You can create custom network ACLs. By default, each custom network ACL denies all inbound and outbound traffic until you add rules.
- ✓ **Subnet Associations:** Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- ✓ **Block IP Addresses:** Block IP addresses using network ACLs, not security groups.

- ✓ You can associate a network ACL with multiple subnets; however, a subnet can be associated with **only 1 network ACL** at a time. When you associate a network ACL with a subnet, the previous association is **removed**.
- ✓ Network ACLs contain a **numbered list of rules** that are evaluated in order, starting with the **lowest** numbered rule.
- ✓ Network ACLs have **separate** inbound and outbound rules, and each rule can either **allow or deny traffic**.
- ✓ Network ACLs are **stateless**; responses to allowed inbound traffic are subject to the rules for outbound traffic (and vice versa).

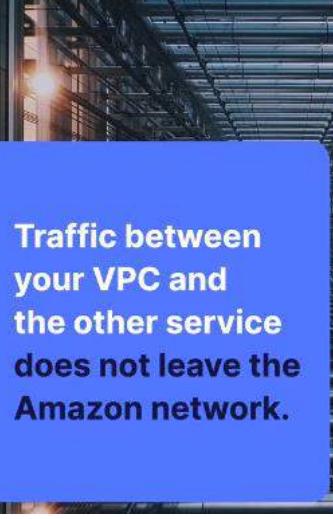
# Endpoint - Private Communication Using VPC Endpoints

Wednesday, October 20, 2021 12:13 AM

## VPC Endpoints

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection.

**Instances in your VPC do not require public IP addresses to communicate with resources in the service.**



Traffic between your VPC and the other service does not leave the Amazon network.

### 💡 STUDY TIP

## Endpoints Are Virtual Devices

They are horizontally scaled, redundant, and highly available VPC components that allow communication between instances in your VPC and services **without imposing availability risks or bandwidth constraints** on your network traffic.

## There are **2** types of endpoints

### OPTION 1

### INTERFACE ENDPOINTS

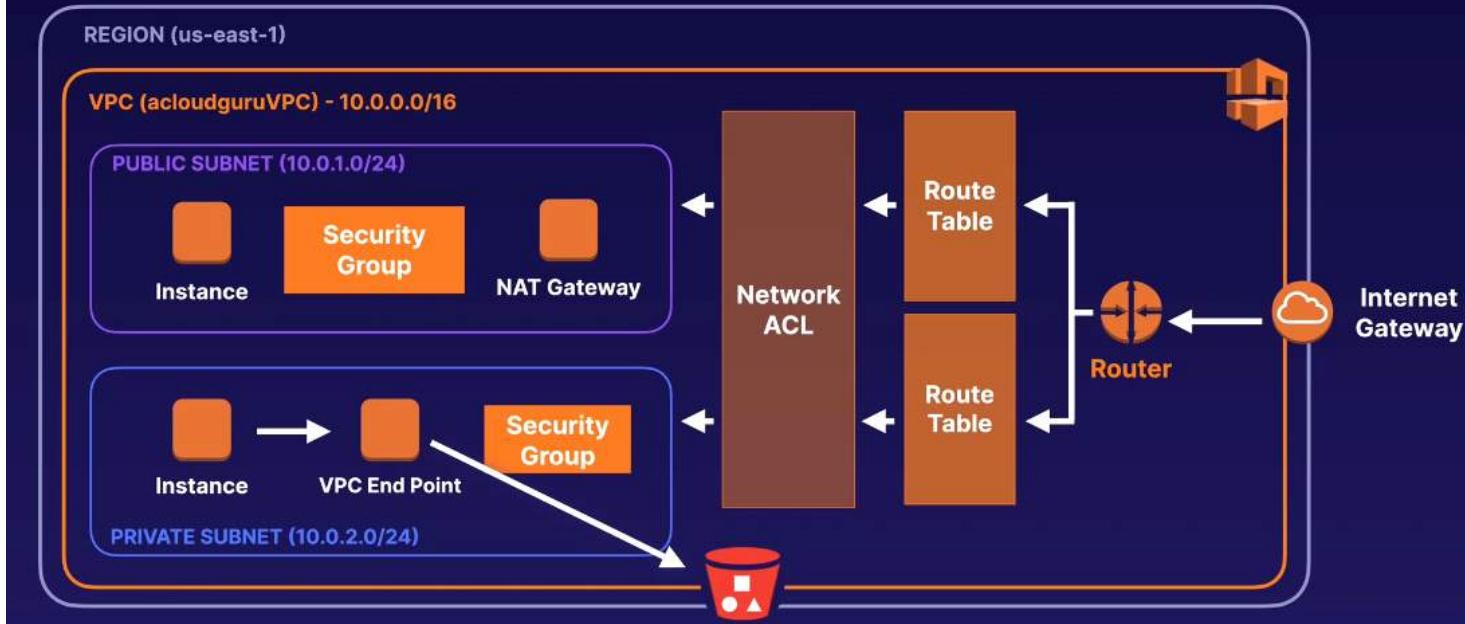
An interface endpoint is an **elastic network interface with a private IP address** that serves as an entry point for traffic headed to a supported service. They support a large number of AWS services.

### OPTION 2

### GATEWAY ENDPOINTS

Similar to NAT gateways, a gateway endpoint is a **virtual device you provision**. It supports connection to S3 and DynamoDB.

## VPC with Public and Private Subnet(s)



## VPC Endpoints

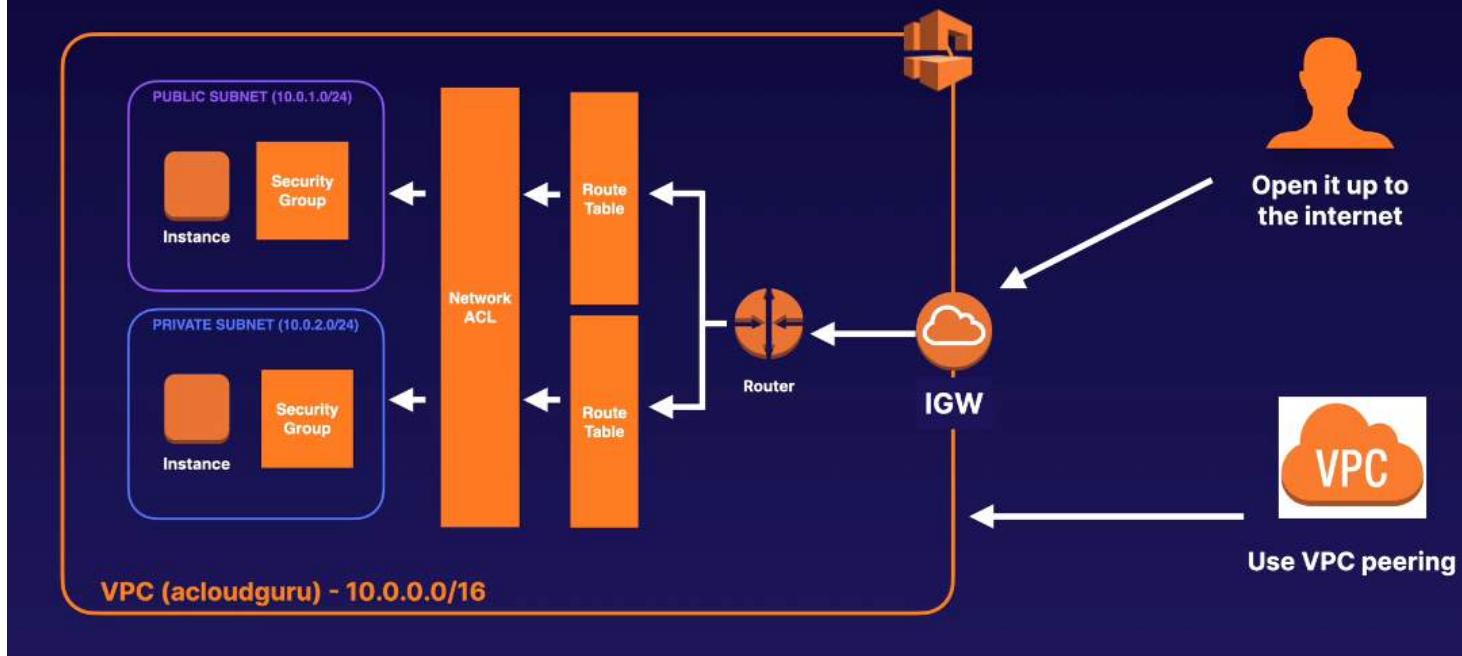
- ✓ **Use Case:** When you want to connect AWS services without leaving the Amazon internal network
- ✓ **2 Types of VPC Endpoints:** Interface endpoints and gateway endpoints
- ✓ **Gateway Endpoints:** Support S3 and DynamoDB

# PrivateLink - Network Privacy with AWS PrivateLink

Thursday, October 21, 2021 11:37 PM

A CLOUD GURU

## Opening Your Services in a VPC to Another VPC



To open our applications up to other VPCs, we can either:

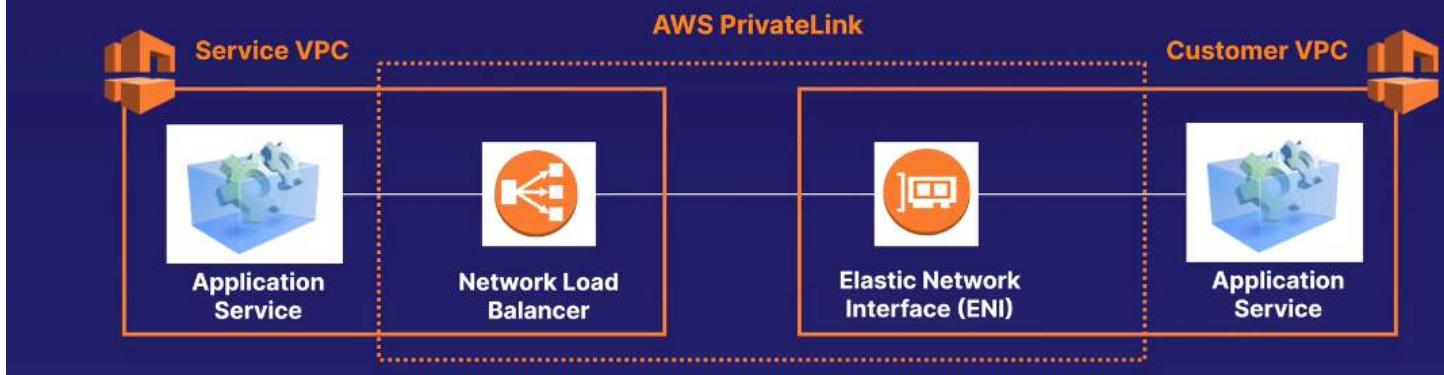
### Open the VPC up to the Internet

- Security considerations; everything in the public subnet is public
- A lot more to manage

### Use VPC Peering

- You will have to create and manage many different peering relationships.
- The whole network will be accessible. This isn't good if you have multiple applications within your VPC.

- ✓ The best way to expose a service VPC to tens, hundreds, or thousands of customer VPCs
- ✓ Doesn't require VPC peering; no route tables, NAT gateways, internet gateways, etc.
- ✓ Requires a Network Load Balancer on the service VPC and an ENI on the customer VPC



## AWS PrivateLink

- ✓ If you see a question asking about peering VPCs to tens, hundreds, or thousands of customer VPCs, think of AWS PrivateLink.
- ✓ Doesn't require VPC peering; no route tables, NAT gateways, internet gateways, etc.
- ✓ Requires a Network Load Balancer on the service VPC and an ENI on the customer VPC.

# Peering - Building Solutions across VPCs with Peering

Thursday, October 21, 2021 11:41 PM

## Multiple VPCs

Sometimes you may need to have several VPCs for different environments, and it may be necessary to connect these VPCs to each other.

**Production  
Web VPC**

**Content VPC**

**Intranet**



Allows you to connect 1 VPC with another via a direct network route using private IP addresses.



Instances behave as if they were on the same private network.



You can peer VPCs with other AWS accounts as well as with other VPCs in the same account.



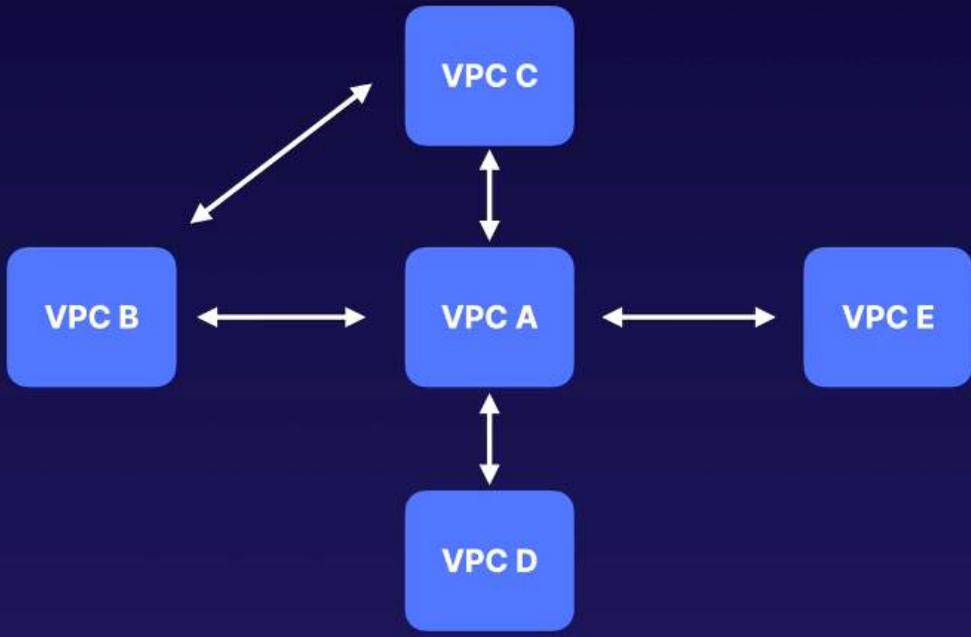
Peering is in a star configuration (e.g., 1 central VPC peers with 4 others). No transitive peering!



You can peer between regions.

VPC  VPC

## Transitive Peering



## VPC Peering

- ✓ Allows you to connect 1 VPC with another via a direct network route using private IP addresses.
- ✓ Transitive peering is not supported.  
This must always be in a hub-and-spoke model.
- ✓ You can peer between regions.
- ✓ No overlapping CIDR address ranges

# CloudHub - Securing Your Network with VPN CloudHub

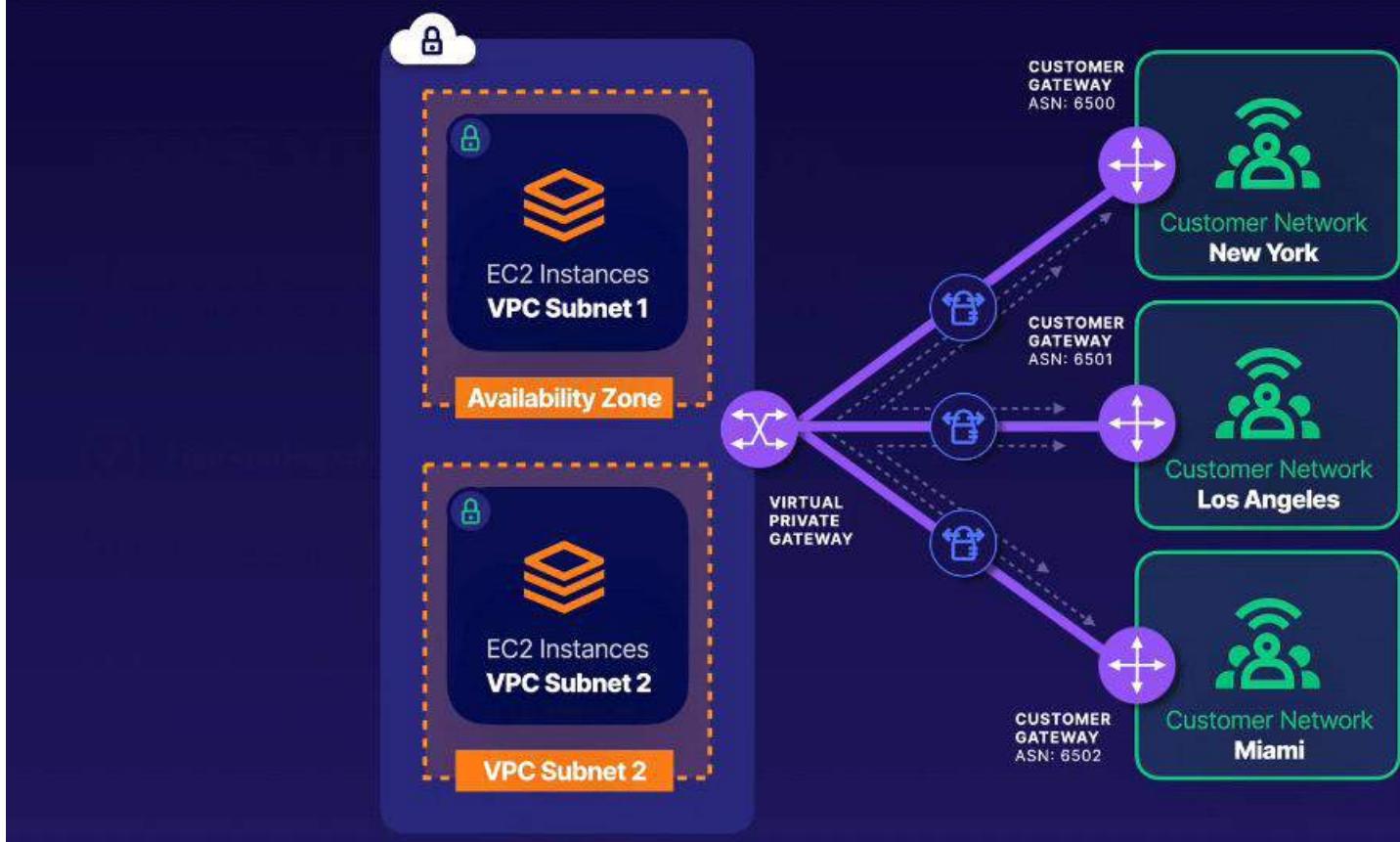
Thursday, October 21, 2021 11:46 PM

## AWS VPN CloudHub

If you have multiple sites, each with its own VPN connection, you can use AWS VPN CloudHub to connect those sites together.

- ✓ Hub-and-spoke model
- ✓ Low cost and easy to manage
- ✓ It operates over the public internet, but all traffic between the customer gateway and the AWS VPN CloudHub is encrypted.

### Network Diagram



# AWS VPN CloudHub

If you have multiple sites, each with its own VPN connection, you can use AWS VPN CloudHub to connect those sites together. It's similar to VPC peering in that it works on a hub-and-spoke model.

AWS VPN CloudHub is low cost and easy to manage. Though it operates over the public internet, all traffic between the customer gateway and the AWS VPN CloudHub is encrypted.

## What Is Direct Connect?

AWS Direct Connect is a cloud service solution that **makes it easy to establish a dedicated network connection** from your premises to AWS.



## Private Connectivity

Using AWS Direct Connect, you can establish private connectivity between AWS and your data center or office.

In many cases, you can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections.

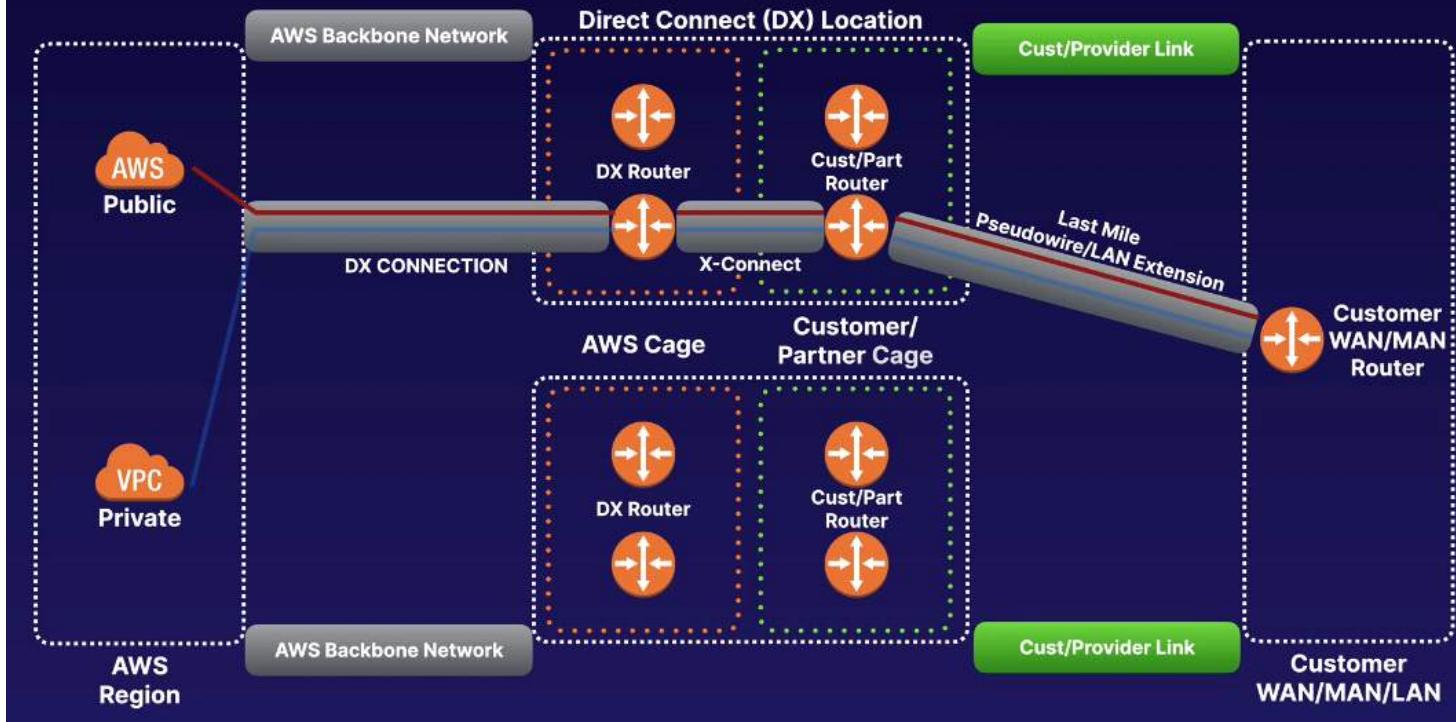
## 2 Types of Direct Connect Connection



**Dedicated Connection:** A physical Ethernet connection associated with a single customer. Customers can request a dedicated connection through the AWS Direct Connect console, the CLI, or the API.



**Hosted Connection:** A physical Ethernet connection that an AWS Direct Connect Partner provisions on behalf of a customer. Customers request a hosted connection by contacting a partner in the AWS Direct Connect Partner Program, who provisions the connection.



## VPNs vs. Direct Connect

VPNs allow private communication, but it still traverses the public internet to get the data delivered. While secure, it can be painfully slow.

### DIRECT CONNECT IS:

- ✓ Fast
- ✓ Secure
- ✓ Reliable
- ✓ Able to take massive throughput

# Direct Connect Exam Tips

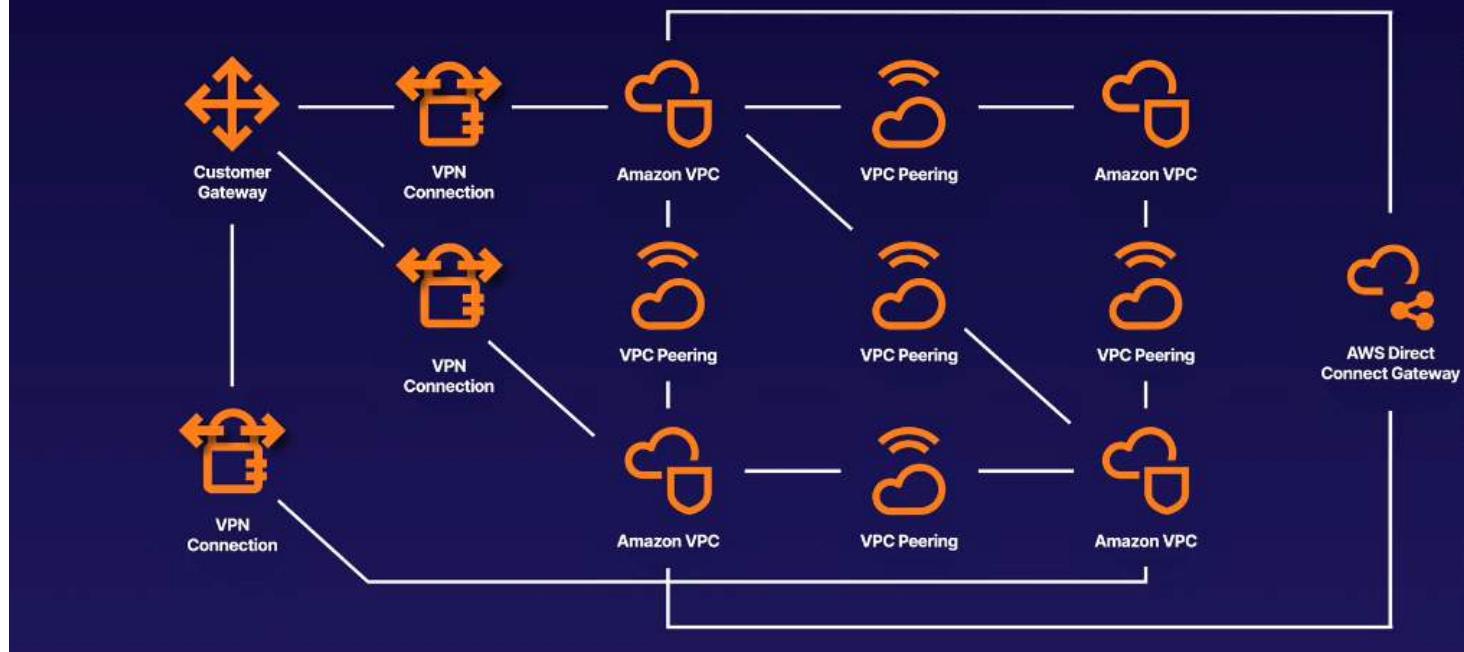
- ✓ Direct Connect directly connects your data center to AWS
- ✓ Useful for high-throughput workloads (e.g., lots of network traffic)
- ✓ Helpful when you need a stable and reliable secure connection

# Transit Gateway - Simplifying Networks with Transit Gateway

Thursday, October 21, 2021 11:54 PM

## Network Architecture Diagram

A CLOUD



## Transit Gateway

AWS Transit Gateway connects VPCs and on-premises networks through a central hub. This simplifies your network and puts an end to complex peering relationships. It acts as a cloud router — each new connection is only made once.



## Transit Gateway Facts

- ✓ Allows you to have transitive peering between thousands of VPCs and on-premises data centers.
- ✓ Works on a hub-and-spoke model.
- ✓ Works on a regional basis, but you can have it across multiple regions.
- ✓ You can use it across multiple AWS accounts using RAM (Resource Access Manager).

## Transit Gateway Exam Tips

- ✓ You can use route tables to limit how VPCs talk to one another.
- ✓ Works with Direct Connect as well as VPN connections.
- ✓ Supports IP multicast (not supported by any other AWS service).

# Jenkinsfile

Monday, 8 May 2023 8:14 PM

```
pipeline {

parameters {
    string(name: 'branchname', defaultValue: 'master', description: 'Branch name')
    string(name: 'committorbranch', defaultValue: 'master', description: 'Enter the commit ID or branchname')
    string(name: 'epmsconfigfile', defaultValue: 'KEA_EPMS_27042022_V1', description: 'Enter the swarm config file for epms')
    string(name: 'epmsadminconfigfile', defaultValue: 'KEA_EPMS_ADMIN_27042022_V1', description: 'Enter the swarm config file for epmsadmin')
    string(name: 'buildcomponents', defaultValue: '1', description: 'which all components you want to build\n 1. Only docker\n 2. only pages\n 3. Both docker and static pages')
    string(name: 'environment', defaultValue: 'uat', description: 'where are you going to deploy? Deployment works only for UAT!!! For Prod Deployment contact Devops Team')
}

environment {
    profile = "epmsDev"
    // branchname= "release"
    // def builddate = sh(script: "echo kea-8.0-", returnStdout: true).trim()
    def builddate = sh(script: "echo kea-8.0-`date +%d%m%Y%H%M%S`", returnStdout: true).trim()
    uiprofile = "prod"
}

tools {
    jdk 'jdk11'
    maven 'maven3'
}

agent any

stages {

stage("SCM Checkout") {
    steps {
        script {
            wrap([$class: 'BuildUser']) {
                sh 'echo ${BUILD_USER}'
                env.user1 = "${BUILD_USER}"
            }
            sh "mvn -version"
            def scmVars = checkout([$class: 'GitSCM', branches: [
                [name: "${params.committorbranch}"]
            ], doGenerateSubmoduleConfigurations: false, extensions: [], submoduleCfg: [], userRemoteConfigs: [
                [credentialsId: 'gitlab', url: 'https://labmirror.kpisoft.com/devops/kea.git']
            ]])
            svnRevision = scmVars.GIT_COMMIT
            shortcommitId = sh(returnStdout: true, script: 'git rev-parse HEAD|cut -c1-7')
            commitId = sh(returnStdout: true, script: 'git rev-parse HEAD')
            env.BUILDVERSION = "${env.builddate}-${svnRevision}-${params.branchname}"
            env.BUILDVERSION1 = "${env.builddate}-${shortcommitId}"
            echo "${env.BUILDVERSION}"
        }
    }
}

stage("Build Skylark Application") {
    when {
        expression {
            params.buildcomponents == "1" || params.buildcomponents == "3"
        }
    }
    steps {
        sh "mvn clean install -DskipTests=true -Dsvn.revision=${env.BUILDVERSION} -PepmsDev"
    }
}
```

```

stage("Build Docker image") {
    when {
        expression {
            params.buildcomponents == "1" || params.buildcomponents == "3"
        }
    }
    steps {
        dir('app-bundle') {
            sh 'cp /data/dockerfiles/Dockerfilekea Dockerfile'
            sh 'cp /data/dockerfiles/secret.sh .'
            sh 'mvn dockerfile:build'
            sh 'cp target/app-bundle.jar /tmp/'
        }
    }
}

stage("push docker image to dockerhub") {
    when {
        expression {
            params.buildcomponents == "1" || params.buildcomponents == "3"
        }
    }
    steps {
        sh "docker tag skylarkkpisoft/app-bundle:8.0-SNAPSHOT mysjentomo/kea:myclouduatepms${env.BUILDVERSION}"
        sh "docker push mysjentomo/kea:myclouduatepms${env.BUILDVERSION}"
    }
}

stage("Version Configuration") {
    steps {
        sh "sed -i 's/{version}/ ${env.BUILDVERSION}'g' kea-ui/webshellV2/src/app/core/core-service/configuration.ts"
        sh "sed -i 's/{version}/ ${env.BUILDVERSION}'g' kea-ui/webadmin/src/app/shared/services/configuration.ts"
        sh "sed -i 's/{version}/ ${env.BUILDVERSION}'g' kea-ui/web-search-adm/src/app/core/core-service/configuration.ts"
    }
}

stage("UI Build") {
    parallel {
        stage("HOME UI Build using NPM") {
            steps {
                script {
                    dir('kea-ui/webshellV2') {
                        if (params.buildcomponents == "2" || params.buildcomponents == "3") {
                            sh 'npm install'
                            sh 'npm rebuild node-saas'
                            sh "npm run build:aot:prod"
                        } else {
                            echo "UI Build is not select in build components"
                        }
                    }
                }
            }
        }
        stage("Building Admin UI using NPM") {
            steps {
                script {
                    dir('kea-ui/webadmin') {
                        if (params.buildcomponents == "2" || params.buildcomponents == "3") {
                            sh 'npm install'
                            sh "npm run build:aot:prod"
                        } else {
                            echo "UI Build is not select in build components"
                        }
                    }
                }
            }
        }
        stage("TracerHub UI Build using NPM") {
            steps {
                script {
                    dir('kea-ui/web-search-adm') {
                        if (params.buildcomponents == "2" || params.buildcomponents == "3") {
                            sh 'npm install'
                            sh 'npm rebuild node-saas'
                        }
                    }
                }
            }
        }
    }

```

```

        sh "npm run build:aot:prod"
    } else {
        echo "TracerHub UI Build is not select in build components"
    }
}
}

stage("Move the static files to the deployment location /data/staticfiles/myclouduat-epms") {
when {
    expression {
        params.buildcomponents == "2" || params.buildcomponents == "3"
    }
}
steps {
    sh 'rm -rf /data/staticfiles/myclouduat-epms/home'
    sh 'rm -rf /data/staticfiles/myclouduat-epms/admin'
    sh 'rm -rf /data/staticfiles/myclouduat-epms/search'
    sh 'mkdir /data/staticfiles/myclouduat-epms/home'
    sh 'mkdir /data/staticfiles/myclouduat-epms/admin'
    sh 'mkdir /data/staticfiles/myclouduat-epms/search'
    sh 'cp -r kea-ui/webshellV2/dist/* /data/staticfiles/myclouduat-epms/home'
    sh 'cp -r kea-ui/webadmin/dist/* /data/staticfiles/myclouduat-epms/admin'
    sh 'cp -r kea-ui/web-search-adm/dist/* /data/staticfiles/myclouduat-epms/search'
    sh "mkdir /data/staticfiles/myclouduat-epms/${env.BUILDVERSION}-${env.uiprofile}"
    sh "cp -r /data/staticfiles/myclouduat-epms/home /data/staticfiles/myclouduat-epms/${env.BUILDVERSION}-${env.uiprofile}"
    sh "cp -r /data/staticfiles/myclouduat-epms/admin /data/staticfiles/myclouduat-epms/${env.BUILDVERSION}-${env.uiprofile}"
    sh "cp -r /data/staticfiles/myclouduat-epms/search /data/staticfiles/myclouduat-epms/${env.BUILDVERSION}-${env.uiprofile}"
}
}

stage("deploy docker image") {
when {
    expression {
        params.buildcomponents == "1" || params.buildcomponents == "3"
    }
}
steps {
    script {
        if (params.environment == "uat") {
            env.tagname = "myclouduatepms${env.BUILDVERSION}"
            sh "ansible-playbook /data/playbooks/myclouduat-onetouch.yml --extra-vars 'tagname=${env.tagname} epmsconfigfile=${params.epmsconfigfile} epmsadminconfigfile=${params.epmsadminconfigfile}'"
        } else {
            echo "contact your heroes <<<<<DevOps Team >>>>>"
        }
    }
}
}

stage("deploy static files to nginx server") {
when {
    expression {
        params.buildcomponents == "2" || params.buildcomponents == "3"
    }
}
steps {
    script {
        if (params.environment == "uat") {
            sh "ansible-playbook /data/playbooks/myclouduat-epmsnginx.yml"
        } else {
            echo "contact your heroes <<<<<DevOps Team >>>>>"
        }
    }
}
post{
    always{

```

```

echo "=====always====="
cleanWs()
}
success{
    echo "=====pipeline execution success====="
    emailext body: """<p>Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]</p><p>Status: Success</p><p>Check it here <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]</a></p>""", subject: "Success: Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]",
    recipientProviders: [developers(), requestor()]
}
failure{
    echo "=====pipeline execution failed====="

    emailext body: """<p>Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]</p><p>Status: Failed</p><p>Check it here <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]</a></p>""", subject: "Failed: Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]",
    recipientProviders: [developers(), requestor()], to: 'cc:mysj.alert@entomo.co'
}
}
}

```

# Pipeline

Saturday, September 11, 2021 4:49 PM

**Pipeline:** is a sequence of events or jobs that can be executed.

That allow Jenkins users to define pipelined job processes with code, stored and versioned in a source repository.

- It is used to build and test your software projects continuously that making it easier for developers to integrate changes to the project and make it easier for users to obtain a fresh build.
- It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.

## Why Plugin pipeline?

- If your application have less JOB to execute then go for build plugin pipeline.
- If your application have mutliple job like development, Code review, Unit Test, Coverage Test, Load Test, Integration Test, Packaging, Build Ets. Then go for Declarative or Else Scripted Pipeline.

To use Pipeline as Code, projects must contain a file named **Jenkinsfile** in the repository root, which contains a “Pipeline script.”

**Stage:** A block that contains a series of steps. A stage block can be named anything; it is used to visualize the pipeline process.

**Step:** A task that says what to do. Steps are defined inside a stage block.

There are 3 types –

1. CI CD pipeline (Continuous Integration Continuous Delivery)
2. Scripted pipeline
3. Declarative pipeline

## Scripted Pipeline:

It is based on Groovy script. That executes the pipeline or any of its stages on any available agent.

1. Which defines the build stage and performs steps related to building stage.
2. If it is defines the test stage then it performs steps related to the test stage.
3. If it is defines the deploy stage then it performs steps related to the deploy stage.

```

Jenkinsfile (Scripted Pipeline)
node { ①
    stage('Build') { ②
        // ③
    }
    stage('Test') { ④
        // ⑤
    }
    stage('Deploy') { ⑥
        // ⑦
    }
}

```

### Declarative Pipeline:

It provides a simple and friendly syntax to define a pipeline.

1. It executes the pipeline or any of its stages on any available agent.
2. Defines the build stage it performs steps related to building stage
3. Defines the test stage it performs steps related to the test stage
4. Defines the deploy stage performs steps related to the deploy stage

```

Jenkinsfile (Declarative Pipeline)
pipeline {
    agent any ①
    stages {
        stage('Build') { ②
            steps {
                // ③
            }
        }
        stage('Test') { ④
            steps {
                // ⑤
            }
        }
        stage('Deploy') { ⑥
            steps {
                // ⑦
            }
        }
    }
}

```

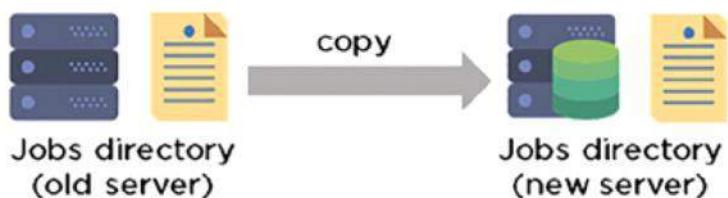
- **Multibranch Pipeline:** build multiple branches of a *single* repository automatically
- **Organization Folders:** scan a **GitHub Organization** or **Bitbucket Team** to discover an organization's repositories, automatically creating managed Multibranch Pipeline jobs for them

# Backup

Tuesday, 14 February 2023 10:36 PM

- Periodically backup your JENKINS\_HOME directory.
  - o Build jobs
  - o Configurations
  - o Slave node configurations
  - o Build history

Copy the jobs directory from the old server to the new one



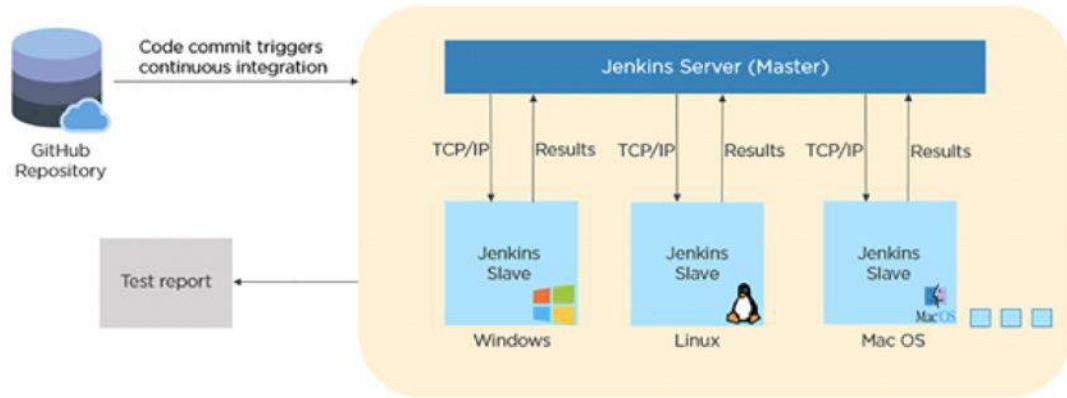
# Master / Slave

Tuesday, 14 February 2023 10:28 PM

Jenkins master pulls the code from remote GitHub repository every time there is a code commit

It distributes the workload to all the jenkins slaves.

On request from the master, slaves carry out, build, test and produce test report



# CMD

Tuesday, 14 February 2023 10:33 PM

```
# java -jar Jenkins.war
```

### What is Continuous Integration In Jenkins?

In software development, multiple developers or teams work on different segments of the same web application. So in this case, you have to perform integration testing by integrating all modules. In order to do that an automated process for each piece of code is performed on a daily bases so that all your codes get tested. This process is known as continuous integration.

1. Dev team have the responsibility to commit the code to Dev-Branch
2. Jenkins will fetch the code from Github and will map with the job enabled for a specific task.
3. We will make sure that CI and CD is done for the job/task.
4. Jenkins will pull the code and will enter the commit phase of the task
5. Jenkins then will compile code and its called **build phase** of the task
6. The code is deployed by Jenkins after the **code is merged to Master branch by DevOps team** and the job is started for a specific application
7. The code is ready to be deployed and enter the deployment phase cycle
8. The code after deployed from Jenkins then get deployed to the server using docker container
9. After the code is working fine in staging server with unit testing, Same code then is deployed on the production server

### How do you achieve continuous integration using Jenkins?

Here are the steps –

- All the developers commit their source code changes to the shared Git repository.
- Jenkins server checks the shared Git repository at specified intervals and detected changes are then taken into the build.
- The build results and test results are shared to the respective developers
- The built application is displayed on a test server like Selenium and automated tests are run.
- The clean and tested build is deployed to the production server.

# Security

Tuesday, 14 February 2023 10:39 PM

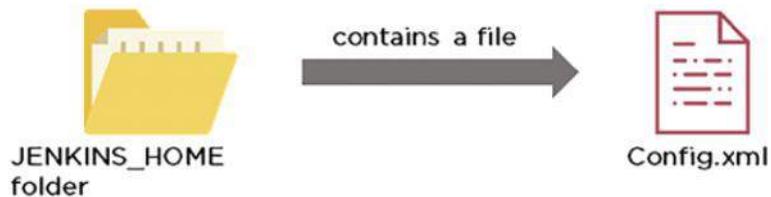
- Jenkins uses an internal database to store user data and credentials.
- It uses LDAP to authenticate users.

## Secure Jenkins?

- Make sure that the global security is on.
- Check if Jenkins is integrated with my company's user directory with an appropriate plugin.
- Ensure that the matrix/Project matrix is enabled to fine-tune access.
- Automate the process of setting rights/privileges in Jenkins with custom version controlled script.
- Limit physical access to Jenkins data/folders.
- Periodically run security audits on the same.

## Forget Password

- ◆ When security is enabled, the Config file contains an XML element named useSecurity that will be set to true.
- ◆ By changing this setting to false, security will be disabled the next time Jenkins is restarted.



## Intermediate Questions

Tuesday, September 7, 2021 1:38 AM

### Clone A Git Repository Via Jenkins?

If you want to clone a Git repository via Jenkins, you have to enter the e-mail and user name for your Jenkins system. Switch into your job directory and execute the “git config” command for that.

### Set up Jenkins job.

Go to Jenkins top page, select “New Job”, then choose “Build a free-style software project”.

Now you can tell the elements of this freestyle job:

- Optional SCM, such as CVS or Subversion where your source code resides.
- Optional triggers to control when Jenkins will perform builds.
- Some sort of build script that performs the build (ant, maven, shell script, batch file, etc.) where the real work happens.
- Optional steps to collect information out of the build, such as archiving the artifacts and/or recording javadoc and test results.
- Optional steps to notify other people/systems with the build result, such as sending e-mails, IMs, updating issue tracker, etc..

### Backup and copy files in Jenkins?

- To create a backup all you need to do is to periodically back up your JENKINS\_HOME directory.
- This contains all of your build jobs configurations, your slave node configurations, and your build history.
- To create a back-up of your Jenkins setup, just copy this directory.
- You can also copy a job directory to clone or replicate a job or rename the directory.

### Custom build of a core plugin?

Below are the steps to deploy a custom build of a core plugin:

- Stop Jenkins.
- Copy the custom HPI to \$Jenkins\_Home/plugins.
- Delete the previously expanded plugin directory.
- Make an empty file called <plugin>.hpi.pinned.
- Start Jenkins.

### Broken build for your project in Jenkins?

- I will open the console output for the broken build and try to see if any file changes were missed.
- If I am unable to find the issue that way, then I will clean and update my local workspace to replicate the problem on my local and try to solve it.

### Build scheduled in Jenkins?

- By source code management commits
- After completion of other builds
- Can be scheduled to run at a specified time (crons)
- Manual Build Requests

### Agent, post-section, Jenkinsfile

**Agent:** It is directive to tell Jenkins to execute the pipeline in a particular manner and order.

**Post-section:** If we have to add some notification and to perform other tasks at the end of a pipeline, post-section will definitely run at the end of every pipeline's execution.

**Jenkinsfile:** The text file where all the definitions of pipelines are defined is called Jenkinsfile. It is being checked in the source control repository.

### Automated tests on Jenkins? How is it done?

Yes, this can be done easily. Automated tests can be run through tools like Selenium or maven. Developers can schedule the test runs. Jenkins displays the test results and sends a report to the developers.

## **The first job was successful, but the second failed. What should you do next?**

You just need to restart the pipeline from the point where it failed by doing ‘restart from stage’.

### **JENKINS HOME directory?**

All the settings, logs and configurations are stored in the JENKINS\_HOME directory.

### **backup plugin? Why is it used?**

This is a helpful plugin that backs up all the critical settings and configurations to be used in the future. This is useful in cases when there is a failure so that we don’t lose the settings.

### **Q31. What is a trigger? Give an example of how the repository is polled when a new commit is detected.**

Triggers are used to define when and how pipelines should be executed.

When Jenkins is integrated with an SCM tool, for example, Git, the repository can be pulled every time there is a commit.

- The Git plugin should be first installed and set up.
- After this, you can build a trigger that specifies when a new build should be started. For example, you can create a job that polls the repository and triggers a build when a change is committed.

### **Q32. How do you define parameters for a build in Jenkins?**

A build can take several input parameters to execute. For example, if you have multiple test suites, but you want to run only one. You can set a parameter so that you are able to decide which one should be run. To have parameters in a job, you need to specify the same while defining the parameter. The parameter can be anything like a string, a file or a custom.

### **Q33. What are the ways to configure Jenkins node agent to communicate with Jenkins master?**

There are 2 ways to start the node agent –

- **Browser** – if Jenkins node agent is launched from a browser, a JNLP (Java Web Start) file is downloaded. This file launches a new process on the client machine to run jobs.
- **Command-line** – to start the node agent using the command line, the client needs the executable agent.jar file. When this file is run, it simply launches a process on the client to communicate with the Jenkins master to run build jobs.

### **Q34. How does Jenkins authenticate users?**

There are 3 ways –

- The default way is to store user data and credentials in an internal database.
- Configure Jenkins to use the authentication mechanism defined by the application server on which it is deployed.
- Configure Jenkins to authenticate against LDAP server.

### **Q35. How can you use a third-party tool in Jenkins?**

Below are the steps used for working with a third-party tool in Jenkins.

- First install the third-party software
- Download the plug-in that supports the third-party tool.
- Configure the third-party tool in the admin console.
- Then use the required plug-in from the Jenkins build job.

For different third-party tools, the procedure may vary slightly, because of the difference in configuration settings.

### **Q36. Q37. What syntax does Jenkins use to schedule build job or SVN polling?**

The cron syntax.

Cron syntax is represented using five asterisks each separated by a space. The syntax is as follows – [minutes] [hours] [day of the month] [month] [day of the week]. Example, if you want to set up a cron for every Monday at 11.59 pm, it would be 59 11 \* \* 1

### **Q38. What is DevOps and in which stage does Jenkins fit in?**

DevOps is a software development practice that blends software development (Dev) with the IT operations (Ops) making the whole development lifecycle simpler and shorter by constantly delivering builds, fixes, updates, and features. Jenkins plays a crucial role because it helps in this integration by automating the build, test and deployment process.

### **Q39. Do you know any other Continuous Integration tools? How is Jenkins better than any of those?**

There are many other CI tools, and the most prominent ones are –

- TeamCity
- Bamboo
- Perforce
- Circle CI
- Go
- ThoughtWorks
- Integrity
- Travis CI

There are many more. We cannot say if Jenkins is better than each because each has its own unique features. For example, TeamCity offers great .NET support but is complex and costly, Travis CI is free just like Jenkins and has good documentation too. Bamboo too offers efficient and faster builds but is not completely free and so on.

#### **Q40. Name a Jenkins environment variable you have used in a shell script or batch file.**

There are numerous environment variables that are available by default in any Jenkins build job. A few commonly used ones include:

- \$JOB\_NAME
- \$NODE\_NAME
- \$WORKSPACE
- \$BUILD\_URL
- \$JOB\_URL

Note that, as new Jenkins plug-ins are configured, more environment variables become available. For example, when the Jenkins Git plug-in is configured, new Jenkins Git environment variables, such as \$GIT\_COMMIT and \$GIT\_URL, become available to be used in scripts.

#### **Q41.Q43. What is a DSL Jenkins?**

The Jenkins “Job DSL / Plugin” is made up of two parts – first, The Domain Specific Language (DSL) itself that allows users to describe jobs using a Groovy-based language, and second, a Jenkins plugin which manages the scripts and the updating of the Jenkins jobs which are created and maintained as a result.

#### **Q44. How do you create Multibranch Pipeline in Jenkins?**

The Multibranch Pipeline project type enables you to implement different Jenkinsfiles for different branches of the same project. In a Multibranch Pipeline project, Jenkins automatically discovers, manages and executes Pipelines for branches that contain a Jenkinsfile in source control.

#### **Q45 What are the types of jobs or projects in Jenkins?**

These are the types of jobs/projects in Jenkins –

- Freestyle project
- Maven project
- Pipeline
- Multibranch pipeline
- External Job
- Multi-configuration project
- Github organization

#### **Q46. What is blue ocean in Jenkins?**

It is a project that was started with the purpose to rethink the user experience of Jenkins, modeling and presenting the process of software delivery by surfacing information that's important to development teams. This is done with as few clicks as possible, while still staying true to the extensibility that is core to Jenkins. While this project is in the alpha stage of development, the intent is that Jenkins users can install Blue Ocean side-by-side with the Jenkins Classic UI via a plugin

## Advanced Questions

Tuesday, September 7, 2021 1:32 AM

### Q47. What is Continuous Testing?

Continuous Testing is the process where you execute automated tests as part of the software delivery pipeline. This is done so that you get the feedback on the business risks associated with software as early as possible. It consists of evolving and extending test automation to address the increased complexity and pace of modern application development and delivery. Continuous Testing means that testing takes place on a continuous basis without any disruption of any kind. In a Continuous DevOps process, a software change is continuously moving from Development to Testing to Deployment. The code undergoes continuous development, delivery, testing and deployment.

### Q48. Explain how you can move or copy Jenkins from one server to another?

I will approach this task by copying the jobs directory from the old server to the new one. There are multiple ways to do that, I have mentioned it below:

You can:

- Move a job from one installation of Jenkins to another by simply copying the corresponding job directory.
- Make a copy of an existing job by making a clone of a job directory by a different name.
- Rename an existing job by renaming a directory. Note that if you change a job name you will need to change any other job that tries to call the renamed job.

### Q49. How do you integrate Git with Jenkins?

These are the steps to integrate [Git](#) with Jenkins –

1. Click on the **Manage Jenkins** button on your Jenkins dashboard:

The screenshot shows the Jenkins dashboard with a black header containing the Jenkins logo and the word "Jenkins". Below the header is a navigation menu with the following items: "New Item", "People", "Build History", "Manage Jenkins" (which is highlighted with a red rectangle), "My Views", "Credentials", and "New View". To the right of the menu is a "Welcome to Jenkins!" message with a teal background and white text that says "Please [create new jobs](#) to get started." At the bottom left of the dashboard is a "Build Queue" button with a minus sign next to it.

2. Click on **Manage Plugins**.



# Jenkins

Jenkins >

## Manage Jenkins



### Configure System

Configure global settings and paths.



### Configure Global Security

Secure Jenkins; define who is allowed to access/use the system.



### Configure Credentials

Configure the credential providers and types



### Global Tool Configuration

Configure tools, their locations and automatic installers.



### Reload Configuration from Disk

Discard all the loaded data in memory and reload everything from file system. Useful when



### Manage Plugins

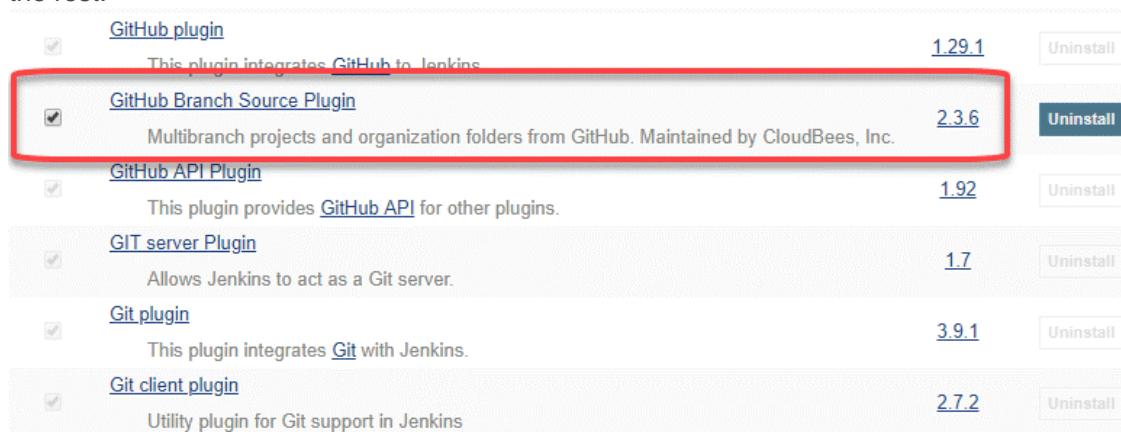
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

There are updates available

### 3. In the Plugins Page

1. Select the GIT Plugin
2. Click on **Install without restart**. The plugin will take a few moments to finish downloading depending on your internet connection, and will be installed automatically.
3. You can also select the option **Download now and Install after restart**.
4. You will now see a “No updates available” message if you already have the Git plugin installed.

### 4. Once you install the plugins , go to **Manage Jenkins** on your Jenkins dashboard. You will see your plugins listed among the rest.



The screenshot shows the Jenkins Manage Plugins page. A red box highlights the "GitHub Branch Source Plugin" entry, which is version 2.3.6 and is maintained by CloudBees, Inc. Other visible plugins include "GitHub API Plugin" (version 1.92), "GIT server Plugin" (version 1.7), "Git plugin" (version 3.9.1), and "Git client plugin" (version 2.7.2). Each plugin entry includes a checkbox, a link to its documentation, its version number, and a "Uninstall" button.

Plugin	Version	Action
GitHub Branch Source Plugin	2.3.6	Uninstall
GitHub API Plugin	1.92	Uninstall
GIT server Plugin	1.7	Uninstall
Git plugin	3.9.1	Uninstall
Git client plugin	2.7.2	Uninstall

### Q50. How can you temporarily turn off Jenkins security if the administrative users have locked themselves out of the admin console?

The JENKINS\_HOME folder contains a file named config.xml. When you enable the security, this file contains an XML element named useSecurity that changes to true. If you change this setting to false, security will be disabled the next time Jenkins is restarted.

```
<useSecurity>false</useSecurity>
```

However, we must understand that disabling security should always be both a last resort and a temporary measure. Once you resolve the authentication issues, make sure that you re-enable Jenkins security and reboot the CI server.

# High Availability Jenkins

Tuesday, 27 December 2022 11:38 PM

# reverse of a number

Friday, 16 December 2022 4:16 AM

```
read -p "Enter a number: " number
temp=$number
while [ $temp -ne 0 ]
do
    reverse=$reverse$((temp%10))
    temp=$((temp/10))
done
echo "Reverse of $number is $reverse"
```

How to get 2nd column?

```
# awk '{print $2}' file.txt
```

Print 1st character from list?

```
# cut -d: -f1 /etc/passwd
```

Show all lines except #?

```
# cat file.txt | grep -v ^#
```

Print 10th line of the file?

```
# cat -n 2  
# head -10 file.txt | tail -1
```

Remove character from word

```
# tr -d "("  
# tr -s " " "
```

# sed

Tuesday, 11 April 2023 12:27 PM

sed -n -e 's/^.\*stalled: //p'

Detailed explanation:

- -n means not to print anything by default.
- -e is followed by a sed command.
- s is the pattern replacement command.
- The regular expression ^.\*stalled: matches the pattern you're looking for, plus any preceding text (. \* meaning any text, with an initial ^ to say that the match begins at the beginning of the line). Note that if stalled: occurs several times on the line, this will match the last occurrence.
- The match, i.e. everything on the line up to stalled: , is replaced by the empty string (i.e. deleted).
- The final p means to print the transformed line.

Flags	Operation	Command	Operation
-e	combine multiple commands	s	substitution
-f	read commands from file	g	global replacement
-h	print help info	p	print
-n	disable print	i	ignore case
-v	print version info	d	delete
-r	use extended regex	G	add newline
		w	write to file
		x	exchange pattern with hold buffer
		h	copy pattern to hold buffer
		;	separate commands

- Delete blank lines from a file

```
sed '/^$/d' hello.sh
```

```
#!/bin/bash
# My First Script
echo "Hello World!"
```

- Delete line n through m in a file

```
sed '2,4d' hello.sh
```

```
#!/bin/bash
echo "Hello World!"
```

- Add flag -e to carry out multiple matches.

```
cat hello.sh | sed -e 's/bash/tcsh/g' -e 's/First/Second/g'
#!/bin/tcsh
# My Second Script
echo "Hello World!"
```

- Alternate form

```
sed 's/bash/tcsh/g; s/First/Second/g' hello.sh

#!/bin/tcsh
# My Second Script
echo "Hello World!"
```

- The default delimiter is slash (/), can be changed

```
sed 's:/bin/bash:/bin/tcsh:g' hello.sh

#!/bin/tcsh
# My First Script
echo "Hello World!"
```

# Awk

Wednesday, 19 April 2023 12:10 AM

```
uptime
11:18am up 14 days 0:40, 5 users, load average: 0.15, 0.11,
0.17

uptime | awk '{print $0}'
11:18am up 14 days 0:40, 5 users, load average: 0.15, 0.11,
0.17

uptime | awk '{print $1,NF}'
11:18am 12

uptime | awk '{print NR}'
1

uptime | awk -F, '{print $1}'
11:18am up 14 days 0:40

for i in $(seq 1 3); do touch file${i}.dat ; done
for i in file* ; do
> prefix=$(echo $i | awk -F. '{print $1}')
> suffix=$(echo $i | awk -F. '{print $NF}')
> echo $prefix $suffix $i; done

file1 dat file1.dat
file2 dat file2.dat
file3 dat file3.dat
```

- Print list of files that are bash script files

```
awk '/^#!/bin/bash/{print $0, FILENAME}' *
→ #!/bin/bash Fun1.sh
    #!/bin/bash fun_pam.sh
    #!/bin/bash hello.sh
    #!/bin/bash parm.sh
```

- Print extra lines below patterns

```
awk '/sh/{print;getline;print}' <hello.sh
#!/bin/bash
```

# Log Parsing

Tuesday, 18 April 2023

1:12 PM

## LOG PARSING CHEAT SHEET

 <b>GREP</b>	GREP allows you to search patterns in files. ZGREP for GZIP files.  \$grep <pattern> file.log	-n: Number of lines that matches -i: Case insensitive -v: Invert matches -E: Extended regex -c: Count number of matches -l: Find filenames that matches the pattern
 <b>NGREP</b>	NGREP is used for analyzing network packets.  \$ngrep -I file.pcap	-d: Specify network interface -i: Case insensitive -x: Print in alternate hexdump -t: Print timestamp -I: Read pcap file
 <b>CUT</b>	The CUT command is used to parse fields from delimited logs.  \$cut -d ":" -f 2 file.log	-d: Use the field delimiter -f: The field numbers -c: Specifies characters position
 <b>SED</b>	SED (Stream Editor) is used to replace strings in a file.  \$sed s/regex/replace/g	S: Search            -e: Execute command g: Replace        -n: Suppress output d: Delete w: Append to file
 <b>SORT</b>	SORT is used to sort a file.  \$sort foo.txt	-o: Output to file            -c: Check if ordered -r: Reverse order        -u: Sort and remove -n: Numerical sort        -f: Ignore case -k: Sort by column        -h: Human sort
 <b>UNIQ</b>	UNIQ is used to extract uniq occurrences.  \$uniq foo.txt	-c: Count the number of duplicates -d: Print duplicates -i: Case insensitive
 <b>DIFF</b>	DIFF is used to display differences in files by comparing line by line.  \$diff foo.log bar.log	How to read output? a: Add            #: Line numbers c: Change        <: File 1 d: Delete        >: File 2
 <b>AWK</b>	AWK is a programming language use to manipulate data.  \$awk {print \$2} foo.log	Print first column with separator ":" \$awk -F ":" {print \$1} /etc/passwd  Extract uniq value from two files: awk 'FNR==NR {a[\$0]++; next} !(\$0 in a)' f1.txt f2.txt

# File Operation

Tuesday, 18 April 2023 11:44 PM

## **Operation bash**

File exists if [ -e test ]

File is a regular file if [ -f test ]

File is a directory if [ -d /home ]

File is not zero size if [ -s test ]

File has read permission if [ -r test ]

File has write permission if [ -w test ]

File has execute permission if [ -x test ]

# Integer Comparisons

Tuesday, 18 April 2023 11:46 PM

**Equal to** if [ 1 –eq 2 ]  
**Not equal to** if [ \$a –ne \$b ]  
**Greater than** if [ \$a –gt \$b ]  
**Greater than or equal to** if [ 1 –ge \$b ]  
**Less than** if [ \$a –lt 2 ]  
**Less than or equal to** if [ \$a –le \$b ]

**Equal to** if [ \$a == \$b ]  
**Not equal to** if [ \$a != \$b ]  
**Zero length or null** if [ -z \$a ]  
**Non zero length** if [ -n \$a ]

# Logical Operation

Tuesday, 18 April 2023 11:48 PM

Operation	Example
! (NOT)	<code>if [ ! -e test ]</code>
&& (AND)	<code>if [ -f test] &amp;&amp; [ -s test ]</code> <code>if [[ -f test &amp;&amp; -s test ]]</code> <code>if ( -e test &amp;&amp; ! -z test )</code>
(OR)	<code>if [ -f test1 ]    [ -f test2 ]</code> <code>if [[ -f test1    -f test2 ]]</code>

# For Loop

Tuesday, 18 April 2023 11:49 PM

Exmaple1:  
for arg in `seq 1 4`  
do  
echo \$arg  
touch test.\$arg  
done

How to delete test files using a loop?  
rm test.[1-4]

Example 2:  
for file in `ls /home/\$USER`  
do cat \$file  
done

# While loop

Tuesday, 18 April 2023 11:51 PM

- The `while` construct test for a condition at the top of a loop and keeps going as long as that condition is true.
- In contrast to a `for` loop, a `while` is used when loop repetitions is not known beforehand.

```
read counter
while [ $counter -ge 0 ]
do let counter--
echo $counter
done
```

# Until loop

Tuesday, 18 April 2023 11:53 PM

The `until` construct test a condition at the top of a loop, and stops looping when the condition is met (opposite of `while` loop)

```
read counter
until [ $counter -lt 0 ]
do let counter--
echo $counter
done
```

# Regex

Wednesday, 19 April 2023 12:03 AM

## One slide about Regular Expression

- What are Regular Expressions (regex)?
  - They describe patterns in strings
  - These patterns can be used to modify strings
  - Invented by Stephen Cole Kleene
  - Idea of RegEx dates back to the 1950s
- Today, they come indifferent “flavors”
- PCRE, POSIX Basic & Extended RegEx, ECMA RegEx and loads more!
- Examples:

## Regex examples

- Anchors - `^` and `$`
  - `^The` matches any string that starts with The
  - `end$` matches a string that ends with end
  - `^The end$` exact string match (starts and ends with The end)
  - `roar` matches any string that has the text roar in it
- Quantifiers - `*` `+` `?` and `{}`
  - `abc*` matches a string that has ab followed by zero or more c
  - `abc+` matches a string that has ab followed by one or more c
  - `abc?` matches a string that has ab followed by zero or one c
  - `abc{2}` matches a string that has ab followed by 2 c
  - `abc{2,}` matches a string that has ab followed by 2 or more c
  - `abc{2,5}` matches a string that has ab followed by 2 up to 5 c
- OR operator - `|` `or` `[]`
  - `a(b|c)` matches a string that has a followed by b or c
  - `a[bc]` same as previous

# grep & egrep

Wednesday, 19 April 2023 12:05 AM

- **grep:** Unix utility that searches a pattern through either information piped to it or files.
- **egrep:** extended grep, same as `grep -E`
- **zgrep:** compressed files.
- Usage: `grep <options> <search pattern> <files>`
- Options:
  - `-i` ignore case during search
  - `-r, -R` search recursively
  - `-v` invert match i.e. match everything except *pattern*
  - `-l` list files that match *pattern*
  - `-L` list files that do not match *pattern*
  - `-n` prefix each line of output with the line number within its input file.
  - `-A num` print *num* lines of trailing context after matching lines.
  - `-B num` print *num* lines of leading context before matching lines.

- Search files containing the word `bash` in current directory

```
grep bash *
```

- Search files NOT containing the word `bash` in current directory

```
grep -v bash *
```

- Repeat above search using a case insensitive pattern match and print line number that matches the search pattern

```
grep -in bash *
```

- Search files not matching certain name pattern

```
ls | grep -vi fun
```

---

100	Thomas	Manager	Sales	\$5,000
200	Jason	Developer	Technology	\$5,500
300	Raj	Sysadmin	Technology	\$7,000
500	Randy	Manager	Sales	\$6,000

---

- grep OR

```
grep 'Man\|Sales' employee.txt
-> 100 Thomas Manager Sales $5,000
    300 Raj Sysadmin Technology $7,000
    500 Randy Manager Sales $6,000
```

```
500 Randy Manager Sales $6,000
```

- grep AND

```
grep -i 'sys.*Tech' employee.txt  
-> 100300 Raj Sysadmin Technology $7,000
```

# Corosync

Sunday, 13 November 2022 7:07 PM

Important component of pacemaker  
Handling communication between cluster nodes  
Also check the cluster membership and quorum data

# Quorum

Sunday, 13 November 2022 7:07 PM

It is required to maintain the cluster integrity.

If cluster lose the quorum then cluster will stop or terminate resources and resources group to maintain the data integrity.

It can be defined as voting system which is required to maintain the cluster integrity.

# Fencing

Sunday, 13 November 2022 7:08 PM

Fencing is a technique or method to power off or terminate the faulty node from the cluster. Fencing is very important component of a cluster, Red Hat Cluster will not start resource and service recovery for non responsive node until that node has been fenced.

# PCS

Sunday, 13 November 2022 7:08 PM

**pcs** is command line utility, used to configure and manage cluster nodes. In other terms we can say pcs mange every aspect of Pacemaker cluster.

# Cluster commands

Sunday, 13 November 2022 7:08 PM

## **Cluster status**

```
~]# pcs cluster status  
~]# pcs status
```

## **Automatic startup**

```
~]# pcs cluster enable --all
```

## **Standby**

```
~]# pcs cluster standby {Cluster_Node_Name}
```

## **Resume**

```
~]# pcs cluster unstandby {Cluster_Node_Name}
```

## **Quorum status**

```
~]# corosync-quorumtool
```

## **Configure fencing**

```
~]# pcs stonith create name fencing_agent parameters
```

## **View Fencing**

```
~]# pcs stonith show --full
```

## **Fence node**

```
~]# pcs stonith fence nodeb.example.com
```

## **Useful info of cluster resource**

```
~]# pcs resource describe {resource_name}
```

### **Example:**

```
~]# pcs resource describe Filesystem
```

**To Display the list of all the resources of a cluster, use the beneath command,**  
~]# pcs resource list

## **Create resource in cluster**

```
~]# pcs resource create {resource_name} {resource_provider} {resource_parameters} --group  
{group_name}
```

## **Filesystem resource**

```
~]# pcs resource create my_fs Filesystem device=/dev/sdb1 directory=/var/www/html fstype=xfs --group  
my_group
```

## **Clear the fail count**

```
~]# pcs resource failcount show
```

**To clear or reset the failcount**

```
~]# pcs resource failcount reset {resource_name} {cluster_node_name}
```

# Move cluster resource to another

Sunday, 13 November 2022 7:30 PM

Ans: Cluster resources and resource groups can be moved away from the cluster node using the below command,

```
~]# pcs resource move {resource_or_resources_group} {cluster_node_name}
```

When a cluster resource or resources group moved away from a cluster node then a temporary constraint rule is enabled on the cluster for that node , means that resource / resources group can't be run that cluster node, so to remove that constraint use the following command,

```
~]# pcs resource clear {resource_or_resource_group} {cluster_node_name}
```

# Logs

Sunday, 13 November 2022 7:31 PM

Default log file for pacemaker is “/var/log/pacemaker.log” and for corosync is “/var/log/messages”

# Storage based fencing

Sunday, 13 November 2022 7:10 PM

storage based fence device will cut off the faulty cluster node from storage access, it will not power off or terminate the cluster node.

Let's assume shared storage like "/dev/sda" is assigned to all the cluster node, then you create the storage based fencing device using the below command,

```
~]# pcs stonith create {Name_Of_Fence_Device} fence_scsi devices=/dev/sda meta provides=unfencing
```

Use the following command to fence any cluster node for fence testing,

```
~]# pcs stonith fence {Cluster_Node_Name}
```

# HA-LVM

Sunday, 13 November 2022 7:11 PM

- **HA-LVM** (Volume Group and its logical volumes can be accessed only one node at a time, can be used with traditional file systems ext4 and xfs)
- **Clustered LVM** (It is commonly used while working with shared file system like GFS2)

Let's assume shared storage is provisioned on all the cluster nodes,

- a) On any of the cluster node, do pvcreate, vgcreate and lvcreate on shared storage disk
- b) Format the logical volume on storage disk
- c) on each cluster node, enable HA-LVM tagging in file “**/etc/lvm/lvm.conf**”  
locking\_type = 1

Also define logical volume group that are not shared in the cluster,

Volume\_list = [rootvg,logvg]

rootvg & logvg are OS volume group and not shared among the cluster nodes.

- d) On each cluster node, rebuild initramfs using the following command,  
~]# dracut -H -f /boot/initramfs-\$(uname -r).img \$(uname -r) ; reboot
- e) Once all the cluster nodes are rebooted, verify the cluster status using “pcs status” command,
- f) On any of cluster node , create LVM resource using below command,  
~]# pcs resource create ha\_lvm LVM volumegroup=cluster\_vg exclusive=true --group halvm\_fs
- g) Now create FileSystem resource from any of the cluster node,

```
~]# pcs resource create xfs_fs Filesystem device="/dev/{volume-grp}/{logical_volume}" directory="/mnt" fstype="xfs" --group halvm_fs
```

# Constraints

Sunday, 13 November 2022 7:31 PM

Constraints can be defined as restrictions or rules which determine in which orders cluster resources will start and stopped. Constraints are classified into three types,

- Order Constraints – It decides the orders how resources or resource group will be started or stopped.
- Location constraints – It decides on which nodes resources or resource group may run
- Colocation constraints- It decides whether two resources or resource group may run on the same node.

## Multi-Tier Architect

Wednesday, September 22, 2021 5:56 PM

**CloudFront** - It's **edge caching**, to cache content close to end users for faster delivery.

It pulls static content from an S3 bucket

It pulls dynamic content from an Application Load Balancer in front of the web instances.

The **Application Load Balancer** and a **Bastion host** for administration are **deployed in Public subnet**

**Internet Gateway** - allows communication between resources in your VPC and the internet.

**NAT Gateways** - is in each Availability Zone that **enable EC2 instances in private subnets** (App and Data) to access the internet.

Within the VPC there exist **two types of subnets**: public (Public Subnet) and private (App Subnet and Data Subnet).

**Resources deployed** into the **public subnets** will receive a **public IP address** and will be publicly **visible on the internet**.

Resources deployed into the **private subnets** receive only a **private IP address** and are **not publicly visible on the internet**, Which improving the security of those resources.

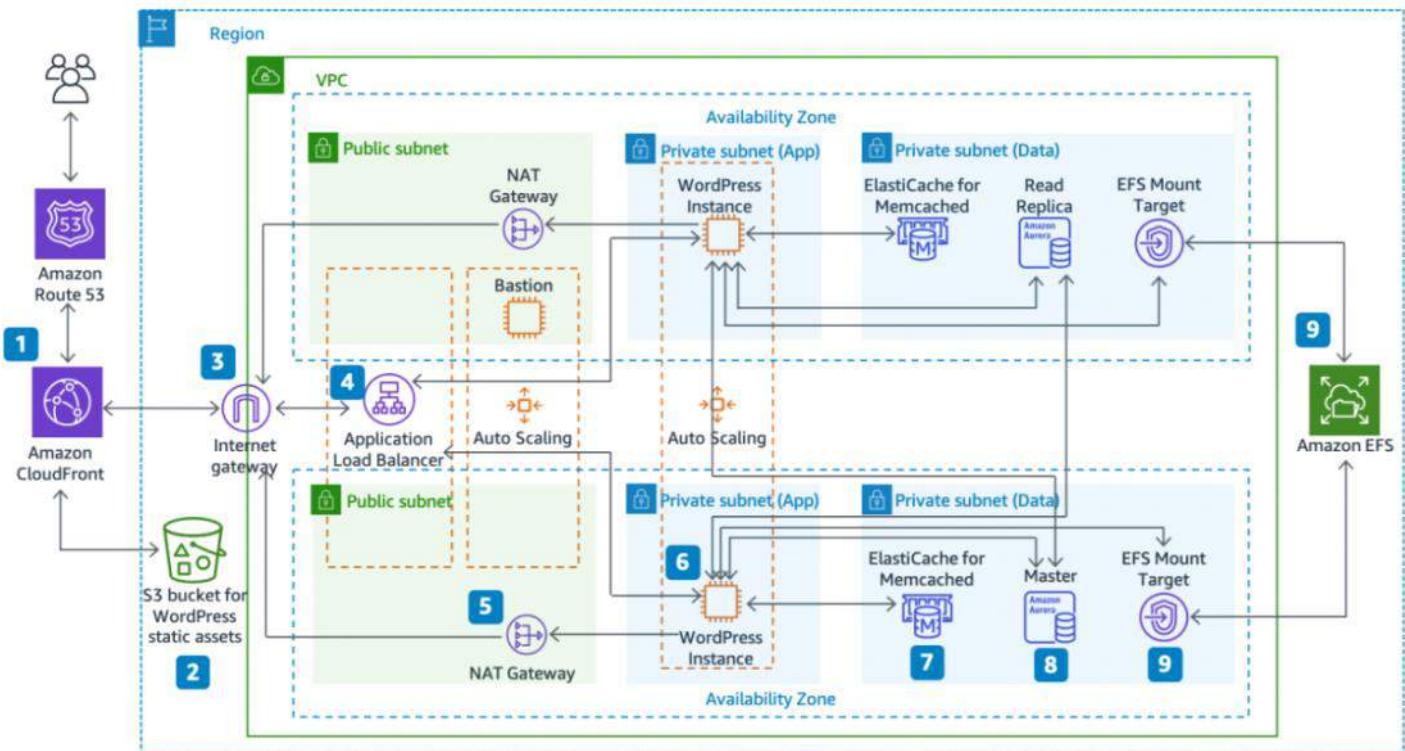
The **web app instances** run in an **Auto Scaling group** of Amazon EC2 instances

**MemCached** - An ElastiCache cluster, which caches often queried data to speed up responses.

**Aurora MySQL instance** - hosts the application database.

**Amazon EFS** - The EC2 instances access **shared application data** on an EFS file system via an EFS Mount Target in each Availability Zone.

The web server instances , ElastiCache cluster , Aurora MySQL db instances , and EFS Mount Targets are all deployed in private subnets.



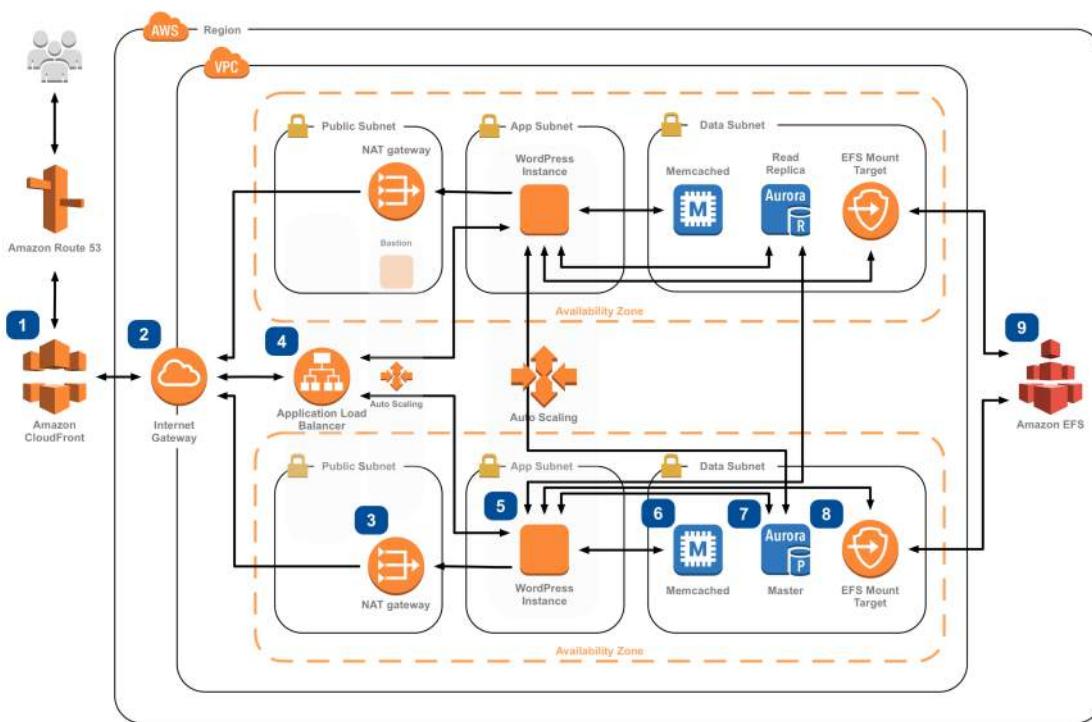
Static content is any file that is stored in a server and is the same every time it is delivered to users. HTML files and images contents.

Dynamic content is content that changes based on factors specific to the user such as time of visit, location, and device.

# WordPress Hosting

## How to run WordPress on AWS

WordPress is one of the world's most popular web publishing platforms, being used to publish 27% of all websites, from personal blogs to some of the biggest news sites. This reference architecture simplifies the complexity of deploying a scalable and highly available WordPress site on AWS.



- 1 Static and dynamic content is delivered by **Amazon CloudFront**.
- 2 An **Internet gateway** allows communication between instances in your VPC and the Internet.
- 3 **NAT gateways** in each public subnet enable Amazon EC2 instances in private subnets (App & Data) to access the Internet.
- 4 Use an **Application Load Balancer** to distribute web traffic across an Auto Scaling Group of Amazon EC2 instances.
- 5 Run your WordPress site using an **Auto Scaling group of Amazon EC2 instances**. Install the latest versions of WordPress, Apache web server, PHP 7, and OPCache and build an Amazon Machine Image that will be used by the Auto Scaling group launch configuration to launch new instances in the Auto Scaling group.
- 6 If database access patterns are read-heavy, consider using a WordPress plugin that takes advantage of a caching layer like **Amazon ElastiCache (Memcached)** in front of the database layer to cache frequently accessed data.
- 7 Simplify your database administration by running your database layer in **Amazon RDS** using either Aurora or MySQL.
- 8 Amazon EC2 instances access shared WordPress data in an Amazon EFS file system using **Mount Targets** in each AZ in your VPC.
- 9 Use **Amazon EFS**, a simple, highly available, and scalable network file system so WordPress instances have access to your shared, unstructured WordPress data, like php files, config, themes, plugins, etc.



## AWS Reference Architectures

© 2017, Amazon Web Services, Inc. or its affiliates. All rights reserved.

**1. Setup the Virtual Private Cloud (VPC):** VPC stands for Virtual Private Cloud (VPC). It is a virtual network where you create and manage your AWS resource in a more secure and scalable manner. Go to the VPC section of the AWS services, and click on the **Create VPC** button.

**2. Setup the Internet Gateway:** The Internet Gateway allows communication between the EC2 instances in the VPC and the internet. To create the Internet Gateway, navigate to the **Internet Gateways** page and then click on **Create internet gateway** button.

**3. Create 4 Subnets:** The subnet is a way for us to group our resources within the VPC with their IP range. A subnet can be public or private. EC2 instances within a public subnet have public IPs and can directly access the internet while those in the private subnet does not have public IPs and can only access the internet through a **NAT gateway**.

For our setup, we shall be creating the following subnets with the corresponding IP ranges.

- demo-public-subnet-1 | CIDR (10.0.1.0/24) | Availability Zone (us-east-1a)
- demo-public-subnet-2 | CIDR (10.0.2.0/24) | Availability Zone (us-east-1b)
- demo-private-subnet-3 | CIDR (10.0.3.0/24) | Availability Zone (us-east-1a)
- demo-private-subnet-4 | CIDR(10.0.4.0/24) | Availability Zone (us-east-1b)

**4. Create Two Route Tables:** Route tables is a set of rule that determines how data moves within our network. We need two route tables; private route table and public route table. The public route table will define which subnets that will have direct access to the internet ( ie public subnets) while the private route table will define which subnet goes through the NAT gateway (ie private subnet).

To create route tables, navigate over to the **Route Tables** page and click on **Create route table** button.

The public and the private subnet needs to be associated with the public and the private route table respectively.

To do that, we select the route table and then choose the **Subnet Association** tab.

**5. Create the NAT Gateway:** The NAT gateway enables the EC2 instances in the **private subnet to access the internet**. The NAT Gateway is an AWS managed service for the NAT instance. To create the NAT gateway, navigate to the **NAT Gateways** page, and then click on the **Create NAT Gateway**.

**6. Create Elastic Load Balancer:** From our architecture, our frontend tier can only accept traffic from the elastic load balancer which connects directly with the internet gateway while our backend tier will receive traffic through the internal load balancer. The essence of the load balancer is to distribute load across the EC2 instances serving that application. If however, the application is using sessions, then the application needs to be rewritten such that sessions can be stored in either the Elastic Cache or the DynamoDB. To create the two load balancers needed in our architecture, we navigate to the **Load Balancer** page and click on **Create Load**

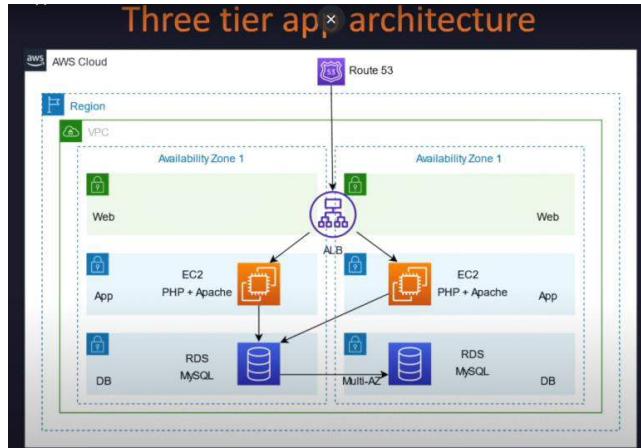
Balancer.

**7. Auto Scaling Group:** We can simply create like two EC2 instances and directly attach these EC2 instances to our load balancer. The problem with that is that our application will no longer scale to accommodate traffic or shrink when there is no traffic to save cost. With Auto Scaling Group, we can achieve this feat. Auto Scaling Group is can automatically adjust the size of the EC2 instances serving the application based on need. This is what makes it a good approach rather than directly attaching the EC2 instances to the load balancer.

## 3 Tier Architecture

Saturday, October 9, 2021 1:02 AM

- Create VPC - make it secure, private env
- Create corresponding subnets (private for App and DB) (Public for Web)
- Create route tables & do subnet associations (helping you traffic being routed through your VPC)
- Create Internet GW (we need to establish the connectivity) & attached to VPC - allows communication between resources in your VPC and the internet.
- Create NAT GW in public subnet: is in each Availability Zone that enable EC2 instances in private subnets (App and Data) to access the internet.
- Make changes in route table so that connectivity to the internet will be established from your internet gateway for the public subnet & NAT gw for private subnets
- Create jump server & App server



**Private Network - VPC** - To make it secure, Private isolate environment

**Web Server (VM) - EC2 / EBS** - using web server hosting on ec2 with attached EBS - front end stuff is taken care by web server

**IP** - User can access from outside

**App Server (VM) - EC2 / EBS** - business logic, suppose it's any social web application, connecting with different people, making new connection etc

**Relational Database - RDS** - You want to extend is then required some kind of Database, like MySQL or oracle whatever you prefer.

Considering your app is doing good, there is more traction from the users and your webserver and app server becomes a bottleneck. Not able to handle the load

**Scaling - Auto Scaling for EC2** - vertical (Increase the capacity of machine components) or horizontal ( adding more machines). If there's load increasing on these ec2 they can scale horizontally automatically.

Now multiple servers and IPs are there,

**Load Balancer - ELB** - you need Intelligent entity that distribute the load to the servers that's load balancing. Which can distributes the incoming traffic to multiple backend EC2 machines

**DNS - Route53** - Map your DNS domain name to load balancer IP

Data is growing, Connection growing so RDS cannot really serve this kind of data storage.

**NoSQL Database (Mongo DB) / DynamoDB** - you need scalable databases and also for connection information and all it makes sense to rather going to NoSQL databases. Some part stored in RDS

Your RDS could be a performance bottleneck. Maybe there is read heavy operations happening on RDS, for that typically you will bring in one more component

**DB Cache - ElasticCache** - (redis and mem chached engines) where you query frequently accessed data so that your application servers don't hit the DB and all request serve from DB Cache

Getting more data like million of picture or videos daily. EBS on Ec2 are not really capable of extending on the fly and have size limitation, so this media never stored typically on these web servers

**External Storage - S3** - unlimited kind of storage, Not a block storage but file storage. You can go on dumping the data and it is accessible over the internet

**Content Filter - Rekognition** - filter the content that are objectionable from the media files

**Click Stream Analysis - Kinesis** - Continue watching what activity you are doing, like what product or what post you are liking, based on that it gives you suggestions.

**Storage for Click Stream - S3** - what data captured will be stored to external storage

**Spark/Hadoop - EMR** - to do some data operations like aggregation, sort the data and find meaning out of that data.

**Data Processing - Glue** - ETL transactions from your dynamoDB tables like maybe you want to do what all friends are there, friend's friends, What activity doing. So year end you want all this data to be extracted & converted into different format data cataloging then further do some data processing using EMR so you need this glue service.

**Data Warehousing - RedShift** - To do data analytics in the end of the year.

**Business Intelligence tool** - Quicksight or Athena - to query the data and generate the report

**Video Convert - Lambda** - Users accessing media files using Mobile so you need converter. Serverless service, just write that code specify how to convert a video. Execution happening into your S3 so new video comes lambda gets triggered, it will convert your video.

**Mobile - External Storage S3**

**Content Delivery Network (Cache) - CloudFront / Edge Location** - Caches these videos and pictures to the nearest caching devices from where the user is accessing data - low latency, better experience  
CloudFront stores or caches your data in edge locations

Notification Service - When users send some request to add connection or message or Email

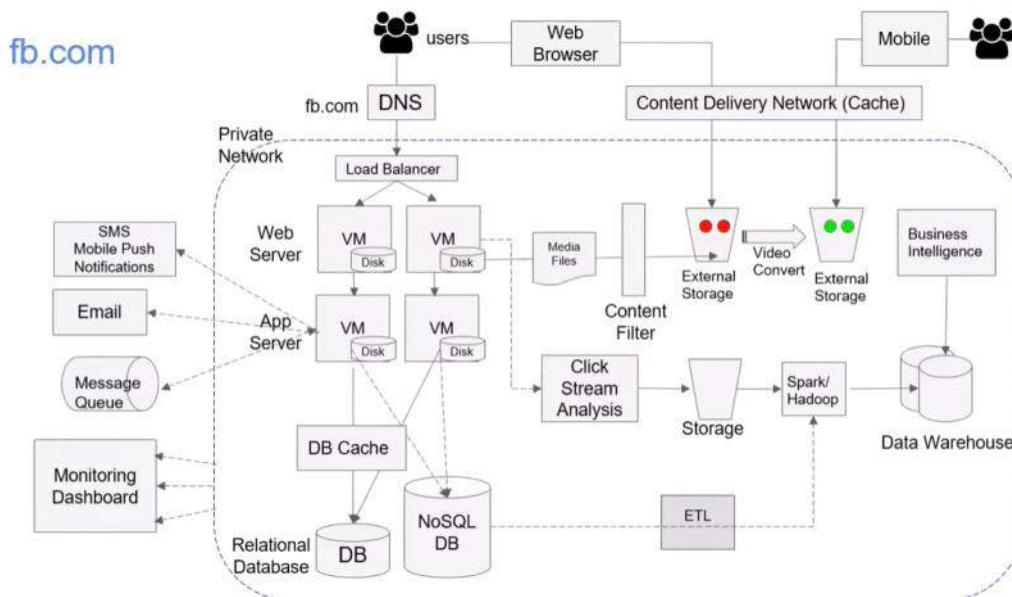
SMS Mobile push notification - SNS (Simple Notification Service)

Email - SES (Simple Email Service)

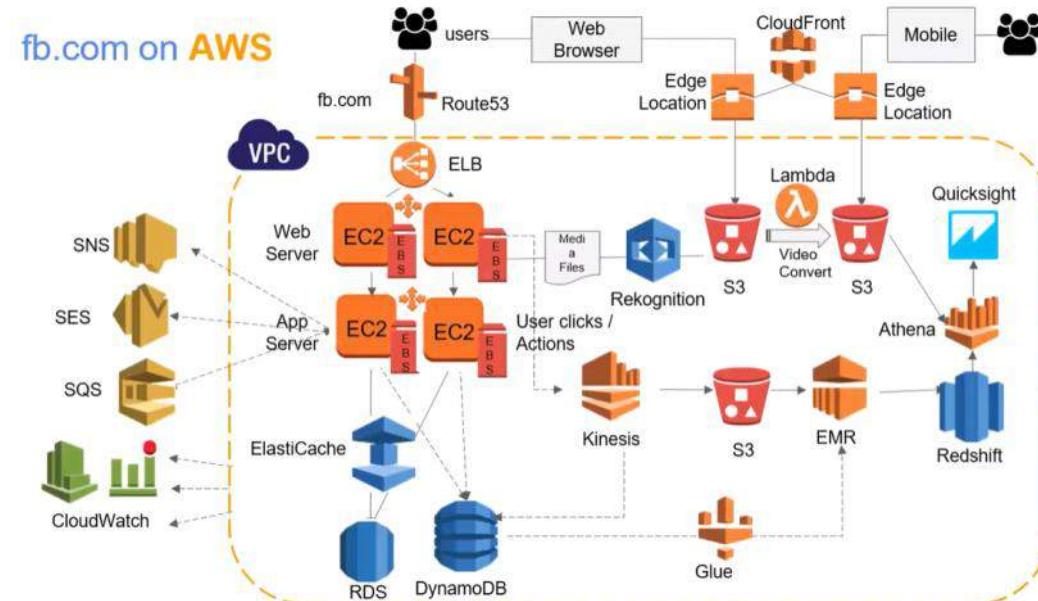
Message Queue - SQS (Simple Queue Service)

Monitoring Dashboard - CloudWatch - Monitoring continuously, Set alarm, Alerts, Take some action, Do some autoscaling

## On-Premises



## On AWS



## AWS Application Services

**REST API - API Gateway** - It exposes all their services through API Calls so that different third-party application can integrate with these applications for that they need REST API services  
In Amazon you can have managed API gateways where it takes care of scaling, throttling, everything so you can just write a code for your APIs,

**Cognito** - Also mobile usage increasing most of your web users, you need to manage their identities like when you develop an application your users must sign up your application. That means you need to manage your user pools, their access and everything.

## AWS Application Services



### Security Service

**IAM - Identity Access Management** - managing all access in your AWS users, What access they have, what service they can use, so all access and authentication and authorization is managed using amazon's IAM.

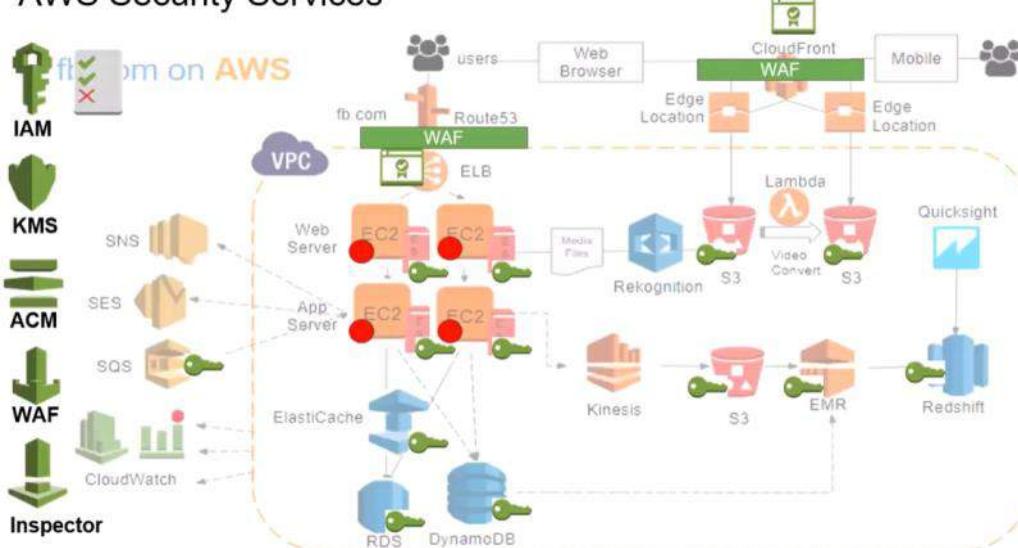
**KMS - Key Management Service** - Encrypt Data which in various storage locations like EBS, S3, EMR, RedShift etc. it manages all the encryption keys for you. You don't need to have your own secure location where you can store your keys and do the encryption.

**ACM - Amazon certificate manager** - Application will be accessed over HTTPS which is SSL enabled connection because if users are doing transaction and they don't want to lose that communication and for this you need digital certificates. That you deploy on Load balancer or CloudFront so that your communication is secure.

**WAF - Web application firewall** - that take care of any attacks. It can prevent like cross-site scripting sql injection, DDoS attacks. WAF can protect your application from that Deploy on CloudFront, Load Balancer or API Gateway.

**AWS Inspector** - Machines need to be patched properly, Free from Vulnerabilities or CVE. Put agent inside your machine and it scan your machine for any Vulnerabilities and gives you report.

## AWS Security Services



### AWS Development and DevOps Services (IaaS)

**CloudFormation** - JSON YAML Template and create your infrastructure from scratch, maybe within 30min. Template will be written by DevOps people and same time you will have Developers and QA

**CodeCommit** - Everyone required some kind of code repository like GIT so AWS have CodeCommit Service.

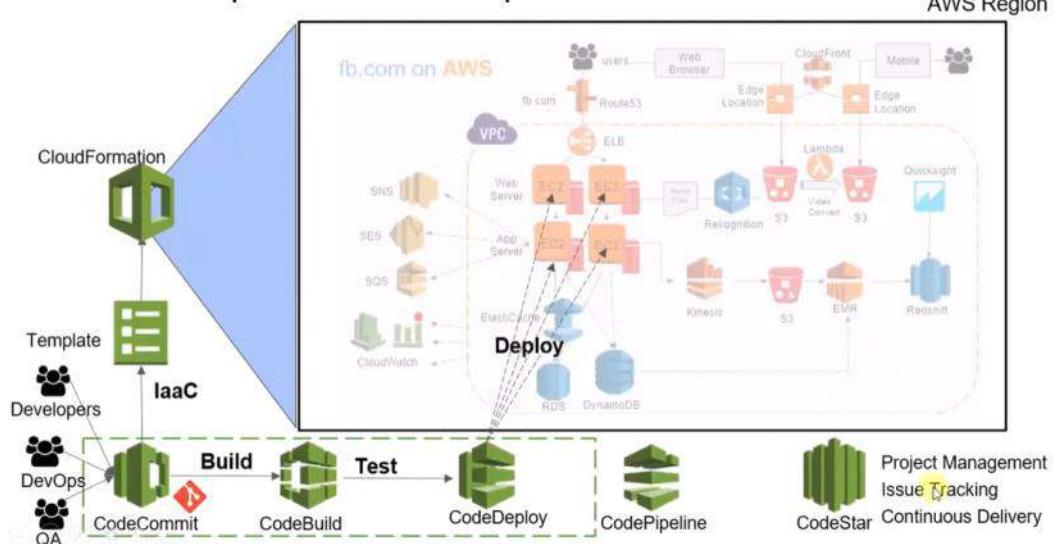
**CodeBuild** - Take the source code and build artifacts. Artifacts are like your exec or binaries, actual your application is executable basically.

**CodeDeploy** - Whatever is produced you have to deploy it. Put your application on EC2 where your app is running, you required deployment

**CI/CD** - This is your pipeline. Automatically deployment

**CodeStar** - Project Management tool like JIRA, Issue Tracking, Continuous Delivery, integrate all these things with project management tool

AWS Development and DevOps Services



# Modules IaaS | PaaS | SaaS

Wednesday, October 13, 2021 11:55 PM

## List different types of cloud services

- Software as a Service (SaaS),
- Data as a Service (DaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS).

### **Infrastructure as a Service (IaaS).**

- Vendor provide you Infra only (AWS/Azure/GCP)
- Compute, Storage, Network, Security
- AWS responsibility - Availability, Server Up and Running, Proper Backup
- Advantage - full control
- Disadvantage - Application running inside infra is not supported

### **Platform as a Service (PaaS)**

- It will provide you platform that is already provisioned.
- I have application but I need Database
- RDS - Relational Database System
- You will get Database platform in very easily without the requirement of DB admin
- We cannot customize it.
- Managed by AWS

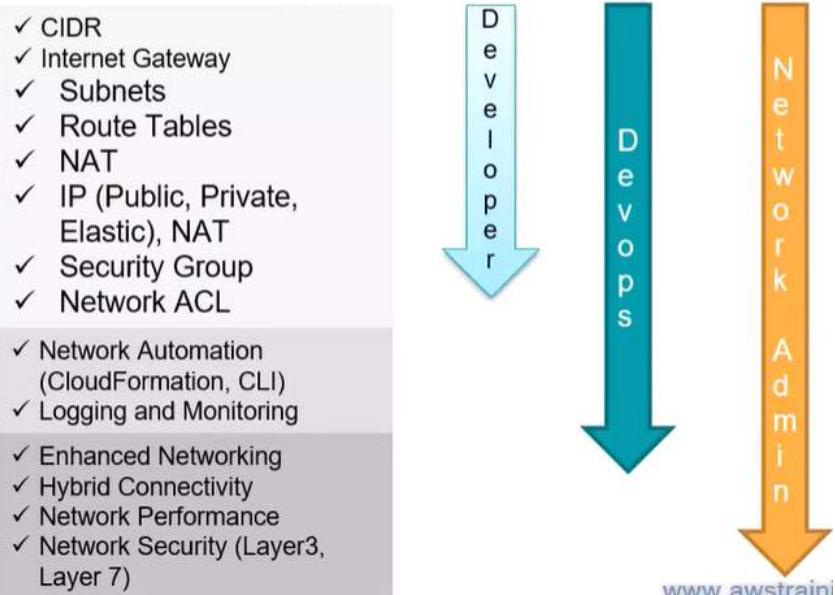
### **Software as a Service (SaaS) - Market Place**

- Office 365, SalesForce CRM, Google App
- All office product, Service Now
- Can't really customized, Limitations are there.

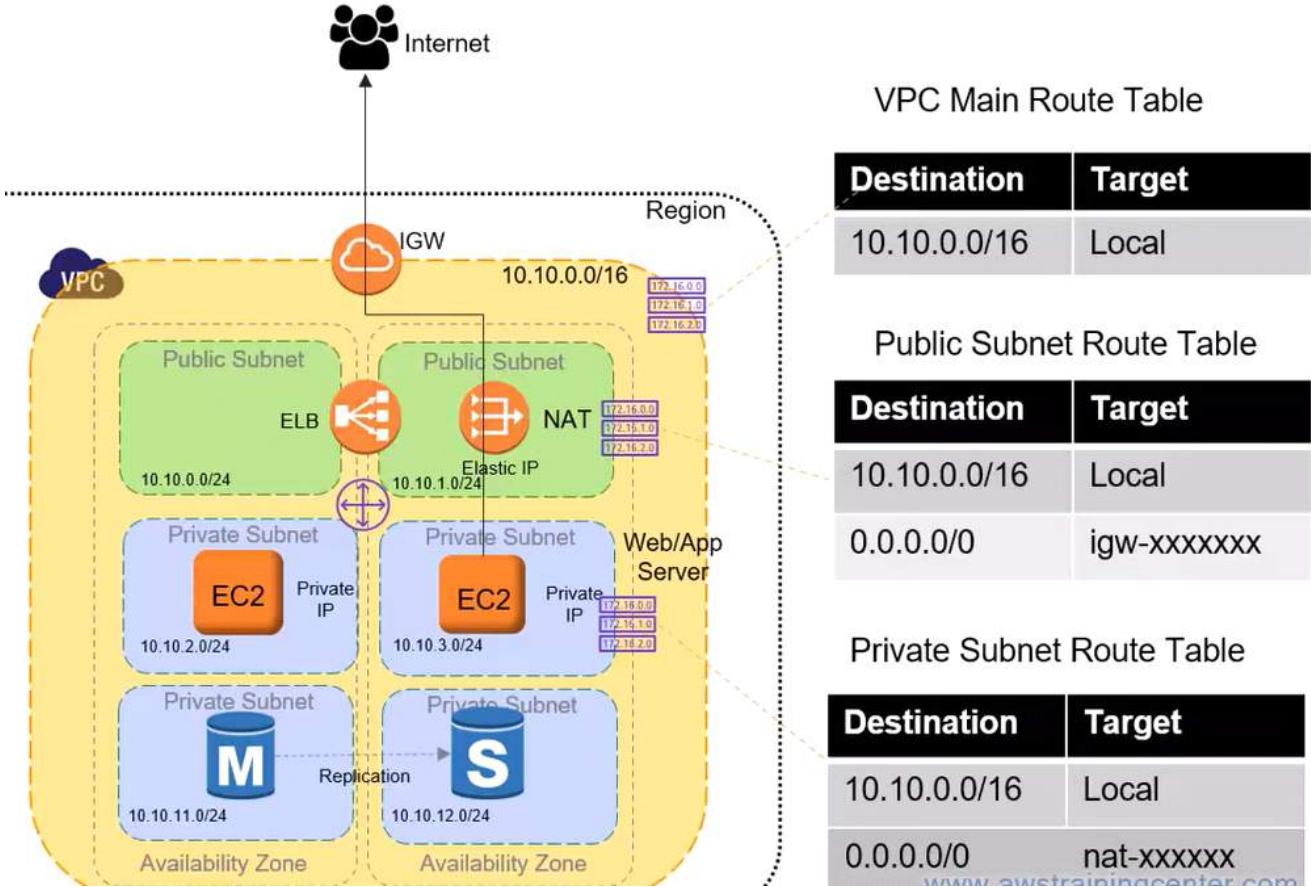
# Networking - AWS

Sunday, October 17, 2021 8:59 PM

- [VPC Basics](#)
- [ELB/CloudFront](#)
- [Route 53 \(DNS\)](#)
- [VPC Endpoint](#)
- [Private Link](#)
- [VPC Peering](#)
- [Transit VPC](#)
- [Transit Gateway](#)
- [Site to Site VPN](#)
- [Client VPN](#)
- [Direct Connect](#)
- [VPC Advanced](#)



- Most of the services run in **VPC private cloud**, Isolate private network, you control what traffic flows in and out of this network
- AWS region: Any region you can choose to create your VPC, Multiple AZ available
- VPC scope is region level that means, you create VPC you can leverage any of the availability zones to create your EC2
- VPC comes with private IP address range which is called **CIDR (Classless inter-domain routing)**
- When you create **VPC it comes with default local router**. Route the traffic within VPC
- **Creating subnet - Private or Public**
- In order to **communicate with talk to internet** from these subnets, **there's IGW. Attached to the VPC**



# Hybrid connectivity

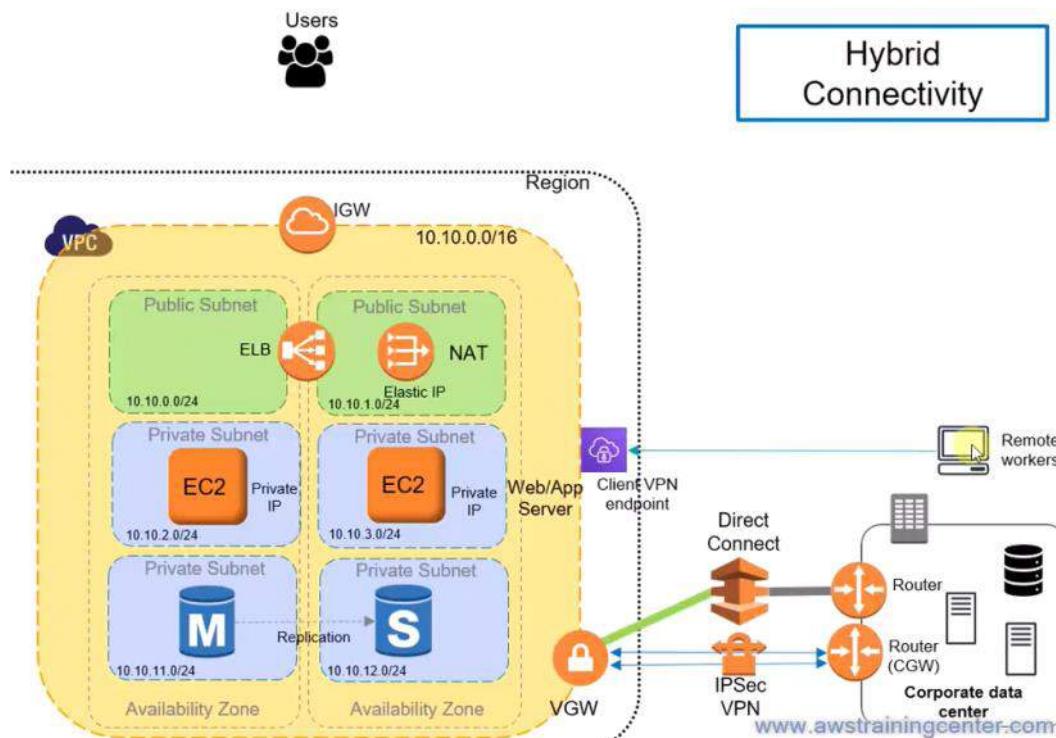
Tuesday, October 19, 2021 3:30 PM

## IPSec VPN Tunnel

- Connect Multiple branches of your companies together using **site to site VPN**
- **VGW - Virtual Private Gateway** : on AWS side
- **CGW Router - Customer Gateway** side
- **IPSec VPC tunnel between VGW and CGW**
- For HA AWS support two tunnel
- Traffic still flows over internet
- It make sure all the traffic is encrypted but it's not reliability

## AWS Direct Connect

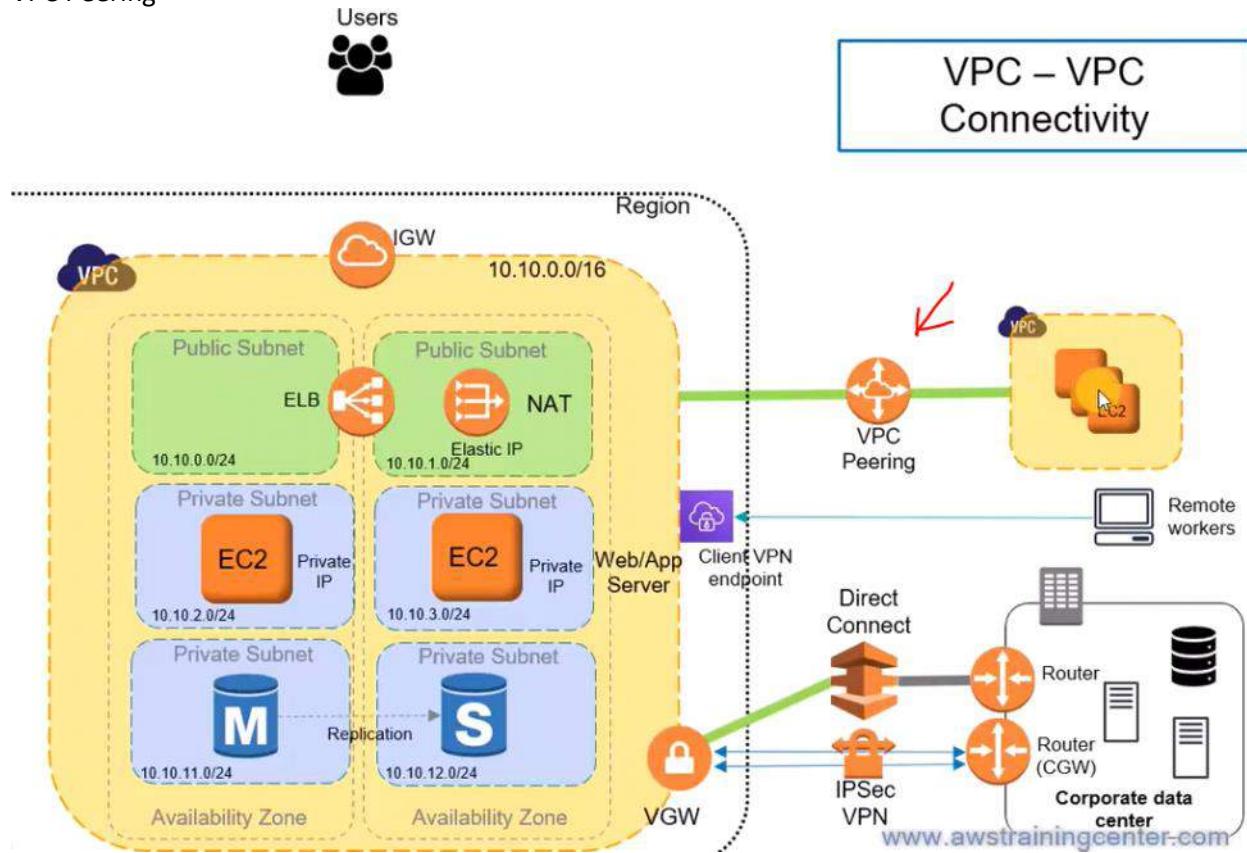
- AWS has different region and the **country in which AWS has region** there will be **multiple direct connect locations**.
- **Not owned by AWS** but owned by **Korean neutral entities** like in india there are 6 or 7 location.
- **Direct connect location already connected to AWS** corresponding region data center.
- It is connected with very **high speed low latency optical fibers more than 100Gbps link**.
- Client need to **connect on-premise network to Direct connect location** in order to get the bandwidth
- You can connect by your own or have direct connect partner who can connect it
- You have 1 Gbps to 10Gbps bandwidth connection to AWS from data center so a lot of big companies for sure goes for Direct Connect



# VPC - VPC Connectivity

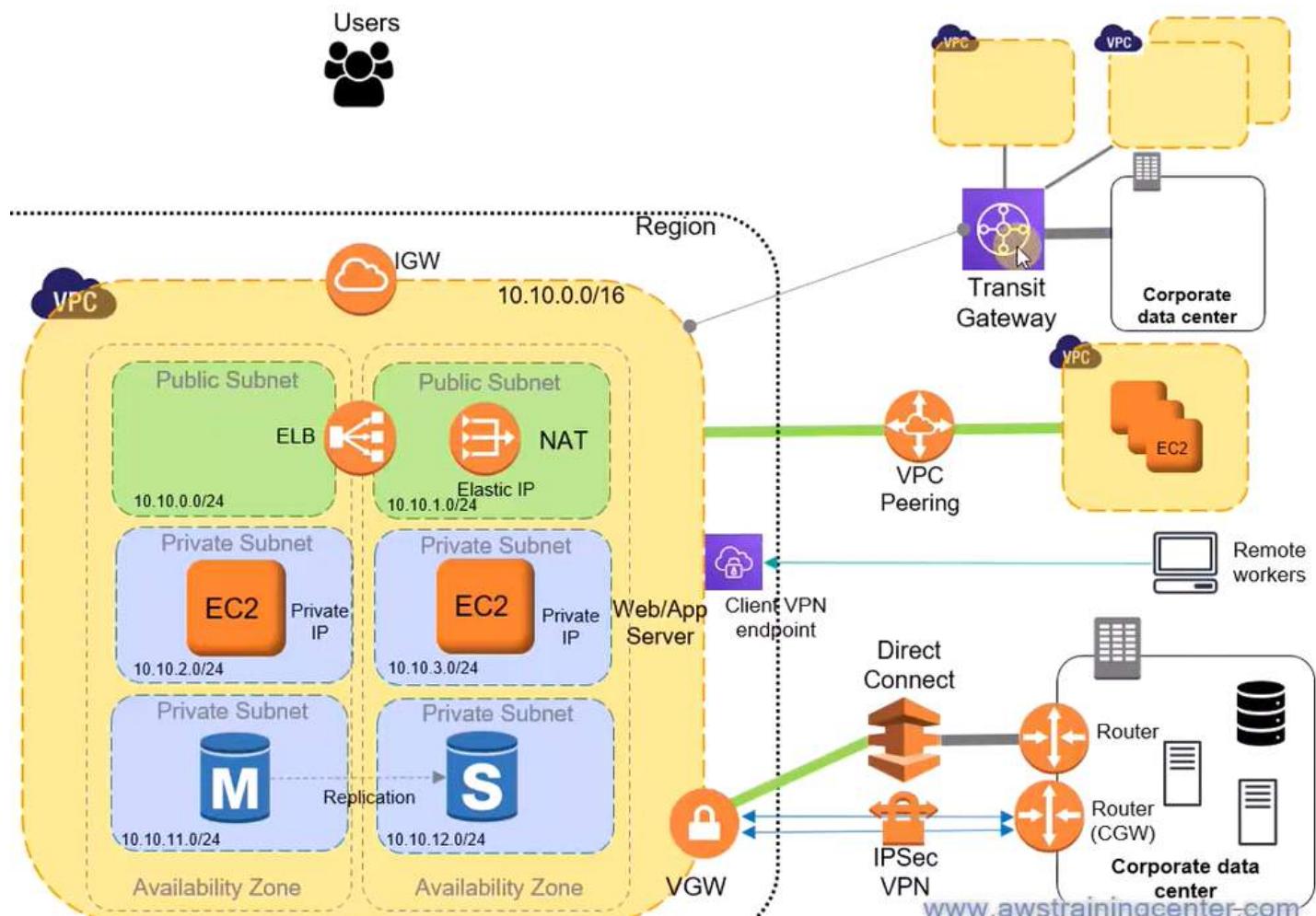
Tuesday, October 19, 2021 3:52 PM

## VPC Peering



# Transit Gateway

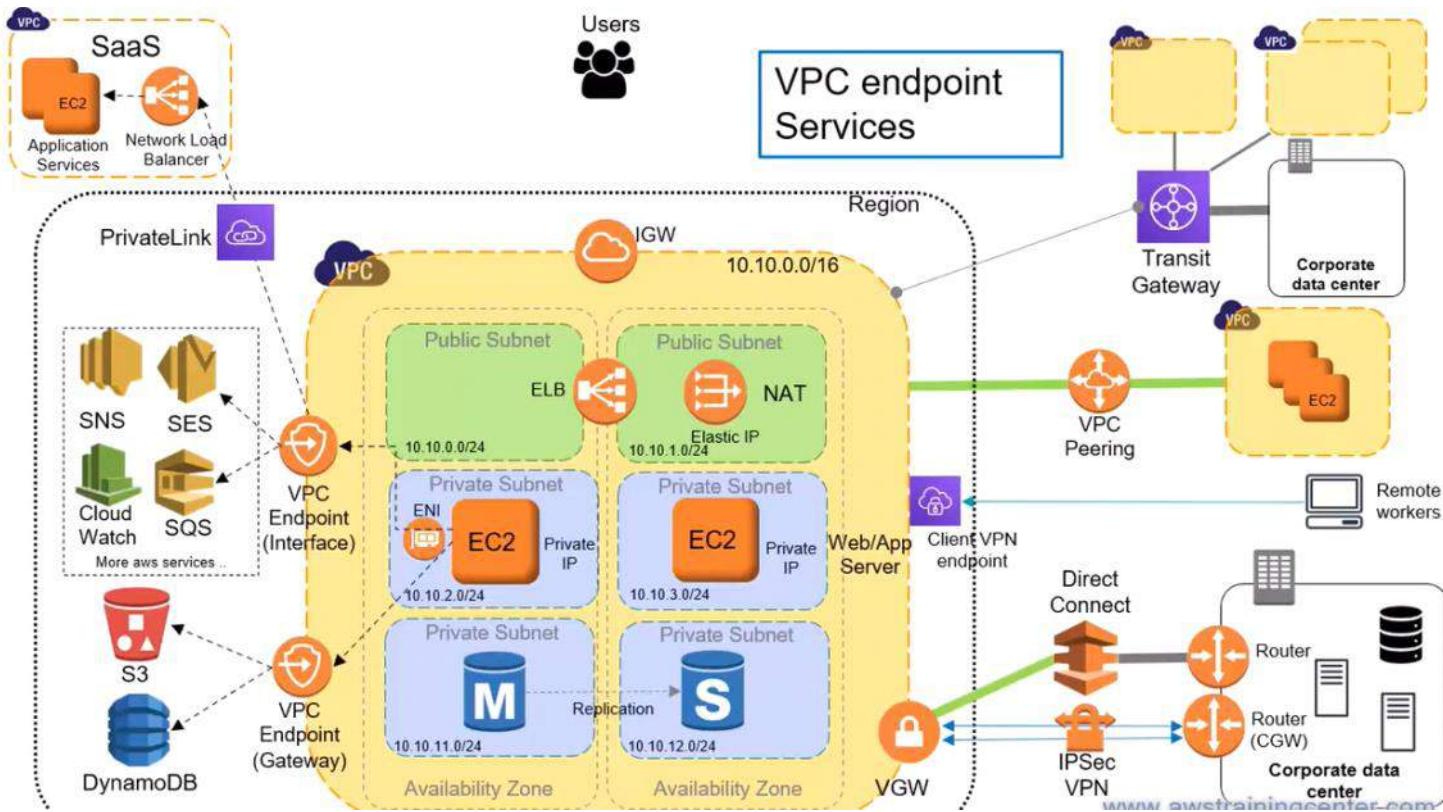
Tuesday, October 19, 2021 3:55 PM



# VPC Endpoint Service

Tuesday, October 19, 2021 3:57 PM

VPC Endpoint Gateway / VPC Endpoint Interface



## 1) Explain what AWS is?

AWS stands for Amazon Web Service; it is a collection of remote computing services also known as a cloud computing platform. also known as IaaS or Infrastructure as a Service.

# ELB - Load Balancing

Tuesday, September 7, 2021 1:56 AM

## ### Elastic Load Balancing (ELB) ###

Load balancing improves the distribution of workloads across multiple computing resources, such as computers, a computer cluster, network links, central processing units, or disk drives.

- It can help to inherently highly-available & fault-tolerant.
- Span region, use every AZ
- Provides DNS endpoint
- Do Not rely on IP address

## Elastic Load Balancing

### 42. What are the different types of load balancers in AWS?

There are three types of load balancers that are supported by Elastic Load Balancing:

1. Application Load Balancer
2. Network Load Balancer
3. Classic Load Balancer

### 43. What are the different uses of the various load balancers in AWS Elastic Load Balancing?

#### *Application Load Balancer*

Used if you need flexible application management and TLS termination.

#### *Network Load Balancer*

Used if you require extreme performance and static IPs for your applications.

#### *Classic Load Balancer*

Used if your application is built within the EC2 Classic network

## 26) What are the advantages of auto-scaling?

Following are the advantages of autoscaling

- Offers fault tolerance
- Better availability
- Better cost management

ELB comes in two different flavors, it call ELB scheme

Functionally identical

Internet-facing : Public IP

Best for public available services

Internal: Private IP

Best for private services

Listeners and SSL Certificates for ELB

In order to receive traffic on ELB we have to enable the listeners

Listen: HTTP:80

Listen: HTTPS:443

If enabling HTTPS or SSL traffic then we have to specify in SSL certificate

The load balancer does support offline SSL certificate uploading so we can remove the burden of encryption from our backend

machine.

You can upload the certificate using CLI tool to IAM services. Right now it's not possible to upload SSL using AWS console.

Port forwarding in the backend of machine because we don't have to SSL on EC2 instances.

Load Balancing Algorithms

Elastic Load Balancer (you can call Node)

Each node gets public or private IP

When request comes from internet or any service then it's distribute the request via DNS round robin

Between the node and backend instance we are going to use least outstanding request as way to determine which backend node should handle the particular request.

Two types of LB

Application LB

Classic LB

#### **40) What is Amazon EMR?**

EMR is a survived cluster stage which helps you to interpret the working of data structures before the intimation. Apache Hadoop and Apache Spark on the Amazon Web Services helps you to investigate a large amount of data. You can prepare data for the analytics goals and marketing intellect workloads using Apache Hive and using other relevant open source designs.

#### **14) Explain how the buffer is used in Amazon web services?**

The buffer is used to make the system more robust to manage traffic or load by synchronizing different component. Usually, components receive and process the requests in an unbalanced way. With the help of buffer, the components will be balanced and will work at the same speed to provide faster services.

# EC2

Tuesday, September 7, 2021 1:52 AM

## ### Elastic Compute Cloud (EC2) ###

We get VM, called them instance. They are Based on Xen hypervisor  
it comes with price fixed menu (Various Combinatinon of CPU, Memory, disk, Network IO) no individualy  
You can launch 1 to 1000 instance  
Consider them disposable

## Virtualization

ParaVirtual & Hvm

### 13) Mention what the security best practices for Amazon EC2 are?

- Use AWS identity and access management to control access to your AWS resources
- Restrict access by allowing only trusted hosts or networks to access ports on your instance
- Review the rules in your security groups regularly
- Only open up permissions that you require
- Disable password-based login, for example, launched from your AMI

### 16) What are key-pairs in AWS?

Key-pairs are secure login information for your virtual machines. To connect to the instances, you can use key-pairs which contain a public-key and private-key.

## Launching an image on EC2

T2 instances are designed to provide moderate baseline performance and the capability to burst to higher performance as required by workload.

t2 - for developer Free

m - good balance of CPU + RAM

C - CPU more and half the family

R Series - more Memory (memory optimize)

D - Large Storage memo

### 17) What are the different types of instances?

Following are the types of instances:

- General purpose
- Computer Optimized
- Memory Optimized
- Storage Optimized
- Accelerated Computing

### 15) While connecting to your instance what are the possible connection issues one might face?

- Connection timed out
- User key not recognized by the server
- Host key not found, permission denied
- An unprotected private key file

- Server refused our key or No supported authentication method available
- Error using MindTerm on Safari Browser
- Error using Mac OS X RDP Client

**10) Explain can you vertically scale an Amazon instance? How?**

Yes, you can vertically scale on Amazon instance. For that

# Elastic Beanstalk

Thursday, September 23, 2021 10:55 AM

## This service for deploying and scaling web applications and services

Applications deployed in the cloud need memory, computing power and an OS to run

AWS Elastic beanstalk can take a lot of the setup work out of development/deployment and can save developers and companies time and hassle.

Elastic Beanstalk will setup an "environment" for you that can contain a number of EC2 instances, an optional database, as well as a few other AWS components such as a Elastic Load Balancer, Auto-Scaling Group, Security Group.

# Auto Scaling

Monday, September 13, 2021 1:15 PM

## ### Auto Scaling ###

Automated approach to increase or decrease the compute, memory or networking resources they have allocated

- It allows us to handle changes in traffic or domain over application.
- It ensures that you have the correct number of Amazon EC2 instances available to handle the load for your application.
- You create collections of EC2 instances, called Auto Scaling groups.
- You can specify the minimum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes below this size.
- You can specify the maximum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes above this size.
- If you specify the desired capacity, either when you create the group or at any time thereafter,
- Auto Scaling ensures that your group has this many instances. If you specify scaling policies, then Auto Scaling can launch or terminate instances as demand on your application increases or decreases.

## Two Components: Launch Configurations and Auto Scaling Groups.

- **Launch Configurations** hold the instructions for the creation of new instances.
- **Scaling Groups**, on the other hand, manage the scaling rules and logic, which are defined in policies.

### Triggers Auto Scaling

The Auto Scaling group in your Elastic Beanstalk environment uses two Amazon CloudWatch alarms to trigger scaling operations. The default triggers scale when the average outbound network traffic from each instance is higher than 6 MB or lower than 2 MB over a period of five minutes.

### Demand-based Scaling Out

When you configure dynamic scaling, you must define how you want to scale in response to changing demand. For example, say you have a web application that currently runs on two instances and you do not want the CPU utilization of the Auto Scaling group to exceed 70 percent. You can configure your Auto Scaling group to scale automatically to meet this need. The policy type determines how the scaling action is performed.

Simple scaling—Increase or decrease the current capacity of the group based on a single scaling adjustment.

Step scaling—Increase or decrease the current capacity of the group based on a set of scaling adjustments, known as step adjustments, that vary based on the size of the alarm breach.

Target tracking scaling—Increase or decrease the current capacity of the group based on a target value for a specific metric. This is similar to the way that your thermostat maintains the temperature of your home – you select a temperature and the thermostat does the rest.

# Lambda

Thursday, September 23, 2021 10:55 AM

AWS Lambda is a serverless compute service that runs your code in response to events and automatically manages the underlying compute resources for you.

# LightSail

Thursday, September 23, 2021 10:55 AM

A Lightsail instance is **a virtual private server (VPS) that lives in the AWS Cloud**. ... Your instances can connect to each other and to other AWS resources through both public (Internet) and private (VPC) networking. You can create, manage, and connect easily to instances right from the **Lightsail console**.

Lightsail has seven virtual server sizes;  
Lightsail tops out at eight cores and 32 GB of memory;  
Lightsail is not endless options.

EC2 has more than 250.  
EC2 instances can get to 128 cores and 3,900 gibibytes (GiB) of memory. But, again, the point of

# IAM

Tuesday, September 7, 2021 1:54 AM

AWS Identity and Access Management (IAM) enables **you to manage access to AWS services and resources securely.**

Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

## 21) What are the roles?

Roles are used to providing permissions to entities which you can trust within your AWS account. Roles are very similar to users. However, with roles, you do not require to create any username and password to work with the resources.

## S3

Tuesday, September 7, 2021 1:51 AM

### ### Simple Storage Service (S3) ###

You can use S3 interface to store and retrieve any amount of data, at any time and from anywhere on the web.

its object storage - Storing the data on object not on FS, its key value store

When we store then S3 Cluster spans entire region, it stores our data in numerous **copy to other region as well**. Durable to loss of 2 AZs, but data is still there. No FS, what we have is bucket and object

Transfer all done via http/s

Different than block storage - not using IOPS

Also used to host static website

No limit to bucket size

Objects up to 5TB

Upload limit 5GB per

Multipart upload

Server side encryption or client side

### 20) Explain default storage class in S3

The default storage class is a Standard frequently accessed.

### 7) How can you send a request to Amazon S3?

Amazon S3 is a REST service, and you can send a request by using the REST API or the AWS SDK wrapper libraries that wrap the underlying Amazon S3 REST API.

#### REST stands for representational state transfer

An API, or application programming interface, is a set of rules that define how applications or devices can connect to and communicate with each other.

### 8) Mention what the difference between Amazon S3 and EC2 is?

The difference between EC2 and Amazon S3 is that

EC2	S3
• It is a cloud web service used for hosting your application	• It is a data storage system where any amount of data can be stored
• It is like a huge computer machine which can run either Linux or Windows and can handle application like PHP, Python, Apache or any databases	• It has a REST interface and uses secure HMAC-SHA1 authentication keys

### 9) How many buckets can you create in AWS by default?

By default, you can create up to 100 buckets in each of your AWS accounts.

### 24) Explain snowball

Snowball is a data transport option. It uses source appliances to move a large amount of data into and out of AWS. With the help of snowball, you can transfer a massive amount of data from one place to another. It helps you to reduce networking costs.

### 52) What are the storage classes available in Amazon S3?

Storage classes available with Amazon S3 are:

- Amazon S3 Standard
- Amazon S3 Standard-Infrequent Access
- Amazon S3 Reduced Redundancy Storage
- Amazon Glacier

# VPC

Tuesday, September 7, 2021 1:52 AM

## ### Virtual Private Cloud (VPC) ###

A logically isolated section of the AWS cloud where you can launch AWS resources in a virtual network that you define". It allows you to have your IP address range, internet gateways, subnet and security groups.

### 19) How many Elastic IPs is allows you to create by AWS?

5 VPC Elastic IP addresses are allowed for each AWS account.

### 27) What is meant by subnet?

A large section of IP Address divided into chunks is known as subnets.

A subnet is a range of IP addresses in your VPC

### 42) Do you need an internet gateway to use peering connections?

Yes, the Internet gateway is needed to use VPC (virtual private cloud peering) connections.

Application generally need:

IP Addresses - need ip range

Segmentation - internal or external

Public

Private

Firewalls - ports allow for app or DB or block

AWS Direct Connect makes it easy to establish a dedicated network connection from your premises to AWS. Using AWS Direct Connect, you can establish private connectivity between AWS and your datacenter, office, or colocation environment, which in many cases can reduce your network costs,

Creating VPC - Networking Session

Routing

Is method where we can control the flow of internet and security in VPC.

Routing and Subnet through the internet.

### Network Access Control Lists (NACL) - Firewall utility for entire subnets

These are like firewall rules but there are applied to subnet as whole Stateless

Public, Private, and Hybrid subnets

Must specify ingress and egress

Allow or Deny

Specify:

Protocol

Source IP range

Destination port range

Security Groups:

Applied to instance

State full

Can specify ingress or egress

Denied by default

Specify

Protocol

source IP range

Destination port range

Creating Security Groups:

### VPC Peering

Connection between two VPCs

Helps Segregate networks

No Bottleneck or SPOF (single point of failure)

Can peer between accounts

IP range must not overlap  
always one to one  
no transitive peering  
no edge to edge routing  
it's very stable device  
Peering connection (PCX) is resource in/of itself

**12) In VPC with private and public subnets, database servers should ideally be launched into which subnet?**  
With private and public subnets in VPC, database servers should ideally launch into private subnets.

**28) Can you establish a Peering connection to a VPC in a different region?**  
No, it's only possible between VPCs in the same region.

**30) How many subnets can you have per VPC?**  
You can have 200 subnets per VPC.

AWS Route 53 takes its name with **reference to Port 53**, which handles DNS for both the TCP and UDP traffic requests

Amazon Route 53 effectively **connects user requests to infrastructure running in AWS** – such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets – and can also be used to route users to infrastructure outside of AWS.

**Amazon's Route 53 provides three services:**

**Record creation** (which registers the human-readable names you'd like associated with your web domains),

**Request handling** (to direct web traffic to the right servers)

**Health checks** (to ensure that traffic isn't being directed to servers that can't handle the load)

## AWS Route 53

### 51. What is the difference between Latency Based Routing and Geo DNS?

The Geo Based DNS routing takes decisions based on the geographic location of the request. Whereas, the Latency Based Routing utilizes latency measurements between networks and AWS data centers. Latency Based Routing is used when you want to give your customers the lowest latency possible. On the other hand, Geo Based routing is used when you want to direct the customer to different websites based on the country or region they are browsing from.

### 52. What is the difference between a Domain and a Hosted Zone?

**Domain**

A domain is a collection of data describing a self-contained administrative and technical unit. For example, [www.simplilearn.com](http://www.simplilearn.com) is a domain and a general DNS concept.

**Hosted zone**

A hosted zone is a container that holds information about how you want to route traffic on the internet for a specific domain. For example, [lms.simplilearn.com](http://lms.simplilearn.com) is a hosted zone.

### 53. How does Amazon Route 53 provide high availability and low latency?

Here's how Amazon Route 53 provides the resources in question:

**Globally Distributed Servers**

Amazon is a global service and consequently has DNS services globally. Any customer creating a query from any part of the world gets to reach a DNS server local to them that provides low latency.

**Dependency**

Route 53 provides a high level of dependability required by critical applications

**Optimal Locations**

Route 53 uses a global anycast network to answer queries from the optimal position automatically.

# RDS

Tuesday, September 7, 2021 1:55 AM

Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud easier to set up, manage, and maintain than running Oracle Database in Amazon EC2

RDS is a Database as a Service (DBaaS) that automatically configures and maintains your databases in the AWS cloud

A read replica is a copy of the primary instance that reflects changes to the primary in almost real time, in normal circumstances.

Amazon RDS replicates all databases in the source DB instance

Amazon RDS sets up a secure communications channel between the primary DB instance and the read replica.

## 53) Name some of the DB engines which can be used in AWS RDS

1. MS-SQL DB
2. MariaDB
3. MySQL DB
4. OracleDB
5. PostgreDB

## 34) What is SimpleDB?

SimpleDB is a data repository of structure record which encourages data doubts and indexing both S3 and EC2 are called SimpleDB.

## 25) What is a redshift?

Redshift is a big data warehouse product. It is fast and powerful, fully managed data warehouse service in the cloud.

## 9) What is SQL?

Simple Queues Services also known as SQL.

# AMI

Isnin, 6 September 2021 2:42 PTG

## ### AMI ###

- AMI stands for Amazon Machine Image. It's a template that provides the information (an operating system, an application server, and applications) required to launch an instance, which is a copy of the AMI running as a virtual server in the cloud. You can launch instances from as many different AMIs as you need.
- AMI where all EC2 instance starts.
- it's bit for bit copy of root volume that stored in S3, and it launch clones of original.
- The user have no direct access to that bucket or object.
- We can launch different types of instances from a single AMI.
- Each instance type offers different compute and memory capabilities.
- we can use sudo to run commands that require root privileges.

## 5) Mention what the relationship between an instance and AMI is?

From a single AMI, you can launch multiple types of instances. Each instance type provides different computer and memory capabilities.

## 6) What does an AMI include?

- A template for the root volume for the instance
- Launch permissions decide which AWS accounts can avail the AMI to launch instances
- A block device mapping that determines the volumes to attach to the instance when it is launched

## Difference between An Instance and AMI

AMI is a template consisting software configuration part. For example Operating systems, applications, application server if you start an instance, a duplicate of the AMI in a row as an attendant in the cloud.

## 41) What is boot time taken for the instance stored backed AMI?

less than 5 minutes.

## 37) Name the types of AMI provided by AWS

The types of AMI provided by AWS are:

1. Instance store backed
2. EBS backed

# Cognito

Thursday, September 23, 2021 1:16 PM

**a simple user identity and data synchronization service** that helps you **securely manage and synchronize app data for your users across their mobile devices.**

**"Securely manage and synchronize app data for your users across their mobile devices"**

AWS Cognito- AWS Service Used for User authentication and Authorization. **This service can be integrated with onprem AD also.**

**IAM** is detailed as "**Securely control access to AWS services and resources for your users**"

**A user pool** is a user directory in Amazon Cognito. With a user pool, **your users can sign in to your web or mobile app through Amazon Cognito**. Your users can also sign in through social identity providers like Google, Facebook, Amazon, or Apple, and through SAML

## EBS

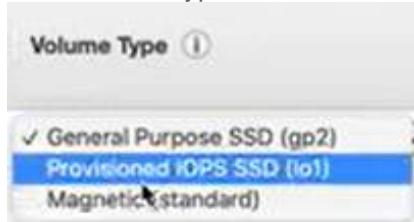
Tuesday, September 7, 2021 1:56 AM

It's Raw block-level storage that can be attached to Amazon EC2 instances and is used by Amazon Relational Database Service. Amazon EBS provides a range of options for storage performance and cost.

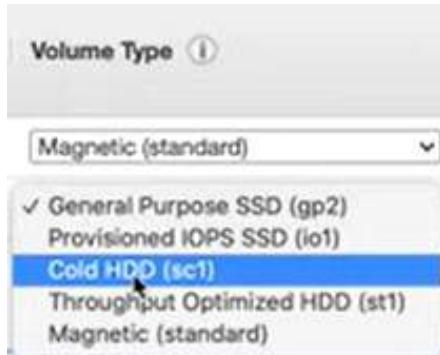
### 43) How to connect EBS volume to multiple instances?

We can't be able to connect EBS volume to multiple instances. Although, you can connect various EBS Volumes to a single instance.

Root Device Type



EBS



## Elastic Block Storage

### 38. How can you automate EC2 backup using EBS?

Use the following steps in order to automate EC2 backup using EBS:

1. Get the list of instances and connect to AWS through API to list the Amazon EBS volumes that are attached locally to the instance.
2. List the snapshots of each volume, and assign a retention period of the snapshot. Later on, create a snapshot of each volume.
3. Make sure to remove the snapshot if it is older than the retention period.

### 39. What is the difference between EBS and Instance Store?

EBS is a kind of permanent storage in which the data can be restored at a later point. When you save data in the EBS, it stays even after the lifetime of the EC2 instance. On the other hand, Instance Store is temporary storage that is physically attached to a host machine. With an Instance Store, you cannot detach one instance and attach it to another. Unlike in EBS, data in an Instance Store is lost if any instance is stopped or terminated.

### 40. Can you take a backup of EFS like EBS, and if yes, how?

Yes, you can use the EFS-to-EFS backup solution to recover from unintended changes or deletion in Amazon EFS. Follow these steps:

1. Sign in to the AWS Management Console
2. Click the launch EFS-to-EFS-restore button
3. Use the region selector in the console navigation bar to select region
4. Verify if you have chosen the right template on the Select Template page
5. Assign a name to your solution stack
6. Review the parameters for the template and modify them if necessary

### 41. How do you auto-delete old snapshots?

Here's the procedure for auto-deleting old snapshots:

- As per procedure and best practices, take snapshots of the EBS volumes on Amazon S3.
- Use AWS Ops Automator to handle all the snapshots automatically.
- This allows you to create, copy, and delete Amazon EBS snapshots.

# CloudFormation

Monday, September 13, 2021 1:39 PM

A template is a declaration of the AWS resources that make up a stack.

**CloudFormation** creates a bucket for each region in which you upload a template file. The buckets are accessible to anyone with S3 permissions in your AWS account. If a bucket created by CloudFormation is already present, the template is added to that bucket.

A CloudFormation template consists of **6 sections – Description, Parameters, Mappings, Conditions, Resources and Outputs**.

Upload the template file on **CloudFormation console** while creating a stack. View your template in Designer mode for the resource flow and creation clarity or to make any changes if needed. Complete the configuration and create a stack.

<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/quickref-ec2.html>

### 35. How is AWS CloudFormation different from AWS Elastic Beanstalk?

Here are some differences between AWS CloudFormation and AWS Elastic Beanstalk:

- AWS CloudFormation helps you provision and describe all of the infrastructure resources that are present in your cloud environment. On the other hand, AWS Elastic Beanstalk provides an environment that makes it easy to deploy and run applications in the cloud.
- AWS CloudFormation supports the infrastructure needs of various types of applications, like legacy applications and existing enterprise applications. On the other hand, AWS Elastic Beanstalk is combined with the developer tools to help you manage the lifecycle of your applications.

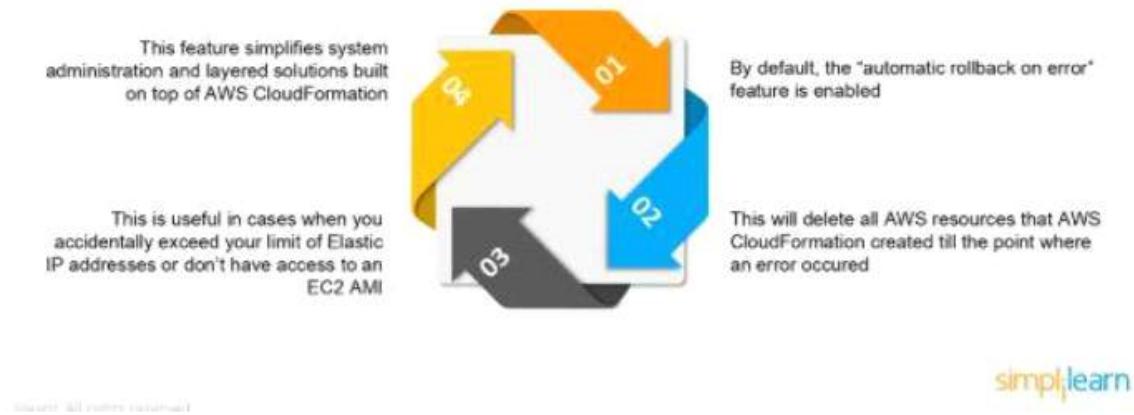
### 36. What are the elements of an AWS CloudFormation template?

AWS CloudFormation templates are YAML or JSON formatted text files that are comprised of five essential elements, they are:

- Template parameters
- Output values
- Data tables
- Resources
- File format version

### 37. What happens when one of the resources in a stack cannot be created successfully?

If the resource in the stack cannot be created, then the CloudFormation automatically rolls back and terminates all the resources that were created in the CloudFormation template. This is a handy feature when you accidentally exceed your limit of Elastic IP addresses or don't have access to an EC2 AMI.



simplilearn

CloudFormation templates to get you up and running quickly

Edge caching in Amazon CloudFront (1) to cache content close to end users for faster delivery.

CloudFront pulls static content from an S3 bucket (2) and dynamic content from an Application Load Balancer (4) in front of the web instances.

The web instances run in an Auto Scaling group of Amazon EC2 instances (6).

An ElastiCache cluster (7) caches frequently queried data to speed up responses.

An Amazon Aurora MySQL instance (8) hosts the WordPress database. The WordPress EC2 instances access shared WordPress data on an Amazon EFS file system via an EFS Mount Target (9) in each Availability Zone. An Internet Gateway (3) allows communication between resources in your VPC and the internet. NAT Gateways (5) in each Availability Zone enable EC2 instances in private subnets (App and Data) to access the internet. Within the Amazon VPC there exist two types of subnets: public (Public Subnet) and private (App Subnet and Data Subnet). Resources deployed into the public subnets will receive a public IP address and will be publicly visible on the internet. The Application Load Balancer (4) and a Bastion host for administration are deployed here. Resources deployed into the

private subnets receive only a private IP address and hence are not publicly visible on the internet, improving the security of those resources.

The WordPress web server instances (6), ElastiCache cluster instances (7), Aurora MySQL database instances (8), and EFS Mount Targets (9) are all deployed in private subnets.

# CodeDeploy

Thursday, September 23, 2021 4:25 PM

**AWS CodeDeploy** is a fully managed deployment service that automates software deployments to a variety of compute services such as Amazon EC2, AWS Fargate, AWS Lambda, and your on-premises servers

How do code deployments work?

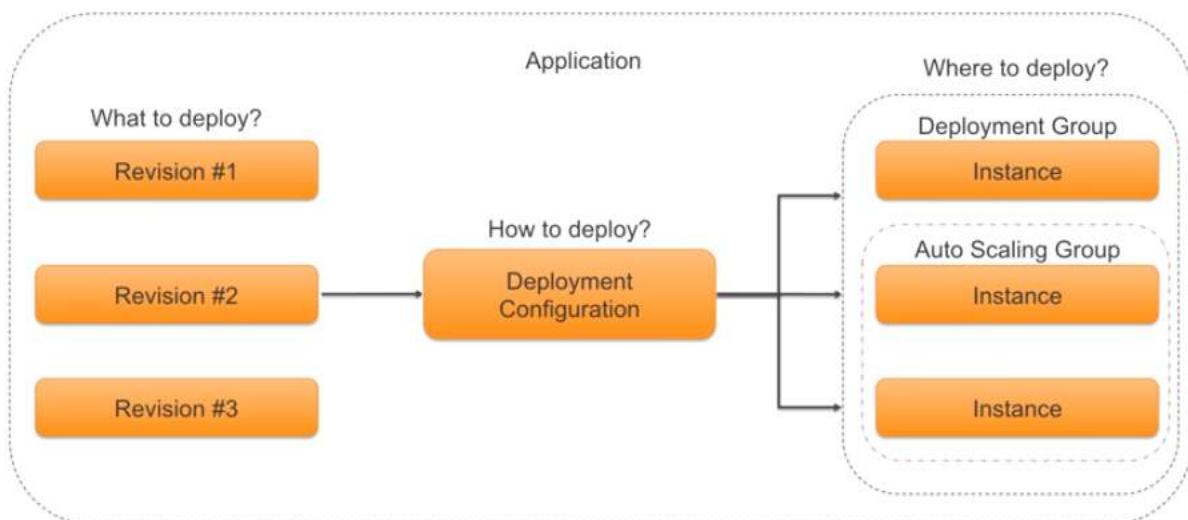
Once developers have written code for a site, they need to place it on the web servers. That process is called code deployment. ... It's called code deployment. It can include code that fixes bugs, adds new features, or upgrades the underlying platform.

**AWS CodeBuild** is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy. With CodeBuild, you don't need to provision, manage, and scale your own build servers.

## Steps

1. Step 1: Create the source code.
2. Step 2: Create the buildspec file.
3. Step 3: Create two S3 buckets.
4. Step 4: Upload the source code and the buildspec file.
5. Step 5: Create the build project.
6. Step 6: Run the build.
7. Step 7: View summarized build information.
8. Step 8: View detailed build information.

# Concepts



# AppSpec File

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/html/WordPress
hooks:
  ApplicationStop:
    - location: helper_scripts/stop_server.sh
  BeforeInstall:
    - location: deploy_hooks/install-apache.sh
    - location: deploy_hooks/install-mysql.sh
  AfterInstall:
    - location: deploy_hooks/configure_app.sh
      timeout: 30
      runas: root
```

Step 1: Welcome

Step 2: Instance Settings

Step 3: Application Name

Step 4: Revision

Step 5: Deployment Group

Step 6: Service Role

Step 7: Deployment Configuration

Step 8: Review

## Instance Settings



To help test the sample application deployment, AWS CodeDeploy will use AWS CloudFormation to launch three Amazon EC2 instances with the configuration below. After they launch, the instances will be ready to participate in deployments.

Operating System\*  Amazon Linux

Windows Server

Instance Type\* t1.micro



Key Pair Name\* EC2KeyPair



Tag Key and Value\* Name CodeDeployDemo



Key Value

Launch the instances in AWS CloudFormation. This may take a few minutes.

**Launch Instances**

Click **Skip This Step** if you have existing Amazon EC2 instances that you want to deploy to.

\*Required

Cancel

Previous

Skip This Step

**Next Step**

AWS CodeDeploy Manager... https://s3-us-west-2.amazonaws.com/codedeploy/home?region=us-west-2&list=run/instance-settings

AWS Services Edit

**Step 1: Welcome**

**Step 2: Instance Settings**

Step 3: Application Name  
Step 4: Revision  
Step 5: Deployment Group  
Step 6: Service Role  
Step 7: Deployment Configuration  
Step 8: Review

## Instance Settings

To help test the sample application deployment, AWS CodeDeploy will use AWS CloudFormation to launch three Amazon EC2 instances with the configuration below. After they launch, the instances will be ready to participate in deployments.

**Operating System\***  Amazon Linux  Windows Server

**Instance Type\*** t1.micro

**Key Pair Name\*** EC2KeyPair

**Tag Key and Value\*** Name: CodeDeployDemo

We're launching your instances now. It may take several minutes for us to set up the instances and run them through some system checks.

Step 30 of 35 complete

See more details in AWS CloudFormation

\*Required Cancel Previous Skip This Step Next Step

## Application Name

Type a name that uniquely identifies the application that you want to deploy. AWS CodeDeploy will group the application revision, deployment group, service role, and deployment configuration under this application name.

Application Name\* DemoApplication

\*Required

Cancel

Previous

Next Step

## Revision

A revision in AWS CodeDeploy is a version of an application that you want to deploy. Our sample application revisions are stored in Amazon S3.

Revision Type Sample Amazon Linux Application

Revision Location

https://s3-us-west-2.amazonaws.com/ aws-codedeploy-us-west-2/samples/latest /SampleApp\_Linux.zip

Download Sample Bundle

The URL of the sample application in Amazon S3.

Revision Description

Sample web page for Amazon Linux. To view the sample web page after deployment, from your web browser go to http://<Public DNS>, for example http://ec2-12-345-678-901.compute-1.amazonaws.com.

\*Required

Cancel

Previous

Next Step

Step 1: Welcome  
Step 2: Instance Settings  
Step 3: Application Name  
Step 4: Revision  
**Step 5: Deployment Group**  
Step 6: Service Role  
Step 7: Deployment Configuration  
Step 8: Review

## Deployment Group



Create a new deployment group, which is a collection of Amazon EC2 instances to deploy to.

Deployment Group Name\*

Development

### Add Instances

Specify existing Amazon EC2 instances to deploy to by entering their Amazon EC2 tags (key/value pairs), Auto Scaling group names, or both.

#### Search by Amazon EC2 Tags

	Key	Value	Instances	X
1	Name	CodeDeployDemo	3	X
2				

Search by Auto Scaling Group Names

Total Matching Amazon EC2 Instances: 3

\*Required

Cancel

Previous

Next Step

Step 1: Welcome  
Step 2: Instance Settings  
Step 3: Application Name  
Step 4: Revision  
Step 5: Deployment Group  
**Step 6: Service Role**  
Step 7: Deployment Configuration  
Step 8: Review

## Service Role



Select an existing service role that allows AWS CodeDeploy to work with other dependent AWS services on your behalf during a deployment. If you're not sure if you already have a service role with the correct permissions, in the Service Role drop-down list select Create A New Service Role, and we will create one for you.

Service Role\*

Use an existing service role

Role Name\*

CFTestNewTemplate-CodeDeployTrus

\*Required

Cancel

Previous

Next Step

Step 1: Welcome  
Step 2: Instance Settings  
Step 3: Application Name  
Step 4: Revision  
Step 5: Deployment Group  
Step 6: Service Role  
**Step 7: Deployment Configuration**  
Step 8: Review

## Deployment Configuration

?

Choose from a list of default deployment configurations, or create a custom configuration.

Default Deployment Configurations

Create Custom Deployment Configuration

### One at a Time

#### The deployment will:

Deploy to one instance at a time. Success if all instances succeed. Fail after the very first failure. Allow the deployment to succeed for some instances, even if the overall deployment fails.

#### Example:

If you deploy your application to 3 instances, this configuration will deploy to one instance at a time.

 Successes if all 3 instances succeed.

 Fails after any instance fails.

Select

### Half at a Time

#### The deployment will:

Deploy to up to half of the instances at a time, with failures rounded down. Success if at least half of the instances succeed; otherwise it will fail. The deployment may succeed for some instances, even if the overall deployment fails.

#### Example:

If you deploy your application to 3 instances, this configuration will deploy to one instance at a time.

 Successes if 2 or more instances succeed.

 Fails if 2 or more instances fail.

Select

### All at Once

#### The deployment will:

Deploy to all instances at once. Success if at least one instance succeeds. Fail after all instances fail.

#### Example:

If you deploy your application to 3 instances, this configuration will deploy to all 3 instances at once.

 Successes if any instance succeeds.

 Fails if all instances fail.

Select

Step 1: Welcome

Step 2: Instance Settings

Step 3: Application Name

Step 4: Revision

Step 5: Deployment Group

Step 6: Service Role

Step 7: Deployment Configuration

**Step 8: Review**

### Review

Review the details of your deployment. To make any changes, click Edit, Previous, or one of the steps in the navigation pane. When you're ready to deploy with these details, click Deploy Now.

**You are about to create the following deployment.**

<b>Application</b>	<a href="#">Edit</a>
You will create the application <b>DemoApplication</b>	
<b>Revision</b>	<a href="#">Edit</a>
You will deploy the following revision of the application <b>DemoApplication</b> .	
Revision: <a href="https://s3-us-west-2.amazonaws.com/aws-codedeploy-us-west-2/samples/latest/SampleApp_Linux.zip">https://s3-us-west-2.amazonaws.com/aws-codedeploy-us-west-2/samples/latest/SampleApp_Linux.zip</a>	
<b>Deployment Group</b>	<a href="#">Edit</a>
DemoApplication will be deployed to your instances using the deployment group <b>Development</b>	
<b>Service Role</b>	<a href="#">Edit</a>
Development will use the <b>CodeDeploy_DemoApplication</b> service role to access the instances.	
<b>Deployment Configuration</b>	<a href="#">Edit</a>
You will deploy DemoApplication to your instances using the following deployment configuration	
Deployment Configuration: <b>CodeDeploy/Default.OneAtATime</b>	

\*Required

[Cancel](#) [Previous](#) [Deploy Now](#)

AWS CodeDeploy

### Deployments

View, diagnose, and manage your deployments.

[Create New Deployment](#)

Filter: All Deployments		Q Search by Deployment ID	Deployments per page: 10				< Viewing 1 to 1 Deployment(s) >	
Deployment ID	Application	Deployment Group	Revision Location	Start Time	End Time	Status	Actions	
d-8BOKNQ6	DemoApplication	Development	s3://aws-codedeploy-us-west-2/samples/latest/SampleApp_Linux.zip	1 second ago		In Progress	<a href="#">Stop</a>	

**Details**

Deployment ID	d-8BOKNQ6
Deployment Config	CodeDeploy/Default.OneAtATime
Minimum Healthy Hosts	2 of 3 instances
Revision Location	s3://aws-codedeploy-us-west-2/samples/latest/SampleApp_Linux.zip

**Instances**

1 of 3 Instances Completed		
<div style="width: 33.33%; background-color: green;">1</div> Succeeded	<div style="width: 33.33%; background-color: blue;">1</div> In Progress	<div style="width: 33.33%; background-color: orange;">1</div> Pending
<a href="#">View All Instances</a>		

AWS CodeDeploy		Deployments	Deployment d-88OKNQ6	Instance i-5aaadb56 Events
Events for Instance i-5aaadb56				
Deployment Group Details			Revision	
Deployment Group	Development		Revision Location	s3://aws-codedeploy-us-west-2
Deployment ID	d-88OKNQ6		Revision Created	44 seconds ago
Deployment Config	CodeDeployDefault.OneAtATime		Description	Application revision registered t
Minimum Healthy Hosts	2 of 3 instances			
Event	Start Time	End Time	Duration	Status
ApplicationStop	8 seconds ago	8 seconds ago	less than one sec	Succeeded
DownloadBundle	7 seconds ago	7 seconds ago	less than one sec	Succeeded
BeforeInstall	6 seconds ago	2 seconds ago	3 secs	Succeeded
Install	1 second ago	1 second ago	less than one sec	Succeeded
AfterInstall				Pending
ApplicationStart				Pending
ValidateService				Pending

# Security

Wednesday, October 6, 2021 10:55 PM

- **1. What are the important cloud security aspects in AWS?**

**Ans:** The two critical cloud security aspects in AWS refer to authentication and authorization and access control. Authentication and authorization allow genuine users to access data and applications. On the other hand, access control helps in restricting the access of other users trying to enter the AWS cloud environment.

- **2. What are the important security precautions before migration to AWS Cloud?**

**Ans:** The important precautions that users must take before migration to AWS cloud should be to focus on the following areas.

- Data integrity
- Data loss
- Data storage
- Business continuity
- Uptime
- Compliance with rules and regulations

- **3. What are the laws implemented for security of cloud data?**

**Ans:** The different security laws applicable to cloud data are relevant for different stages in data lifecycle. The laws for validation of input help in controlling input data. The backup and security laws ensure the security and storage of data, thereby controlling data breaches. Output and reconciliation laws help in ensuring controls of data selected for reconciliation from input to output. The laws for processing ensure proper controls over the data processed in an application.

- **4. What are the infrastructure security products on AWS?**

**Ans:** AWS facilitates different security capabilities and services for increasing privacy and control over network access. You can find connectivity options for enabling private or dedicated connection from on-premises or office environment. Infrastructure security also involves encryption of all traffic on AWS global and regional networks among AWS secured facilities.

- **5. What are inventory and configuration management security features on AWS?**

**Ans:** The important best practices for security of inventory and configuration management in AWS include,

- Inventory and configuration management tools for identification of AWS resources followed by tracking and management of changes to the resources over time.
- Deployment tools for management of creation and decommissioning AWS resources in accordance with organization standards
- Tools for template definition and management for creation of standard, hardened, preconfigured virtual machines for EC2 instances

- **6. What is AWS Identity and Access Management (IAM)?**

**Ans:** AWS Identity and Access Management (IAM) is the service that helps you provide definitions for individual user accounts with permissions across different AWS resources. AWS IAM also includes multi-factor authentication tailored specially for privileged accounts. In addition, you can also find the options for hardware-based and software-based authenticators in AWS IAM.

- **7. What is AWS Directory Service?**

**Ans:** The AWS Directory Service is the ideal service for integration and federating with corporate directories. As a result, users can reduce the administrative overhead alongside ensuring the improvement of end-user experiences.

- **8. What is AWS Single Sign-On?**

**Ans:** AWS Single Sign-On or (AWS SSO) is helpful for users to ensure the management of SSO access. It also provides centralized management of user permissions to all accounts in AWS organizations.

- **9. What is AWS CloudTrail?**

**Ans:** AWS CloudTrail is the cloud monitoring service of AWS that helps in monitoring AWS deployments in the cloud. CloudTrail achieves this through a history of AWS API calls for a concerned account.

- **10. Define Amazon GuardDuty?**

**Ans:** Amazon GuardDuty is the threat detection service for continuous monitoring of malicious activity and unauthorized behavior for safeguarding AWS accounts and workloads.

- **11. What is Amazon CloudWatch?**

**Ans:** Amazon CloudWatch is a reliable cloud service that gives a monitoring solution with an assurance of reliability, flexibility, and scalability. Users can

start and utilize CloudWatch quickly as it does not take long for setup and then management and scaling of your monitoring systems and infrastructure.

- **12. Define AWS Trusted Advisor.**

**Ans:** AWS Trusted Advisor serves as an ideal online tool serving as a customized cloud expert. It can help you with resource configuration in accordance with best practices. It also evaluates the AWS environment thoroughly for addressing any security gaps.

- **13. What is the role of AWS Security Bulletins?**

**Ans:** AWS Security Bulletins are one of the most reliable sources of updated information on existing threats and vulnerabilities. These security bulletins help customers to work in close quarters with AWS security experts to address vulnerabilities and report abuse.

- **14. What is the significance of AWS Well-Architected Framework?**

**Ans:** The AWS Well-Architected Framework establishes the foundation for cloud architects to develop cloud infrastructure for their applications with higher security, efficiency, performance, and resilience. The security pillar in the AWS Well-Architected Framework establishes the ideal precedents for data integrity, system protection, and controls for detection of security events.

- **15. What are the notable advantages of AWS security?**

**Ans:** The striking benefits of AWS security include the following,

- Secure scalability with better visibility and control.
- Automation of security controls and reduction of risk associated with deeply integrated services.
- Compliance with the highest benchmarks of data security and privacy.
- Extensive community support.

From <<https://www.infosectrain.com/blog/top-15-aws-security-interview-questions/>>

## What's an Availability Zone?

Think of an Availability Zone as a **data center**.

## Multiple Data Centers

An Availability Zone may be several data centers, but because they are close together, they are counted as **1 Availability Zone**.

## What's a Data Center?

A data center is just a building filled with **servers**.

## Edge Locations

Edge locations are endpoints for AWS that are used for caching content.

Typically, this consists of **CloudFront**, Amazon's content delivery network (CDN).

There are **many more edge locations** than Regions.

Currently, there are **over 215 edge locations**.

## What's a Region?

A Region is a geographical area. Each Region consists of **2 (or more) Availability Zones**.



## AWS Service Types

A CLOUD GURU

End User Computing	Quantum Technologies	Containers
IOT	Customer Enablement	Game Development
Customer Engagement	Business Applications	Desktop & App Streaming
AR & VR	Application Integration	AWS Cost Management
Analytics	Security, Identity & Compliance	Mobile
Management & Governance	Media Services	Machine Learning
Robotics	Blockchain	Satellite
Migration & Transfer	Network & Content Delivery	Developer Tools
Compute	Storage	Databases
AWS Global Infrastructure		

End User Computing	Quantum Technologies	Containers
IOT	Customer Enablement	Game Development
Customer Engagement	Business Applications	Desktop & App Streaming
AR & VR	Application Integration	AWS Cost Management
Analytics	Security, Identity & Compliance	Mobile
Management & Governance	Media Services	Machine Learning
Robotics	Blockchain	Satellite
Migration & Transfer	Network & Content Delivery	Developer Tools
Compute	Storage	Databases
AWS Global Infrastructure		

## Compute

You wouldn't be able to build an application without compute power — you need something crunching the data.

- ✓ EC2
- ✓ Lambda
- ✓ Elastic Beanstalk

## Storage

Think of storage like a giant disk in the cloud — it's a safe place to save your information.

- ✓ S3
- ✓ FSx
- ✓ EBS
- ✓ Storage Gateway
- ✓ EFS

## Databases

The easiest way to think of a database is a spreadsheet. It's a reliable way to store and retrieve information.

- ✓ RDS
- ✓ DynamoDB
- ✓ Redshift

## Networking

We need a way for our compute, storage, and databases to communicate and even a place for them to live. This is where networking comes in.

- ✓ VPCs
- ✓ API Gateway
- ✓ Direct Connect
- ✓ AWS Global Accelerator
- ✓ Route 53

# Responsibility

Tuesday, September 21, 2021 10:54 PM



## The Shared Responsibility Model



## Can you do this yourself in the AWS Management Console?

- **If yes, you are likely responsible.**  
Security groups, IAM users, patching EC2 operating systems, patching databases running on EC2, etc.
- **If not, AWS is likely responsible.**  
Management of data centers, security cameras, cabling, patching RDS operating systems, etc.
- **Encryption is a shared responsibility.**

# 5 Pillars of the Well-Architected Framework



## Operational Excellence

Focuses on running and monitoring systems to deliver business value, and continually improving processes and procedures.



## Security

Focuses on protecting information and systems.



## Reliability

Focuses on ensuring a workload performs its intended function correctly and consistently when it's expected to.



## Performance Efficiency

Focuses on using IT and computing resources efficiently.



## Cost Optimization

Focuses on avoiding unnecessary costs.

# Identity and Access Management (IAM)

Tuesday, September 21, 2021 11:25 PM

## What Is IAM?

**IAM** allows you to manage users and their level of access to the **AWS** console.

- ✓ Create users and grant permissions to those users.
- ✓ Create groups and roles.
- ✓ Control access to AWS resources.

The **root account** is the email address you used to sign up for AWS. The root account has **full administrative access** to AWS. For this reason, it is important to secure this account.

## 4 Steps to Secure Your AWS Root Account

- ✓ Enable multi-factor authentication on the root account.
- ✓ Create an admin group for your administrators, and assign the appropriate permissions to this group.
- ✓ Create user accounts for your administrators.
- ✓ Add your users to the admin group.

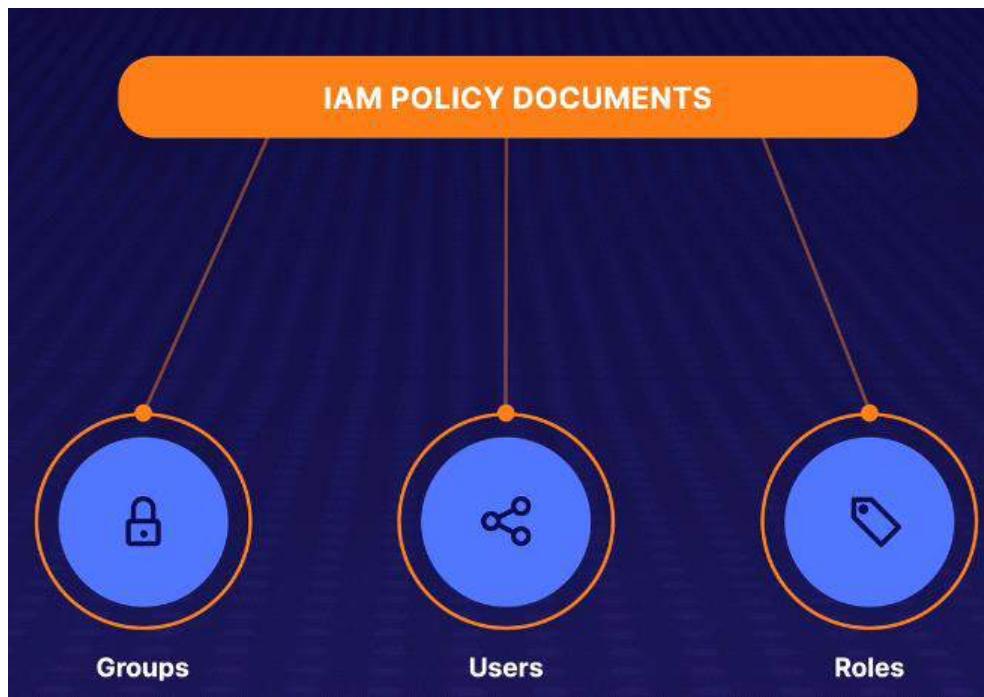
# How Do We Control Permissions Using IAM?

We assign permissions using policy documents, which are made up of JSON (JavaScript Object Notation).

## JSON

### Example of a Policy Document

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "*",  
            "Resource": "*"  
        }  
    ]  
}
```



## S3 Basic

Monday, October 11, 2021 10:36 PM

### What Is S3?

- Object Storage**  
S3 provides secure, durable, highly scalable object storage.
- Scalable**  
S3 allows you to store and retrieve any amount of data from anywhere on the web at a very low cost.
- Simple**  
Amazon S3 is easy to use, with a simple web service interface.

### S3 Basics

- Unlimited Storage**  
The total volume of data and the number of objects you can store is unlimited.
- Objects up to 5 TB in Size**  
S3 objects can range in size from a minimum of 0 bytes to a maximum of 5 terabytes.
- S3 Buckets**  
Store files in buckets (similar to folders).

**S3 IS OBJECT-BASED STORAGE**

**Manages data as objects rather than in file systems or data blocks.**

- ✓ Upload any file type you can think of to S3.
- ✓ Examples include photos, videos, code, documents, and text files.
- ✓ Cannot be used to run an operating system or database.

### Working with S3 Buckets

- Universal Namespace**  
All AWS accounts share the S3 namespace. Each S3 bucket name is globally unique.
- Example S3 URLs**  
<https://bucket-name.s3.Region.amazonaws.com/key-name>  
<https://acloudguru.s3.us-east-1.amazonaws.com/Ralphie.jpg>

### Working with S3 Buckets

- Universal Namespace**  
All AWS accounts share the S3 namespace. Each S3 bucket name is globally unique.
- Example S3 URLs**  
<https://bucket-name.s3.Region.amazonaws.com/key-name>  
<https://acloudguru.s3.us-east-1.amazonaws.com/Ralphie.jpg>
- Uploading Files**  
When you upload a file to an S3 bucket, you will receive an HTTP 200 code if the upload was successful.

<b>Key</b> The name of the object (e.g., Ralphie.jpg)	<b>Version ID</b> Important for storing multiple versions of the same object
<b>Value</b> The data itself, which is made up of a sequence of bytes	<b>Metadata</b> Data about the data you are storing (e.g., content-type, last-modified, etc.)

- ✓ **Built for Availability**  
Built for 99.95% – 99.99% **service availability**, depending on the S3 tier.
- ✓ **Designed for Durability**  
Designed for 99.99999999% (9 decimal places) durability for **data stored** in S3.

### S3 is a safe place to store your files.

The data is spread across multiple devices and facilities to **ensure availability** and **durability**.

### S3 Standard

- High Availability and Durability**  
Data is stored redundantly across multiple devices in multiple facilities (>=3 AZs):
  - **99.99% availability**
  - **99.99999999% durability** (11 9's)
- Designed for Frequent Access**  
Perfect for frequently accessed data.
- Suitable for Most Workloads**
  - The default storage class.

- Tiered Storage**  
S3 offers a range of storage classes designed for different use cases.
- Lifecycle Management**  
Define rules to automatically transition objects to a cheaper storage tier or delete objects that are no longer required after a set period of time.
- Versioning**  
With versioning, all versions of an object are stored and can be retrieved, including deleted objects.

3

**Suitable for Most Workloads**

- The default storage class.

## Strong Read-After-Write Consistency

- After a successful write** of a new object (PUT) or an overwrite of an existing object, any subsequent read request immediately receives the latest version of the object.
- Strong consistency** for list operations, so after a write, you can immediately perform a listing of the objects in a bucket with all changes reflected.

deleted objects.

## Securing Your Data

1

**Server-Side Encryption**

You can set default encryption on a bucket to encrypt all new objects when they are stored in the bucket.

2

**Access Control Lists (ACLs)**

Define which AWS accounts or groups are granted access and the type of access. You can attach S3 ACLs to individual objects within a bucket.

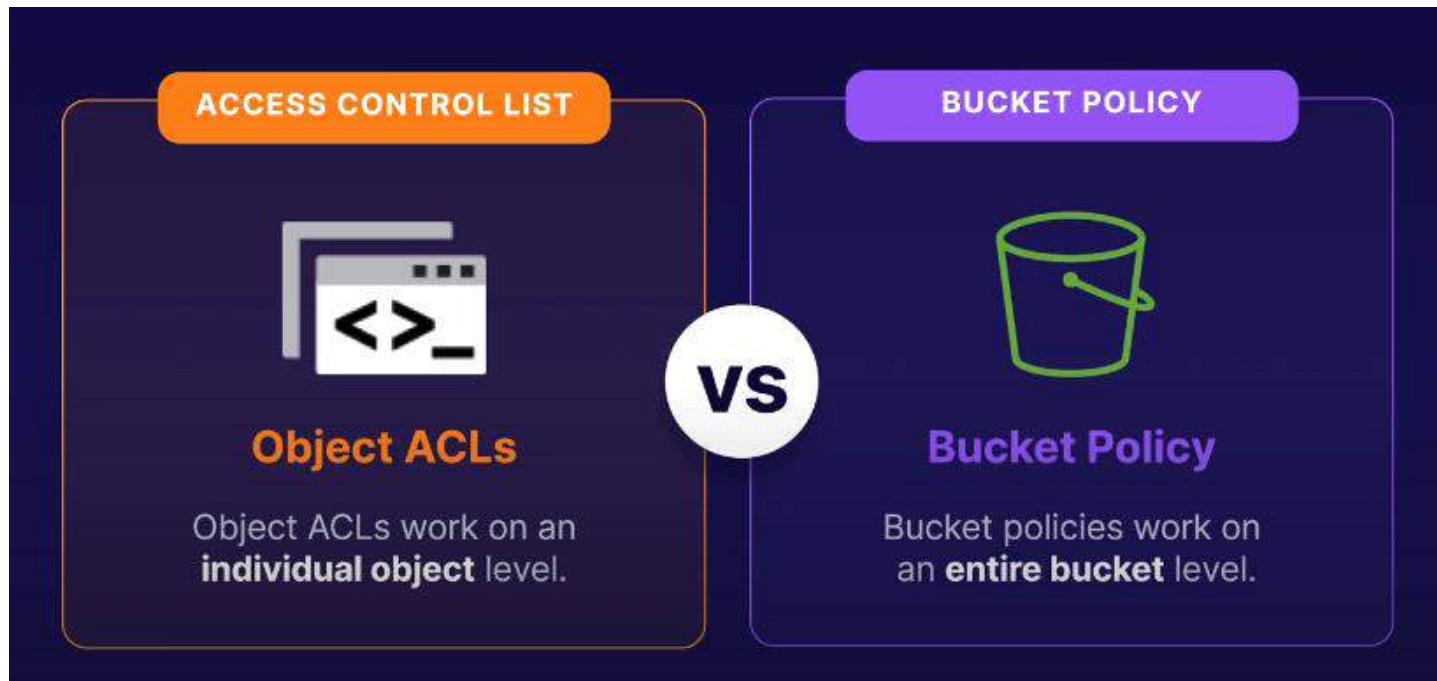
3

**Bucket Policies**

S3 bucket policies specify what actions are allowed or denied (e.g., allow user Alice to PUT but not DELETE objects in the bucket).

# Securing S3 Block Public Access

Monday, October 11, 2021 10:52 PM



## What to Know for the Exam

- ✓ **Buckets are private by default:** When you create an S3 bucket, it is private by default (including all objects within it). You have to allow public access on both the **bucket** and its **objects** in order to make the bucket public.
- ✓ **Object ACLs:** You can make **individual objects** public using object ACLs.
- ✓ **Bucket policies:** You can make **entire buckets** public using bucket policies.
- ✓ **HTTP status code:** When you upload an object to S3 and it's successful, you will receive an **HTTP 200** code.

# Hosting a Static Website Using S3

Monday, October 11, 2021 11:08 PM

## STATIC WEBSITES ON S3

**You can use S3 to host static websites, such as .html sites.**

**Dynamic** websites, such as those that require **database connections**, cannot be hosted on S3.

## S3 Scales Automatically

**S3 scales automatically to meet demand.** Many enterprises will put static websites on S3 when they think there is going to be a large number of requests (e.g., for a movie preview).

## Versioning Objects in S3

Tuesday, October 12, 2021 3:01 AM

# What Is Versioning?

You can enable versioning in S3 so you can have **multiple versions of an object within S3**.

## Advantages of Versioning

- All Versions**  
All versions of an object are stored in S3. This includes all writes and even if you delete an object.
- Backup**  
Can be a great backup tool.
- Cannot Be Disabled**  
Once enabled, versioning cannot be disabled — only suspended.
- Lifecycle Rules**  
Can be integrated with lifecycle rules.
- Supports MFA**  
Can support multi-factor authentication.

Amazon S3 > myjanjuawebiste

myjanjuawebiste [Info](#)

Publicly accessible

Objects Properties Permissions Metrics Management Access Points

### Bucket overview

AWS Region	Amazon Resource Name (ARN)
Asia Pacific (Singapore) ap-southeast-1	<a href="#">arn:aws:s3:::myjanjuawebiste</a>

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and n user actions and application failures. [Learn more](#)

Edit

Find objects by prefix  Show versions

<input type="checkbox"/>	Name	Type	Version ID
<input type="checkbox"/>	<a href="#">error.html</a>	html	null
<input type="checkbox"/>	<a href="#">index.html</a>	html	F7wbAwG_VJvMd_4plUMkaXeadUhfqGeK
<input type="checkbox"/>	<a href="#">index.html</a>	html	nJD4nD1FS23AewwfMZWq1l97z7edZ8tV
<input type="checkbox"/>	<a href="#">index.html</a>	html	null

- All Versions:** All versions of an object are stored in S3. This includes all writes and even if you delete an object.
- Backup:** Can be a great backup tool.
- Cannot Be Disabled:** Once enabled, versioning cannot be disabled — only suspended.
- Lifecycle Rules:** Can be integrated with lifecycle rules.
- Supports MFA:** Can support multi-factor authentication.

## S3 Storage Classes

Tuesday, October 12, 2021 3:30 AM

S3 Standard	
1	<b>High Availability and Durability</b> Data is stored redundantly across multiple devices in multiple facilities (>=3 AZs): <ul style="list-style-type: none"><li>99.99% availability</li><li>99.99999999% durability (11 9's)</li></ul>
2	<b>Designed for Frequent Access</b> Perfect for frequently accessed data.
3	<b>Suitable for Most Workloads</b> <ul style="list-style-type: none"><li>The default storage class.</li><li>Use cases include websites, content distribution, mobile and gaming applications, and big data analytics.</li></ul>



S3 ONE ZONE-INFREQUENT ACCESS	
<b>Like S3 Standard-IA, but data is stored redundantly within a single AZ.</b>	
<ul style="list-style-type: none"> <li>Costs <b>20% less</b> than regular S3 Standard-IA</li> <li>Great for long-lived, infrequently accessed, non-critical data</li> </ul>	

2 Glacier Options	
<ul style="list-style-type: none"> <li>Glacier is cheap storage.</li> <li>Optimized for data that is very infrequently accessed.</li> <li>You pay each time you access your data.</li> <li>Use only for archiving data.</li> </ul>	<b>OPTION 1</b> <b>Glacier</b> Provides <b>long-term data archiving</b> with retrieval times that range from 1 minute to 12 hours (e.g., historical data only accessed a few times per year).
	<b>OPTION 2</b> <b>Glacier Deep Archive</b> Archiving <b>rarely accessed data</b> with a default retrieval time of 12 hours (e.g., financial records that may be accessed once or twice per year).

S3 Intelligent-Tiering	
<b>2 TIERS</b>	
<b>Frequent and Infrequent Access</b>	
Automatically moves your data to the most cost-effective tier based on how frequently you access each object.	

Performance across the S3 Storage Classes						
	S3 Standard	S3 Intelligent-Tiering*	S3 Standard-IA	S3 One Zone-IA+	S3 Glacier	S3 Glacier Deep Archive
<b>Designed for Durability</b>	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
<b>Designed for Availability</b>	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
<b>Availability SLA</b>	99.9%	99%	99%	99%	99.9%	99.9%
<b>Availability Zone(s)</b>	≥ 3	≥ 3	≥ 3	1	≥ 3	≥ 3
<b>Min Capacity Charge per Object</b>	N/A	N/A	128KB	128KB	40KB	40KB
<b>Minimum Storage Duration Charge</b>	N/A	30 days	30 days	30 days	90 days	180 days
<b>Retrieval Fee</b>	N/A	N/A	Per GB retrieved	Per GB retrieved	Per GB retrieved	Per GB retrieved
<b>Storage Type</b>	Object	Object	Object	Object	Object	Object
<b>Lifecycle Transitions</b>	Yes	Yes	Yes	Yes	Yes	Yes

S3 Storage Classes - Storage Costs		Storage Pricing
<b>S3 Standard</b> - General-purpose storage for any type of data, typically used for frequently accessed data		
First 50 TB / Month		\$0.023 per GB
Next 450 TB / Month		\$0.022 per GB
Over 500 TB / Month		\$0.021 per GB
<b>S3 Intelligent-Tiering*</b> - Automatic cost savings for data with unknown or changing access patterns		
Frequent Access Tier, First 50 TB / Month		\$0.023 per GB
Frequent Access Tier, Next 450 TB / Month		\$0.022 per GB
Frequent Access Tier, Over 500 TB / Month		\$0.021 per GB
Frequent Access Tier, All Storage / Month		\$0.0125 per GB
Monitoring and Automation, All Storage / Month		\$0.0025 per 1,000 objects
<b>S3 Standard-Infrequent Access</b> * - For long-lived but infrequently accessed data that needs millisecond access		
All Storage / Month		\$0.0125 per GB
<b>S3 One Zone-Infrequent Access</b> * - For re-createable infrequently accessed data that needs millisecond access		
All Storage / Month		\$0.01 per GB
<b>S3 Glacier</b> ** - For long-term backups and archives with retrieval options from 1 minute to 12 hours		
All Storage / Month		\$0.004 per GB
<b>S3 Glacier Deep Archive</b> ** - For long-term data archiving that is accessed once or twice in a year and can be restored within 12 hours		
All Storage / Month		\$0.00099 per GB

Storage Class	Availability and Durability	AZ(s)	Use Case
S3 Standard	99.99% Availability 11 9's Durability	>=3	Suitable for most workloads (e.g., websites, content distribution, mobile and gaming applications, and big data analytics)
S3 Standard-Infrequent Access	99.9% Availability 11 9's Durability	>=3	Long-term, infrequently accessed critical data (e.g., backups, data store for disaster recovery files, etc.)
S3 One Zone-Infrequent Access	99.5% Availability 11 9's Durability	1	Long-term, infrequently accessed, non-critical data
S3 Glacier	99.99% Availability 11 9's Durability	>=3	Long-term data archiving that occasionally needs to be accessed within a few hours or minutes
S3 Glacier Deep Archive	99.99% Availability 11 9's Durability	>=3	Rarely accessed data archiving with a default retrieval time of 12 hours (e.g., financial records for regulatory purposes)
S3 Intelligent-Tiering	99.9% Availability 11 9's Durability	>=3	Unknown or unpredictable access patterns

11 9's = 99.99999999%

# Lifecycle Management with S3

Tuesday, October 12, 2021 3:40 AM

## What Is Lifecycle Management?

Lifecycle management automates moving your objects between the different storage tiers, thereby maximizing cost effectiveness.



Keep for 30 Days



After 30 Days



After 90 Days

### VERSIONING

## Combining Lifecycle Management with Versioning

You can use lifecycle management to **move different versions** of objects to **different storage tiers**.



Automates moving objects between different storage tiers.



Can be used in conjunction with versioning.



Can be applied to current versions and previous versions.

## S3 Object Lock

You can use S3 Object Lock to store objects using a **write once, read many (WORM)** model. It can help prevent objects from being deleted or modified for a fixed amount of time or indefinitely.

You can use **S3 Object Lock** to meet regulatory requirements that require WORM storage, or add an extra layer of protection against object changes and deletion.

S3 OBJECT LOCK MODES

### Governance Mode

In governance mode, **users can't overwrite or delete an object version or alter its lock settings** unless they have special permissions.

With governance mode, you protect objects against being deleted by most users, but you can still grant some users **permission to alter the retention settings** or delete the object if necessary.

### Compliance Mode

In compliance mode, **a protected object version can't be overwritten or deleted by any user**, including the root user in your AWS account. When an object is locked in compliance mode, its retention mode can't be changed and its retention period can't be shortened. Compliance mode ensures an object version **can't be overwritten or deleted** for the duration of the retention period.

## Retention Periods

A retention period **protects an object version for a fixed amount of time**. When you place a retention period on an object version, Amazon S3 stores a timestamp in the object version's metadata to indicate when the retention period expires.

After the retention period expires, the object version can be **overwritten or deleted** unless you also placed a legal hold on the object version.

## Legal Holds

S3 Object Lock also enables you to place a legal hold on an object version. Like a retention period, a legal hold **protects an object version from being overwritten or deleted**. However, a legal hold doesn't have an associated retention period and remains in effect until removed. Legal holds can be freely placed and removed by any user who has the `s3:PutObjectLegalHold` permission.

## Glacier Vault Lock

S3 Glacier Vault Lock allows you to **easily deploy and enforce compliance controls for individual S3 Glacier vaults with a vault lock policy**. You can specify controls, such as WORM, in a vault lock policy and lock the policy from future edits. Once locked, the policy can no longer be changed.

## S3 Object Lock Modes

### 1 Compliance Mode

With **compliance mode**, a protected object version can't be overwritten or deleted by any user, including the root user in your AWS account.

### 2 Governance Mode

With **governance mode**, users can't overwrite or delete an object version or alter its lock settings unless they have special permissions.

Use **S3 Object Lock** to store objects using a write once, read many (WORM) model.

Object Lock can be on **individual objects** or applied **across the bucket** as a whole.

Object Lock comes in two modes: **governance mode** and **compliance mode**.

## S3 Glacier Vault Lock Exam Tips

S3 Glacier Vault Lock allows you to **easily deploy and enforce compliance controls** for individual S3 Glacier vaults with a vault lock policy.

You can **specify controls, such as WORM, in a vault lock policy and lock the policy from future edits**. Once locked, the policy can no longer be changed.

## Encrypting S3 Objects

Tuesday, October 12, 2021 4:45 AM

### Types of Encryption

- 1 **Encryption in Transit**
  - SSL/TLS
  - HTTPS
- 2 **Encryption at Rest: Server-Side Encryption**
  - SSE-S3: S3-managed keys, using AES 256-bit encryption
  - SSE-KMS: AWS Key Management Service-managed keys
  - SSE-C: Customer-provided keys
- 3 **Encryption at Rest: Client-Side Encryption**

You encrypt the files yourself before you upload them to S3.

```
PUT /myFile HTTP/1.1
Host: myBucket.s3.amazonaws.com
Date: Wed, 25 Nov 2020 09:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 27364
x-amz-meta-author: Ryan
Expect: 100-continue
[27364 bytes of object data]
```

### Enforcing Server-Side Encryption

Two Ways to Do It



**Console**  
Select the encryption setting on your S3 bucket. The easiest way is just a checkbox in the console.



**Bucket Policy**  
You can also enforce encryption using a bucket policy. This method sometimes comes up in the exam.

### Enforcing Server-Side Encryption

- 1 **x-amz-server-side-encryption**

If the file is to be encrypted at upload time, the **x-amz-server-side-encryption** parameter will be included in the request header.
- 2 **Two Options**

**x-amz-server-side-encryption: AES256** (SSE-S3 — S3-managed keys)

**x-amz-server-side-encryption: aws:kms** (SSE-KMS — KMS-managed keys)
- 3 **PUT Request Header**

When this parameter is included in the header of the PUT request, it tells S3 to encrypt the object at the time of upload, using the specified encryption method.

### S3 PUT REQUEST

```
PUT /myFile HTTP/1.1
Host: myBucket.s3.amazonaws.com
Date: Wed, 25 Nov 2020 09:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 27364
x-amz-meta-author: Ryan
Expect: 100-continue
x-amz-server-side-encryption: AES256
[27364 bytes of object data]
```

This request header tells S3 to encrypt the file using SSE-S3 (AES 256-bit) at the time of upload.

You can create a bucket policy that denies any S3 PUT request that doesn't include the **x-amz-server-side-encryption** parameter in the request header.

✓ **Encryption in Transit**

- SSL/TLS
- HTTPS

✓ **Client-Side Encryption**

You encrypt the files yourself before you upload them to S3.

✓ **Encryption at Rest: SSE**

- Server-side encryption
- SSE-S3 (AES 256-bit)
- SSE-KMS
- SSE-C

✓ **Enforcing Encryption with a Bucket Policy**

A bucket policy can deny all PUT requests that don't include the **x-amz-server-side-encryption** parameter in the request header.

## Optimizing S3 Performance

Tuesday, October 12, 2021 5:00 AM

### S3 Prefixes Explained

- ✓ mybucketname/folder1/subfolder1/myfile.jpg > /folder1/subfolder1
- ✓ mybucketname/folder2/subfolder1/myfile.jpg > /folder2/subfolder1
- ✓ mybucketname/folder3/myfile.jpg > /folder3

## S3 Performance

S3 has extremely low latency. You can get the first byte out of S3 within **100-200 milliseconds**.

You can also achieve a high number of requests: **3,500 PUT/COPY/POST/DELETE** and **5,500 GET/HEAD** requests per second, per prefix.

1 You can get better performance by spreading your reads across **different prefixes**. For example, if you are using **2 prefixes**, you can achieve **11,000 requests per second**.

2 If we used all **4 prefixes** in the last example, you would achieve **22,000 requests per second**.

### S3 LIMITATIONS WHEN USING KMS

- If you are using **SSE-KMS** to encrypt your objects in S3, you must keep in mind the **KMS limits**.
- When you **upload** a file, you will call `GenerateDataKey` in the KMS API.
- When you **download** a file, you will call `Decrypt` in the KMS API.

## KMS Request Rates

- ✓ Uploading/downloading will count toward the **KMS quota**.
- ✓ Region-specific, however, it's either **5,500, 10,000, or 30,000 requests per second**.
- ✓ Currently, you **cannot** request a quota increase for KMS.

### Multipart Uploads

- Recommended for files **over 100 MB**
- Required for files **over 5 GB**
- Parallelize uploads (increases **efficiency**)



## S3 Byte-Range Fetches



Can be used to **speed up** downloads



Can be used to download **partial amounts of the file** (e.g., header information)

## S3 Byte-Range Fetches

- Parallelize **downloads** by specifying byte ranges.
- If there's a failure in the download, it's only for a specific byte range.



mybucketname/folder1/subfolder1/myfile.jpg > /**Folder1/Subfolder1**



You can also achieve a high number of requests: **3,500 PUT/COPY/POST/DELETE** and **5,500 GET/HEAD** requests per second, per prefix.



You can get better performance by spreading your reads across **different prefixes**. For example, if you are using **2 prefixes**, you can achieve **11,000 requests per second**.

# Backing up Data With S3 Replication

Monday, October 18, 2021 6:53 PM

## S3 Replication

1

**You can replicate objects from one bucket to another.**

Versioning must be enabled on both the source and destination buckets.

2

**Objects in an existing bucket are not replicated automatically.**

Once replication is turned on, all subsequent updated objects will be replicated automatically.

3

**Delete markers are not replicated by default.**

Deleting individual versions or delete markers will not be replicated.

## Remember What S3 Replication Is



**Tip 1:** You can **replicate objects** from one bucket to another.



**Tip 2:** Objects in an existing bucket are **not replicated automatically**.



**Tip 3:** Delete markers are **not replicated by default**.

# Elastic Compute Cloud (EC2)

Secure, resizable compute capacity in the cloud.

Like a VM, only hosted in AWS instead of your own data center.

Designed to make web-scale cloud computing easier for developers.

**Game Changer**

AWS led a big change in the industry by introducing EC2.

**Pay Only for What You Use**

EC2 changed the economics of computing.

**No Wasted Capacity**

Select the capacity you need right now. Grow and shrink when you need.

**ON-PREMISES INFRASTRUCTURE**

## Estimate Capacity

Long-term investment, 3-5 years.

Expectation that the application will "grow into" it.  
Lots of wasted capacity.

**On-Demand**

Pay by the hour or the second, depending on the type of instance you run.

**Reserved**

Reserved capacity for 1 or 3 years. Up to 72% discount on the hourly charge.

**Spot**

Purchase unused capacity at a discount of up to 90%. Prices fluctuate with supply and demand.

**Dedicated**

A physical EC2 server dedicated for your use. The most expensive option.

1

**Flexible**

Low cost and flexibility of Amazon EC2 without any upfront payment or long-term commitment.

2

**Short-Term**

Applications with short-term, spiky, or unpredictable workloads that cannot be interrupted.

3

**Testing the Water**

Applications being developed or tested on Amazon EC2 for the first time.

## Reserved Instances

**Predictable Usage**

Applications with steady state or predictable usage.

**Standard RIs**

Up to 72% off the on-demand price.

**Specific Capacity Requirements**

Applications that require reserved capacity.

**Convertible RIs**

Up to 54% off the on-demand price. Has the option to change to a different RI type of equal or greater value.

**Pay up Front**

You can make upfront payments to reduce the total computing costs even further.

**Scheduled RIs**

Launch within the time window you define. Match your capacity reservation to a predictable recurring schedule that only requires a fraction of a day, week, or month.

## Savings Plans with Reserved Instances

### 1 Save up to 72%

All AWS compute usage, regardless of instance type or Region.

### 2 Commit to 1 or 3 Years

Commit to use a specific amount of compute power (measured by the hour) for a 1-year or 3-year period.

### 3 Super Flexible

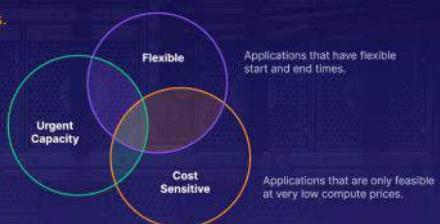
Not only EC2, this also includes serverless technologies like Lambda and Fargate.

## EC2 Pricing Options

### When to Use Spot Instances

Image rendering.  
Genomic sequencing.  
Algorithmic trading engines.

Users with an urgent need for large amounts of additional computing capacity.



## Dedicated Hosts



### Compliance

Regulatory requirements that may not support multi-tenant virtualization.



### On-Demand

Can be purchased on-demand (hourly).



### Licensing

Great for licensing that does not support multi-tenancy or cloud deployments.



### Reserved

Can be purchased as a reservation for up to 70% off the on-demand price.

# AWS CLI

Tuesday, October 19, 2021 1:51 AM

```
# aws configure  
  
# aws s3 ls  
  
# aws s3 mb s3://abcdefghijklm  
  
# echo "Hello" > hello.txt  
  
# aws s3 cp hello.txt s3://abcdefghijklm
```

1

## Secret Access Key

You will only see this once! If you lose it, you can delete the access key ID and secret access key and regenerate them. You will need to run **aws configure** again.

2

## Don't Share Key Pairs

Each developer should have their own access key ID and secret access key. Just like passwords, they should not be shared.

3

## Supports Linux, Windows, MacOS

You can install the CLI on your Mac, Linux, or Windows PC. You can also use it on EC2 instances.

## Using Roles

Tuesday, October 19, 2021 3:10 AM

# What Is an IAM Role?

A role is an identity you can create in IAM that has specific permissions. A role is similar to a user, as it is an AWS identity with permission policies that determine what the identity can and cannot do in AWS.

However, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it.

# What Else Can Roles Do?

Roles can be assumed by people, AWS architecture, or other system-level accounts.

**Roles can allow cross-account access. This allows one AWS account the ability to interact with resources in other AWS accounts.**



### Create an IAM Role

Ensure it has S3 access.



### Create an EC2 Instance

Attach the role we just created.



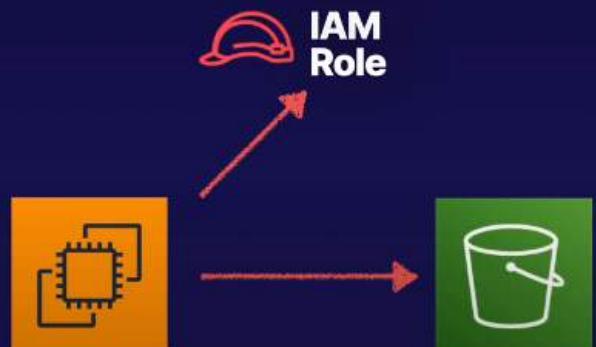
### Create S3 Bucket

Create a bucket in S3.



### Access S3

Try to access S3 from our EC2 instance.



## What to Remember When Using Roles



### The Preferred Option

Roles are preferred from a security perspective.



### Updates

You can update a policy attached to a role, and it will take immediate effect.



### Avoid Hard-Coding Your Credentials

Roles allow you to provide access without the use of access key IDs and secret access keys.



### Attaching and Detaching

You can attach and detach roles to running EC2 instances without having to stop or terminate those instances.



### Policies

Policies control a role's permissions.

## Security Groups and Bootstrap Scripts

Tuesday, October 19, 2021 3:33 AM



## Security Groups

Security groups are **virtual firewalls for your EC2 instance**. By default, everything is blocked.

TO LET EVERYTHING IN: 0.0.0.0/0

In order to be able to **communicate to your EC2 instances via SSH/RDP/HTTP**, you will need to **open up the correct ports**.

## Bootstrap Scripts

A script that runs when the instance first runs

```
#!/bin/bash
yum install httpd -y
#installs apache
yum service httpd start
#starts apache
```

Adding these tasks at boot time adds to the amount of time it takes to boot the instance. However, it allows you to automate the installation of applications.

▼ Advanced Details

Enclave	<input type="checkbox"/> Enable
Metadata accessible	Enabled
Metadata version	V1 and V2 (token optional)
Metadata token response hop limit	1
User data	<input checked="" type="radio"/> As text <input type="radio"/> As file <input type="checkbox"/> Input is already base64 encoded
<pre>#!/bin/bash yum update -y yum install httpd -y service httpd start cd /var/www/html echo "&lt;html&gt;&lt;body&gt;&lt;h1&gt;Hello Cloud Gurus&lt;/h1&gt;&lt;/body&gt;&lt;/html&gt;" &gt;</pre>	

Automate deployment of the server

## Security Groups Exam Tips

- ✓ **Tip 1:** Changes to security groups take effect immediately.
- ✓ **Tip 2:** You can have any number of EC2 instances within a security group.
- ✓ **Tip 3:** You can have multiple security groups attached to EC2 instances.
- ✓ **Tip 4:** All inbound traffic is blocked by default.
- ✓ **Tip 5:** All outbound traffic is allowed.

## EC2 Metadata and User Data

Tuesday, October 19, 2021 4:04 AM

WHAT IS EC2 METADATA?

# EC2 metadata is simply data about your EC2 instance.

This can include information such as private IP address, public IP address, hostname, security groups, etc.

- User data is simply bootstrap scripts.
- Metadata is data about your EC2 instances.
- You can use bootstrap scripts (user data) to access metadata.

### Using User Data to Save Metadata

```
#!/bin/bash
curl http://169.254.169.254/latest/meta-data/local-ipv4 > myIP.txt
```

Combining User Data and Metadata

In this simple bootstrap (user data) script, we use the curl command to save our EC2 metadata.

EC2 METADATA AND USER DATA

### Retrieving Metadata

```
curl http://169.254.169.254/latest/meta-data/local-ipv4
[ec2-user@ip-172-31-31-8 ~]$ curl http://169.254.169.254/latest/meta-
data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hibernation/
http-
identity-credentials/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
managed-ssh-keys/
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
```

Retrieving Metadata

Using the curl command, we can query metadata about our EC2 instance.

```
yum install httpd -y
service httpd start
cd /var/www/html
echo "<html><body><h1>My IP is" > index.html
curl http://169.254.169.254/latest/meta-data/public-ipv4 >> index.html
echo "</h1></body></html>" >> index.html
```

## Networking with EC2

Tuesday, October 19, 2021 5:45 PM

## Networking with EC2

You can attach 3 different types of **virtual networking cards** to your EC2 instances

**ENI**  
Elastic Network Interface  
For basic, day-to-day networking

**EN**  
Enhanced Networking  
Uses single root I/O virtualization (SR-IOV) to provide high performance

**EFA**  
Elastic Fabric Adapter  
Accelerates High Performance Computing (HPC) and machine learning applications

## Common ENI Use Cases

- ✓ Create a management network.
- ✓ Use network and security appliances in your VPC.
- ✓ Create dual-homed instances with workloads/roles on distinct subnets.
- ✓ Create a low-budget, high-availability solution.

### An ENI is simply a virtual network card that allows:

- Private IPv4 Addresses
- Public IPv4 Address
- Many IPv6 Addresses
- MAC Address
- 1 or More Security Groups

#### What Is Enhanced Networking?

### For High-Performance Networking between 10 Gbps - 100 Gbps

**SINGLE ROOT I/O VIRTUALIZATION (SR-IOV)**  
SR-IOV provides higher I/O performance and lower CPU utilization

**PERFORMANCE**  
Provides higher bandwidth, higher packet per second (PPS) performance, and consistently lower inter-instance latencies

#### Depending on your instance type, enhanced networking can be enabled using:

**ELASTIC NETWORK ADAPTER (ENA)** OR **INTEL 82599 VIRTUAL FUNCTION (VF) INTERFACE**

Supports network speeds of up to 100 Gbps for supported instance types. Typically used on older instances.

### What Is an EFA?

Elastic Fabric Adapter

- A network device you can attach to your Amazon EC2 instance to accelerate High Performance Computing (HPC) and machine learning applications.
- Provides lower and more consistent latency and higher throughput than the TCP transport traditionally used in cloud-based HPC systems.



**EFA CAN USE OS-BYPASS**

### It makes it a lot faster with much lower latency.

OS-bypass enables HPC and machine learning applications to bypass the operating system kernel and communicate directly with the EFA device. Not currently supported with Windows — only Linux.

## For different scenarios on the exam, choose the correct networking device.

### 1 ENI

For basic networking. Perhaps you need a separate management network from your production network or a separate logging network, and you need to do this at a low cost. In this scenario, use multiple ENIs for each network.

### 2 Enhanced Networking

For when you need speeds between 10 Gbps and 100 Gbps. Anywhere you need reliable, high throughput.

### 3 EFA

For when you need to accelerate High Performance Computing (HPC) and machine learning applications or if you need to do an OS-bypass. If you see a scenario question mentioning HPC or ML and asking what network adapter you want, choose EFA.



- ✓ A **cluster placement group** can't span multiple Availability Zones, whereas a spread placement group and partition placement group can.
- ✓ Only **certain types of instances** can be launched in a placement group (compute optimized, GPU, memory optimized, storage optimized).
- ✓ AWS recommends **homogenous instances** within cluster placement groups.
- ✓ You can't merge placement groups.
- ✓ You can **move an existing instance into a placement group**. Before you move the instance, the instance must be in the stopped state. You can move or remove an instance using the AWS CLI or an AWS SDK, but you can't do it via the console yet.

## Cluster Placement Groups

**Grouping of instances within a single Availability Zone.** Recommended for applications that need low network latency, high network throughput, or both.

**Fact:**  
Only certain instance types can be launched into a cluster placement group.

## Spread Placement Groups

A spread placement group is a group of instances that are **each placed on distinct underlying hardware**.

Spread placement groups are recommended for applications that have a small number of critical instances that should be kept separate from each other.

**STUDY TIP**  
**Scenario Questions**  
Used for Individual Instances

## Partition Placement Groups

Each partition placement group has its own set of racks. Each rack has its own network and power source. No two partitions within a placement group share the same racks, allowing you to isolate the impact of hardware failure within your application.

**EC2 DIVIDES EACH GROUP INTO LOGICAL SEGMENTS CALLED PARTITIONS.**

**Partition 1**

**Partition 2**

**Partition 3**

# 3 Types of Placement Groups

	<b>Cluster Placement Groups</b> Low network latency, high network throughput
	<b>Spread Placement Groups</b> Individual critical EC2 instances
	<b>Partition Placement Groups</b> Multiple EC2 instances; HDFS, HBase, and Cassandra

## Solving Licensing Issues with Dedicated Hosts

Tuesday, October 19, 2021 5:58 PM



# Any question that talks about special licensing requirements.

An **Amazon EC2 Dedicated Host** is a **physical server** with EC2 instance capacity fully dedicated to your use. Dedicated Hosts allow you to **use your existing** per-socket, per-core, or per-VM software **licenses**, including Windows Server, Microsoft SQL Server, and SUSE Linux Enterprise Server.

# Stateless, fault-tolerant, or flexible applications

Applications such as big data, containerized workloads, CI/CD, high-performance computing (HPC), and other test and development workloads.

To use **Spot Instances**, you must first decide on your maximum Spot price. The instance will be provisioned so long as the Spot price is **BELOW** your maximum Spot price.



The hourly Spot price varies depending on capacity and region.



If the Spot price goes above your maximum, you have 2 minutes to choose whether to stop or terminate your instance.



You may also use a **Spot block** to stop your Spot Instances from being terminated even if the Spot price goes over your max Spot price. You can set Spot blocks for between **1 to 6 hours** currently.

## Spot Instances are useful for the following tasks:



**Big data and analytics**



**Containerized workloads**



**CI/CD and testing**



**Image and media rendering**

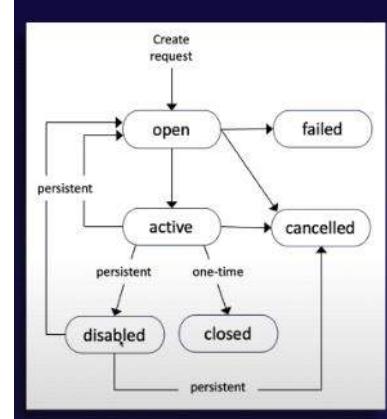
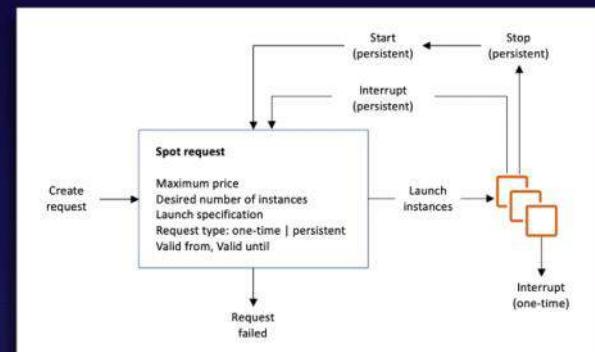


**High-performance computing**

## Spot Instances are **not** good for:

- ✓ Persistent workloads
- ✓ Critical jobs
- ✓ Databases

## Terminating Spot Instances



## Spot Fleets

A Spot Fleet is a collection of Spot Instances and (optionally) On-Demand Instances.

The **Spot Fleet** attempts to launch the number of Spot instances and On-Demand instances to meet the target capacity you specified in the Spot Fleet request. The request for Spot Instances is fulfilled if there is available capacity and the **maximum price you specified in the request exceeds the current Spot price**. The Spot Fleet also attempts to maintain its target capacity fleet if your Spot Instances are interrupted.

Spot Fleets will try and match the target capacity with your price restraints.



1 Set up different launch pools. Define things like **EC2** instance type, operating system, and Availability Zone.

2 You can have **multiple** pools, and the fleet will choose the best way to implement depending on the strategy you define.

3 Spot fleets will **stop launching instances** once you reach your price threshold or capacity desire.

## You can have the following strategies with Spot Fleets.

### ✓ capacityOptimized

The Spot Instances come from the pool with optimal capacity for the number of instances launching.

### ✓ diversified

The Spot Instances are distributed across all pools.

### ✓ lowestPrice

The Spot Instances come from the pool with the lowest price. This is the default strategy.

### ✓ InstancePoolsToUseCount

The Spot Instances are distributed across the number of Spot Instance pools you specify. This parameter is valid only when used in combination with `lowestPrice`.



Spot Instances save up to **90%** of the cost of On-Demand instances.



Useful for any type of computing where you don't need **persistent storage**.



You can block Spot instances from terminating by using a **Spot block**.



A Spot Fleet is a collection of Spot Instances and (optionally) On-Demand instances.

## EC2 Overview

**EC2 is like a VM, hosted in AWS instead of your own data center.**

Select the capacity you need **right now**. Grow and shrink when you need. Pay for what you use. Wait minutes, not months.

## EC2 Instance Pricing Options

- On-Demand**: Pay by the hour or the second, depending on the type of instance you run. Great for flexibility.
- Spot**: Purchase unused capacity at a discount of up to 90%. Prices fluctuate with supply and demand. Great for applications with flexible start and end times.
- Reserved**: Reserved capacity for 1 or 3 years. Up to 72% discount on the hourly charge. Great if you have known, fixed requirements.
- Dedicated**: A physical EC2 server dedicated for your use. Great if you have server-bound licenses to reuse or compliance requirements.

## AWS Command Line Interface

**Least Privilege**

Always give your users the **minimum amount** of access required to do their job.

**Use Groups**

Create **IAM groups** and assign your users to groups. Group permissions are assigned using IAM policy documents. Your users will **automatically inherit** the permissions of the group.

- 1 Secret Access Key**: You will only see this once! If you lose it, you can delete the access key ID and secret access key and regenerate them. You will need to run `aws configure` again.
- 2 Don't Share Key Pairs**: Each developer should have their own access key ID and secret access key. Just like passwords, they should not be shared.
- 3 Supports Linux, Windows, and macOS**: You can install the CLI on your Mac, Linux, or Windows PC. You can also use it on EC2 instances.

## Tips to Remember When Using Roles

- 1 The Preferred Option**: Roles are preferred from a security perspective.
- 2 Avoid Hard Coding Your Credentials**: Roles allow you to provide access without the use of access key IDs and secret access keys.
- 3 Policies**: Policies control a role's permissions.
- 4 Updates**: You can update a policy attached to a role, and it will take immediate effect.
- 5 Attaching and Detaching**: You can attach and detach roles to running EC2 instances without having to stop or terminate these instances.

## Security Groups Exam Tips

- Tip 1:** Changes to security groups take effect immediately.
- Tip 2:** You can have any number of EC2 instances within a security group.
- Tip 3:** You can have multiple security groups attached to EC2 instances.
- Tip 4:** All inbound traffic is blocked by default.
- Tip 5:** All outbound traffic is allowed.

### STUDY TIP

## Bootstrap Scripts

A bootstrap script is a **script that runs when the instance first runs**. It passes user data to the EC2 instance and can be used to install applications (like web servers and databases), as well as do updates and more.

## Comparing User Data and Metadata

### User Data

✓ **User data** are simply bootstrap scripts

### Metadata

✓ **Metadata** is data about your EC2 instances

✓ You can use bootstrap scripts (user data) to access **metadata**

## For different scenarios on the exam, choose the correct networking device.

- 1 ENI**: For basic networking. Perhaps you need a separate management network from your production network or a separate logging network, and you need to do this at a low cost. In this scenario, use multiple ENIs for each network.
- 2 Enhanced Networking**: For when you need speeds between 10 Gbps and 100 Gbps. Anywhere you need reliable, high throughput.
- 3 EFA**: For when you need to accelerate High Performance Computing (HPC) and machine learning applications or if you need to do an OS-bypass. If you see a scenario question mentioning HPC or ML and asking what network adapter you want, choose EFA.

### Cluster Placement Groups

Low network latency, high network throughput

### Spread Placement Groups

Individual critical EC2 instances

### Partition Placement Groups

Multiple EC2 instances; HDFS, HBase, and Cassandra

#### REMEMBER THESE TIPS FOR THE EXAM:

- ✓ A **cluster placement group** can't span multiple Availability Zones, whereas a spread placement group and partition placement group can.
- ✓ Only **certain types of instances** can be launched in a placement group (compute optimized, GPU, memory optimized, storage optimized).
- ✓ AWS recommends **homogenous instances** within cluster placement groups.
- ✓ You can't merge placement groups.
- ✓ You can **move an existing instance into a placement group**. Before you move the instance, the instance must be in the stopped state. You can move or remove an instance using the AWS CLI or an AWS SDK, but you can't do it via the console yet.

## Dedicated Hosts

Any question that talks about special licensing requirements.

An **Amazon EC2 Dedicated Host** is a **physical server** with EC2 instance capacity fully dedicated to your use. Dedicated Hosts allow you to use your **existing** per-socket, per-core, or per-VM **software licenses**, including Windows Server, Microsoft SQL Server, and SUSE Linux Enterprise Server.

## Spot Instance and Spot Fleet Tips



Spot Instances save up to **90%** of the cost of On-Demand Instances.



Useful for any type of computing where you **don't need persistent storage**.



You can block Spot Instances from terminating by using **Spot block**.



A Spot Fleet is a **collection of Spot Instances** and, optionally, On-Demand Instances.

## VPC - Overview

Tuesday, October 19, 2021 6:37 PM

# What Is a VPC?

Think of a VPC as a virtual data center in the cloud.

- ✓ Logically isolated part of AWS Cloud where you can define your own network.
- ✓ Complete control of virtual network, including your own IP address range, subnets, route tables, and network gateways.

## Fully Customizable Network

You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances in each subnet.

### Web

Public-facing subnet.

### Application

Private subnet. Can only speak to web tier and database tier.

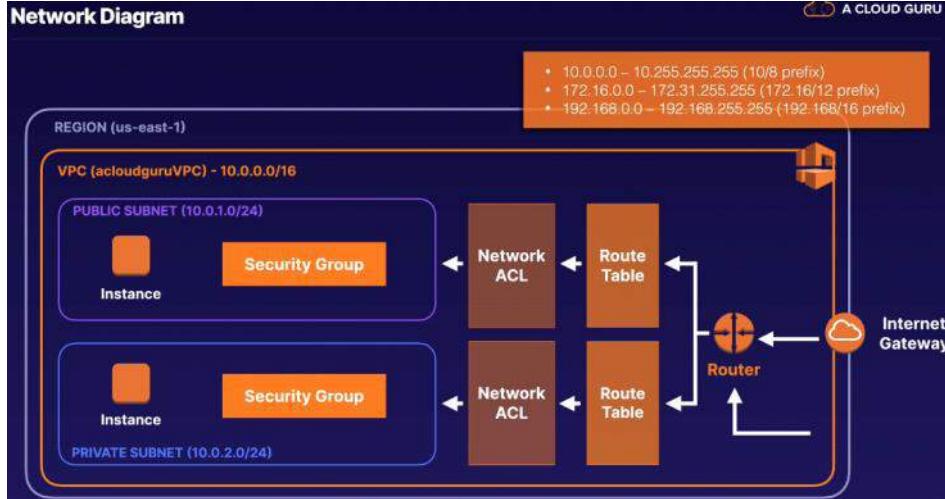
### Database

Private subnet. Can only speak to application tier.

Additionally, you can create a **hardware Virtual Private Network (VPN)** connection between your corporate data center and your VPC and leverage the AWS Cloud as an extension of your corporate data center.



## Network Diagram



## What can we do with a VPC?

**Launch Instances**  
Launch instances into a subnet of your choosing.

**Internet Gateway**  
Create internet gateway and attach it to our VPC.

### BONUS TIP

You can use network access control lists (**NACLs**) to block specific IP addresses.

**Custom IP Addresses**  
Assign custom IP address ranges in each subnet.

**More Control**  
Much better security control over your AWS resources.

**Route Tables**  
Configure route tables between subnets.

**Access Control Lists**  
Subnet network access control lists.

## Default VPC vs. Custom VPC

### Default

- Default VPC is user friendly.
- All subnets in default VPC have a route out to the internet.
- Each EC2 instance has both a public and private IP address.

### Custom

- Fully customizable.
- Takes time to set up.

## VPC Exam Tips

- ✓ Think of a VPC as a logical data center in AWS.
- ✓ Consists of internet gateways (or virtual private gateways), route tables, network access control lists, subnets, and security groups.
- ✓ 1 subnet is always in 1 Availability Zone.

# Building VPC Lab

Tuesday, October 19, 2021 10:14 PM

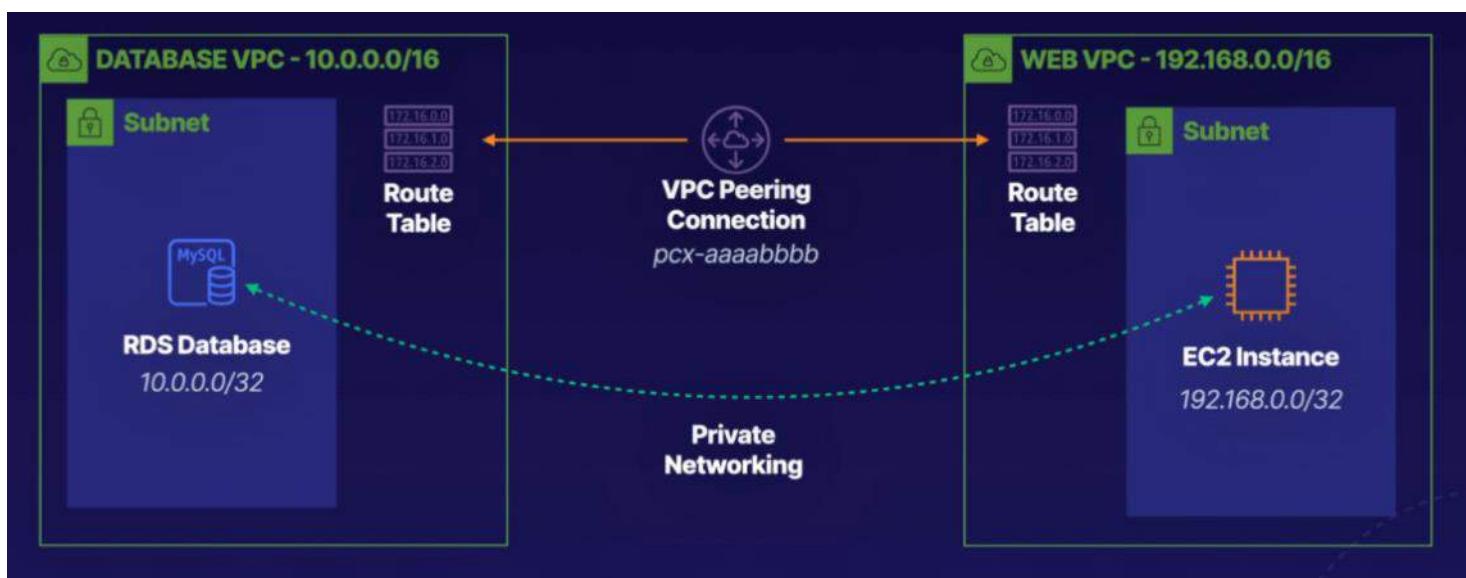
## Build Solutions across VPCs with Peering

### Introduction

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses. In this lab, we create a new VPC for our WordPress blog to run from. We then create a VPC peering connection between the new VPC and an existing database VPC. By the end of this lab, the user will understand how to create a new VPC from scratch, attach internet gateways, edit routing tables, and peer multiple VPCs together.

### Solution

Log in to the AWS Management Console using the credentials provided on the lab instructions page. Make sure you're in the N. Virginia (**us-east-1**) region throughout the lab.



Create Web\_VPC Subnets and Attach a New Internet Gateway

### Create a VPC

1. Navigate to VPC.
2. Under *Resources by Region*, click **VPCs**.
3. Click **Create VPC**.
4. Set the following values:
  - **Name tag:** **Web\_VPC**
  - **IPv4 CIDR block:** **192.168.0.0/16**
5. Leave the rest as their defaults and click **Create VPC**.

### Create a Subnet

3. On the left menu under *VIRTUAL PRIVATE CLOUD*, select **Subnets**.
4. Click **Create subnet**.
5. Select the newly created **Web\_VPC**.
6. Under *Subnet settings*, set the following values:
  - **Subnet name:** **WebPublic**
  - **Availability Zone:** **us-east-1a**
  - **IPv4 CIDR (Classless inter-domain routing) block:** **192.168.0.0/24**

7. Click **Create subnet**.

### Create an Internet Gateway

9. On the left menu, select **Internet Gateways**.
10. Click **Create internet gateway**.
11. In *Name tag*, enter "WebIG".
12. Click **Create internet gateway**.
13. In the green notification at the top of the page, click **Attach to a VPC**.
14. In *Available VPCs*, select the **Web\_VPC** and click **Attach internet gateway**.
15. On the left menu, select **Route Tables**.
16. Select the **Web\_VPC**.
17. Underneath, select the *Routes* tab and click **Edit routes**.
18. Click **Add route**.
19. Set the following values:
  - Destination*: **0.0.0.0/0**
  - Target*: **Internet Gateway > WebIG**
20. Click **Save changes**.

### Create a Peering Connection

1. On the left menu, select **Peering Connections**.
2. Click **Create peering connection**.
3. Set the following values:
  - Name*: **DBtoWeb**
  - VPC (Requester)*: **DB\_VPC**
  - VPC (Acceptor)*: **Web\_VPC**
4. Click **Create peering connection**.
5. At the top of the page, click **Actions > Accept Request**.
6. Click **Accept request**.
7. On the left menu, select **Route Tables**.
8. Select the **Web\_VPC**.
9. Underneath, select the *Routes* tab and click **Edit routes**.
10. Click **Add route**.
11. Set the following values:
  - Destination*: **10.0.0.0/16**
  - Target*: **Peering Connection > DBtoWeb**
12. Click **Save changes**.
13. Go back to *Route Tables* and select the **DB\_VPC** instance with a *Main* column value of Yes.
14. Under the *Routes* tab, click **Edit routes**.
15. Click **Add route**.
16. Set the following values:
  - Destination*: **192.168.0.0/16**
  - Target*: **Peering Connection > DBtoWeb**
17. Click **Save changes**.

### Create an EC2 Instance and Configure WordPress

1. In a new browser tab, navigate to EC2.
2. Click **Launch instance > Launch instance**.
3. Scroll down to *Ubuntu Server 20.04 LTS* and click **Select**.

4. Select **t3.micro** as the instance type.
5. Click **Next: Configure Instance Details**.
6. Set the following values:
  - **Network: Web\_VPC**
  - **Auto-assign Public IP: Enable**

7. At the bottom under *User data*, paste in the following bootstrap script:

```
#!/bin/bash
sudo apt update -y
sudo apt install php-curl php-gd php-mbstring php-xml php-xmlrpc php-soap php-intl php-zip perl
mysql-server apache2 libapache2-mod-php php-mysql -y
wget https://github.com/ACloudGuru-Resources/course-aws-certified-solutions-architect-
associate/raw/main/lab/5/wordpress.tar.gz
tar zxvf wordpress.tar.gz
cd wordpress
wget https://raw.githubusercontent.com/ACloudGuru-Resources/course-aws-certified-solutions-
architect-associate/main/lab/5/000-default.conf
cp wp-config-sample.php wp-config.php
perl -pi -e "s/database_name_here/wordpress/g" wp-config.php
perl -pi -e "s/username_here/wordpress/g" wp-config.php
perl -pi -e "s/password_here/wordpress/g" wp-config.php
perl -i -pe' BEGIN {
    @chars = ("a" .. "z", "A" .. "Z", 0 .. 9);
    push @chars, split //, "!@#$%^&*()-_ []{}<>~\`+=,.:/?|";
    sub salt { join "", map $chars[ rand @chars ], 1 .. 64 }
}
s/put your unique phrase here/salt()/ge
' wp-config.php
mkdir wp-content/uploads
chmod 775 wp-content/uploads
mv 000-default.conf /etc/apache2/sites-enabled/
mv /wordpress /var/www/
apache2ctl restart
```

8. Click **Review and Launch**.
9. Scroll down to *Security Groups* and click **Edit security groups**.
10. Click **Add Rule**.
11. Select **HTTP** and click **Review and Launch**.
12. Click **Launch**.
13. On the dropdown, select **Proceed without a key pair**.
14. Select the checkbox acknowledging that you will not be able to connect to this instance unless you already know the password built into this AMI.
15. Click **Launch Instances** and then click **View Instances**.  
**Note:** It may take a few minutes for the new instance to launch.
16. In a new browser tab, navigate to RDS.
17. Select the provisioned RDS instance.
18. Under *Connectivity & security*, copy the RDS endpoint for later use.
19. Navigate back to EC2.

20. Select the new instance and click **Connect**.
21. Click **Connect**.
22. To confirm WordPress installed correctly, view the configuration files:  

```
cd /var/www/wordpress  
ls
```
23. To configure WordPress, open **wp-config.php**:  

```
sudo nano wp-config.php
```
24. Scroll down to **/\*\* MySQL hostname \*/** and replace **localhost** with the RDS endpoint previously copied.
25. To save, press **Ctrl+X**, and then type **Y** and press **Enter**.

### Modify the RDS Security Groups to Allow Connections from the Web\_VPC VPC

1. Navigate to RDS.
2. In *Connectivity & security*, click the active link under *VPC security groups*.
3. Select the *Inbound rules* tab and click **Edit inbound rules**.
4. Click **Add rule**.
5. Under *Type*, type and select **MySQL/Aurora**.
6. Under *Source*, type and select **192.168.0.0/16**.
7. Click **Save rules**.
8. Return to the terminal.
9. At the bottom of the terminal window, copy the public IP address of our server.
10. Open a new browser tab and paste the public IP address in the address bar. You should now see the WordPress installation page.
11. Set the the following values :
  - *Site Title*: **A Blog Guru**
  - *Username*: **guru**
  - *Your Email*: Your email address
12. Click **Install WordPress**.
13. Reload the public IP address in the address bar to view our newly created WordPress blog.

### Conclusion

Congratulations — you've completed this hands-on lab!

### Troubleshooting

If the website isn't loading the way you'd expected at the end of this guide, here are some tips to help troubleshoot:

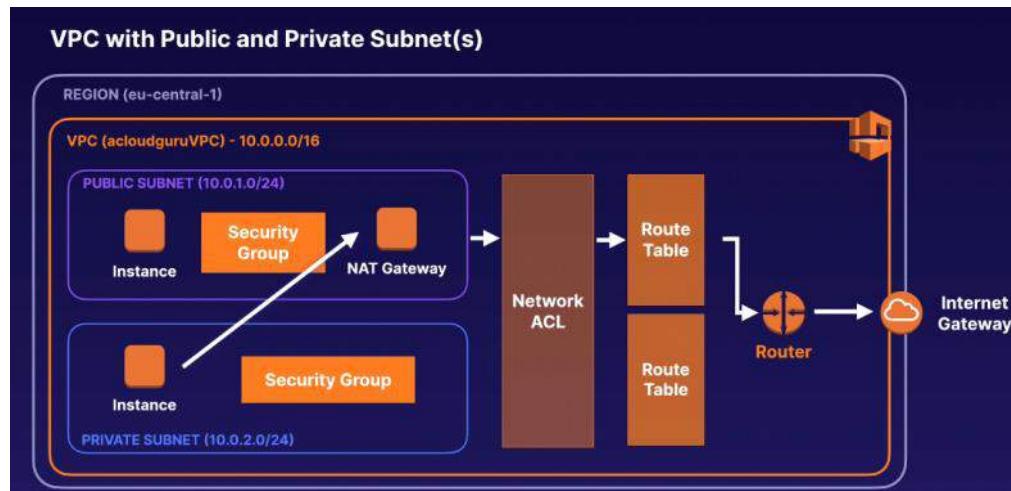
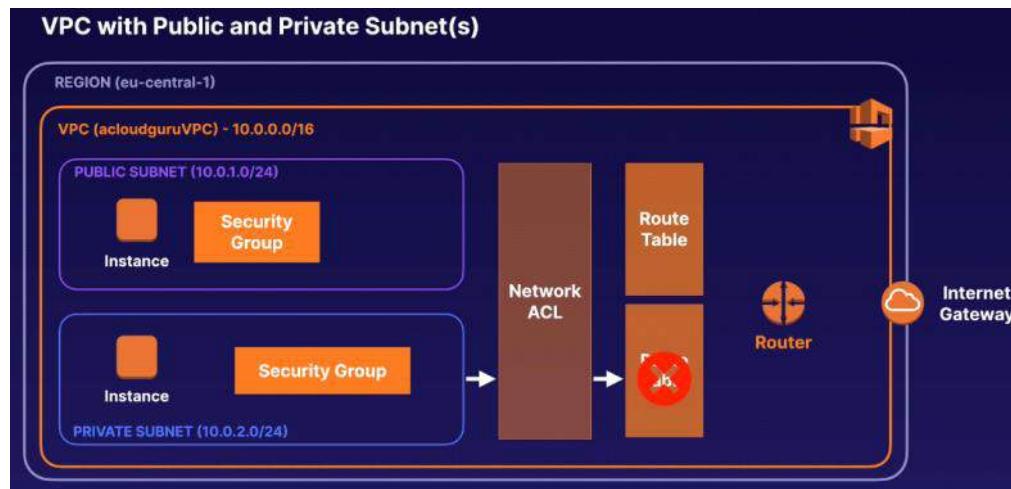
- Check the status of the lab objectives - are any not yet completed?
- Has everything we've setup successfully become ready to use? Check things like the VPC Peering Connection, which requires you to specifically accept the connection request.
- Does the database error page load after a minute or so of waiting, or does no page load at all? This gives a hint to whether the issue may be with the peering, or the security groups.

# NAT - Using NAT Gateways for Internet Access

Tuesday, October 19, 2021 11:23 PM



- ✓ Redundant inside the Availability Zone
- ✓ Starts at 5 Gbps and scales currently to 45 Gbps
- ✓ No need to patch
- ✓ Not associated with security groups
- ✓ Automatically assigned a public IP address



# Security Groups -Protecting Your Resources

Tuesday, October 19, 2021 11:35 PM



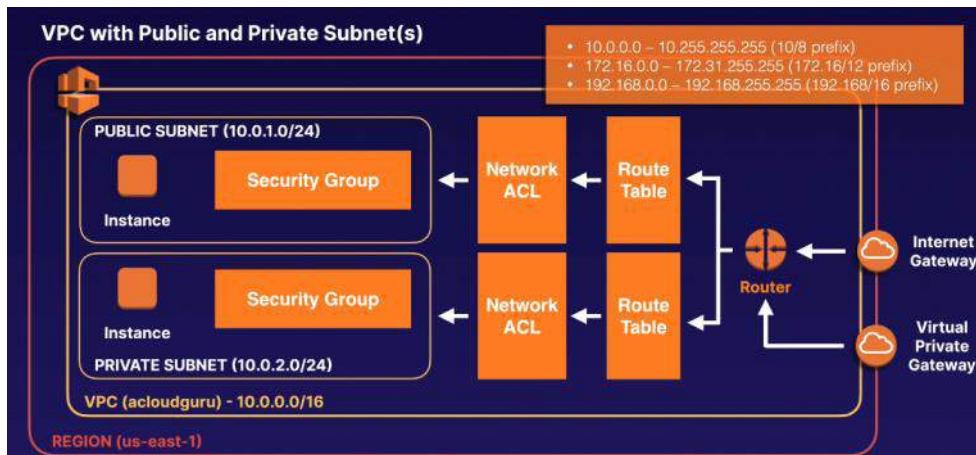
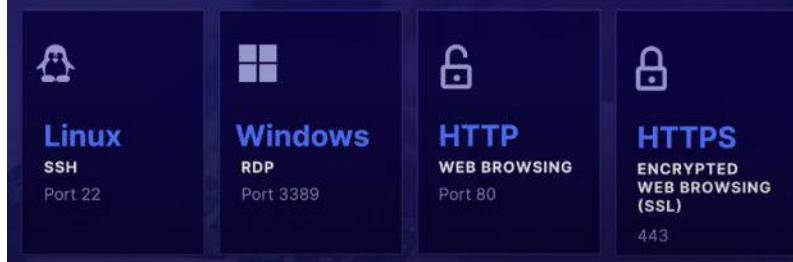
## Security Groups

Security groups are **virtual firewalls** for an EC2 instance. By default, everything is blocked.

TO LET EVERYTHING IN: 0.0.0.0/0

In order to communicate to your EC2 instances via SSH, RDP, or HTTP, you will need to open up the correct ports.

# How Computers Communicate



## Security Groups

Security groups are stateful — if you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules.

**Responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules.**

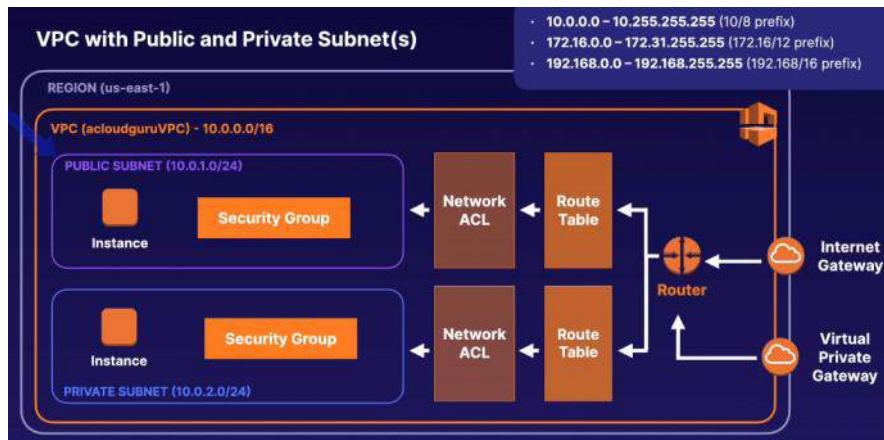
## ACLs - Controlling Subnet Traffic with Network ACLs

Tuesday, October 19, 2021 11:38 PM

# Network ACLs

## The first line of defense

- ✓ A network access control list (ACL) is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.
- ✓ You might set up network ACLs with rules similar to your security groups in order to add another layer of security to your VPC.



## Overview of Network ACLs

- ✓ **Default Network ACLs:** Your VPC automatically comes with a default network ACL, and by default it allows all outbound and inbound traffic.
- ✓ **Custom Network ACLs:** You can create custom network ACLs. By default, each custom network ACL denies all inbound and outbound traffic until you add rules.
- ✓ **Subnet Associations:** Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a network ACL, the subnet is automatically associated with the default network ACL.
- ✓ **Block IP Addresses:** Block IP addresses using network ACLs, not security groups.

- ✓ You can associate a network ACL with multiple subnets; however, a subnet can be associated with **only 1 network ACL** at a time. When you associate a network ACL with a subnet, the previous association is **removed**.
- ✓ Network ACLs contain a **numbered list of rules** that are evaluated in order, starting with the **lowest** numbered rule.
- ✓ Network ACLs have **separate** inbound and outbound rules, and each rule can either **allow or deny traffic**.
- ✓ Network ACLs are **stateless**; responses to allowed inbound traffic are subject to the rules for outbound traffic (and vice versa).

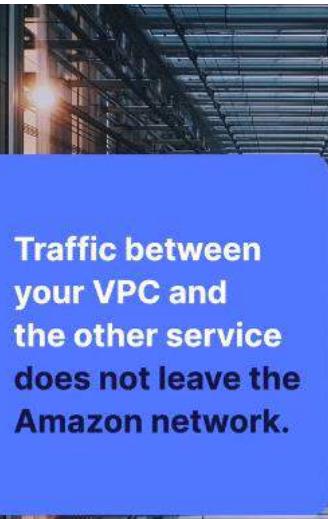
# Endpoint - Private Communication Using VPC Endpoints

Wednesday, October 20, 2021 12:13 AM

## VPC Endpoints

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection.

**Instances in your VPC do not require public IP addresses to communicate with resources in the service.**



### 💡 STUDY TIP

## Endpoints Are Virtual Devices

They are horizontally scaled, redundant, and highly available VPC components that allow communication between instances in your VPC and services **without imposing availability risks or bandwidth constraints** on your network traffic.

## There are **2** types of endpoints

### OPTION 1

#### INTERFACE ENDPOINTS

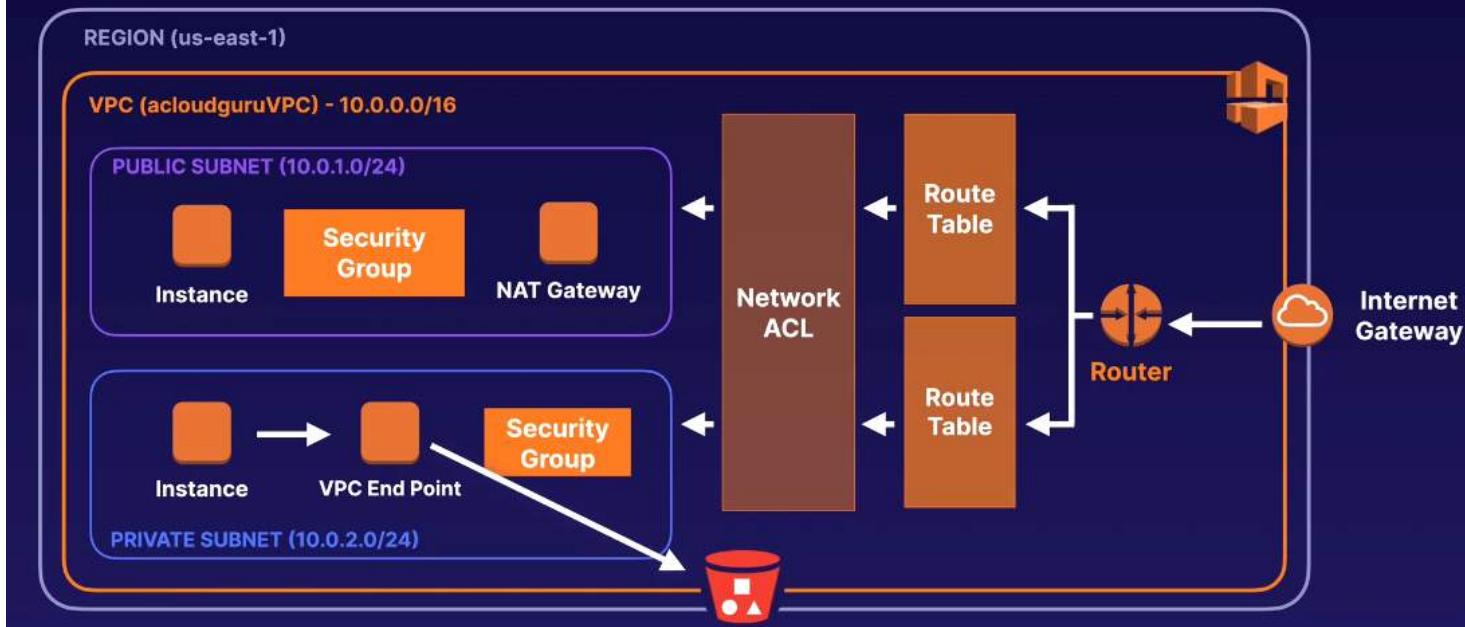
An interface endpoint is an **elastic network interface with a private IP address** that serves as an entry point for traffic headed to a supported service. They support a large number of AWS services.

### OPTION 2

#### GATEWAY ENDPOINTS

Similar to NAT gateways, a gateway endpoint is a **virtual device you provision**. It supports connection to S3 and DynamoDB.

## VPC with Public and Private Subnet(s)



## VPC Endpoints

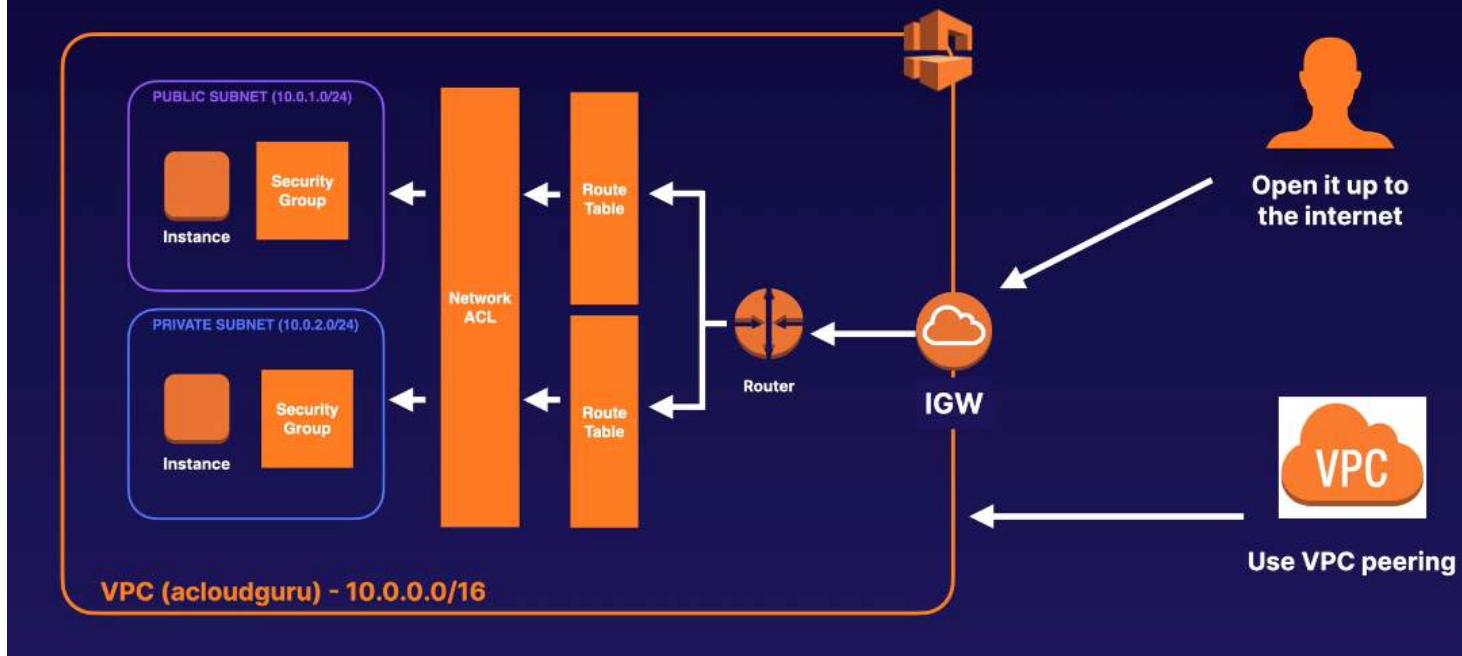
- ✓ **Use Case:** When you want to connect AWS services without leaving the Amazon internal network
- ✓ **2 Types of VPC Endpoints:** Interface endpoints and gateway endpoints
- ✓ **Gateway Endpoints:** Support S3 and DynamoDB

# PrivateLink - Network Privacy with AWS PrivateLink

Thursday, October 21, 2021 11:37 PM

A CLOUD GURU

## Opening Your Services in a VPC to Another VPC



To open our applications up to other VPCs, we can either:

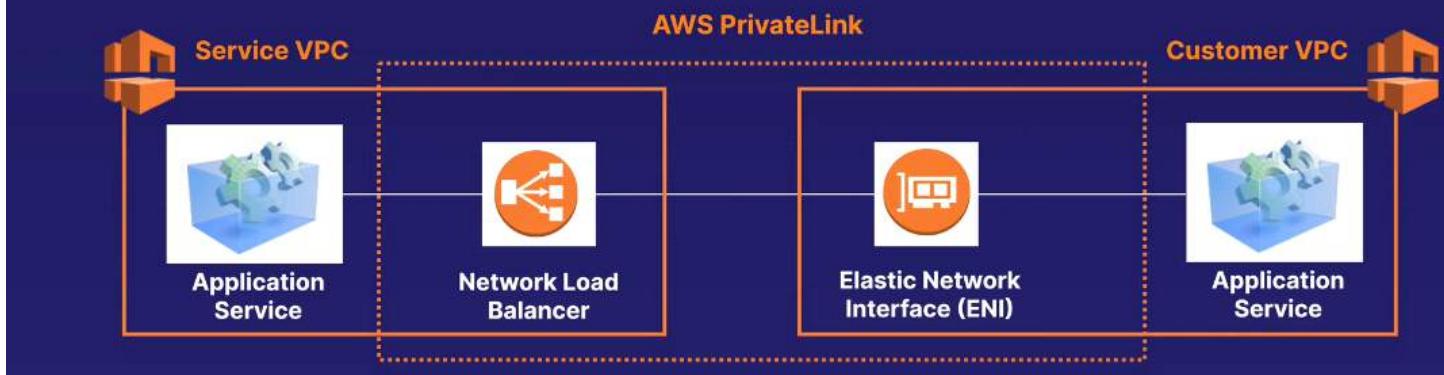
### Open the VPC up to the Internet

- Security considerations; everything in the public subnet is public
- A lot more to manage

### Use VPC Peering

- You will have to create and manage many different peering relationships.
- The whole network will be accessible. This isn't good if you have multiple applications within your VPC.

- ✓ The best way to expose a service VPC to tens, hundreds, or thousands of customer VPCs
- ✓ Doesn't require VPC peering; no route tables, NAT gateways, internet gateways, etc.
- ✓ Requires a Network Load Balancer on the service VPC and an ENI on the customer VPC



## AWS PrivateLink

- ✓ If you see a question asking about peering VPCs to tens, hundreds, or thousands of customer VPCs, think of AWS PrivateLink.
- ✓ Doesn't require VPC peering; no route tables, NAT gateways, internet gateways, etc.
- ✓ Requires a Network Load Balancer on the service VPC and an ENI on the customer VPC.

# Peering - Building Solutions across VPCs with Peering

Thursday, October 21, 2021 11:41 PM

## Multiple VPCs

Sometimes you may need to have several VPCs for different environments, and it may be necessary to connect these VPCs to each other.

**Production  
Web VPC**

**Content VPC**

**Intranet**



Allows you to connect 1 VPC with another via a direct network route using private IP addresses.



Instances behave as if they were on the same private network.



You can peer VPCs with other AWS accounts as well as with other VPCs in the same account.



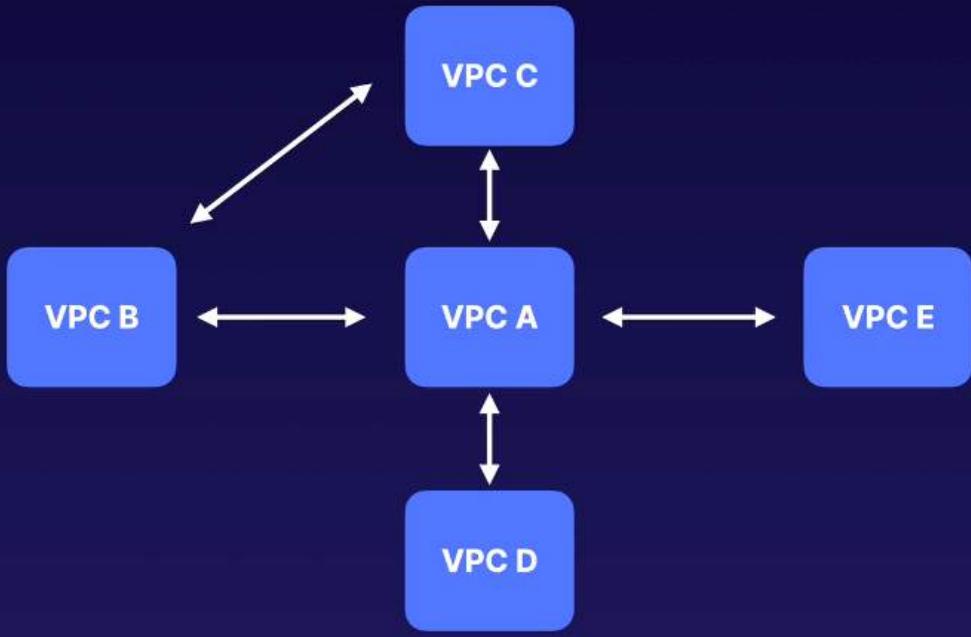
Peering is in a star configuration (e.g., 1 central VPC peers with 4 others). No transitive peering!



You can peer between regions.

VPC VPC

## Transitive Peering



## VPC Peering

- ✓ Allows you to connect 1 VPC with another via a direct network route using private IP addresses.
- ✓ Transitive peering is not supported.  
This must always be in a hub-and-spoke model.
- ✓ You can peer between regions.
- ✓ No overlapping CIDR address ranges

# CloudHub - Securing Your Network with VPN CloudHub

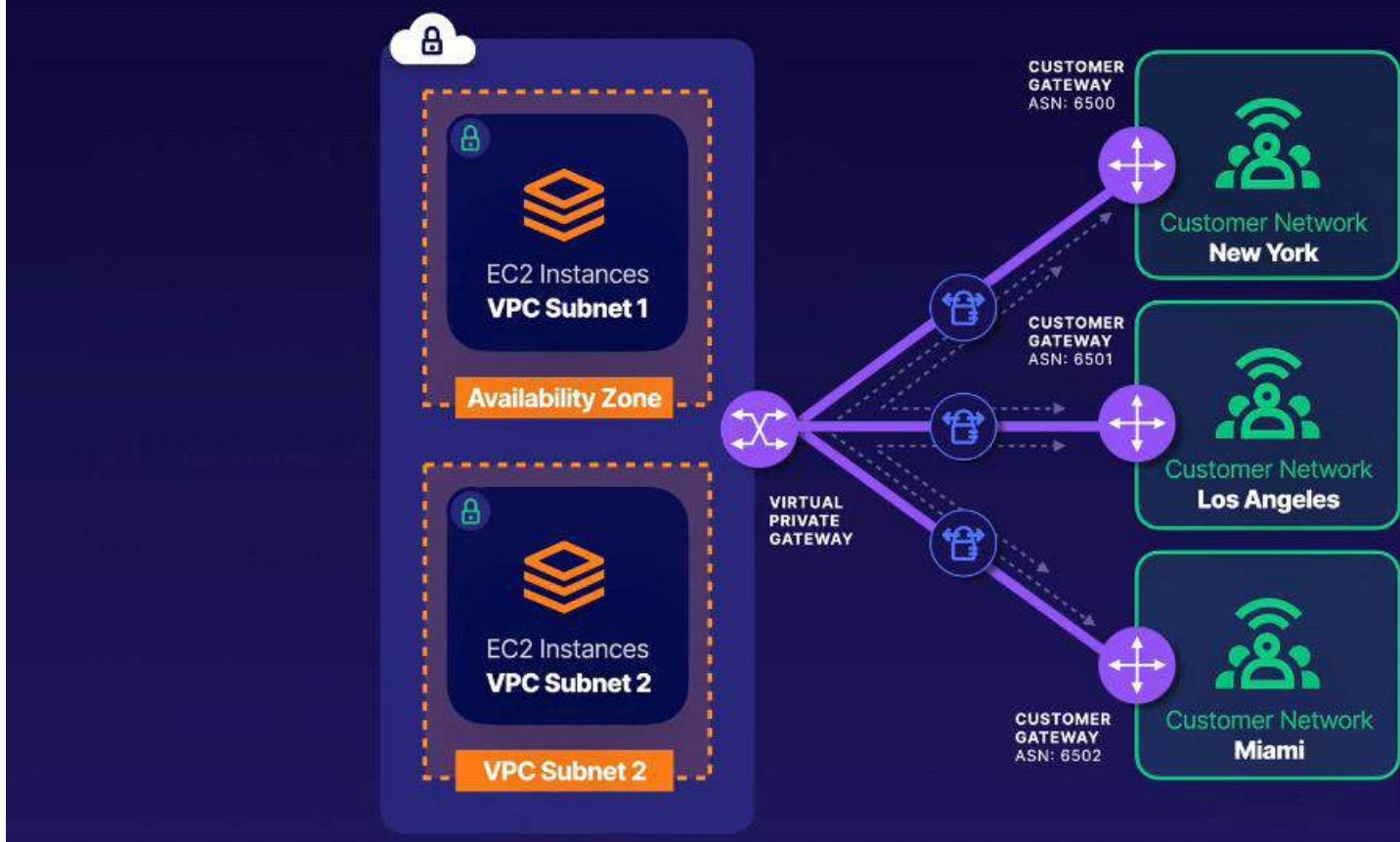
Thursday, October 21, 2021 11:46 PM

## AWS VPN CloudHub

If you have multiple sites, each with its own VPN connection, you can use AWS VPN CloudHub to connect those sites together.

- ✓ Hub-and-spoke model
- ✓ Low cost and easy to manage
- ✓ It operates over the public internet, but all traffic between the customer gateway and the AWS VPN CloudHub is encrypted.

### Network Diagram



# AWS VPN CloudHub

If you have multiple sites, each with its own VPN connection, you can use AWS VPN CloudHub to connect those sites together. It's similar to VPC peering in that it works on a hub-and-spoke model.

AWS VPN CloudHub is low cost and easy to manage. Though it operates over the public internet, all traffic between the customer gateway and the AWS VPN CloudHub is encrypted.

## Direct Connect - Connecting on Premise with Direct Connect

Thursday, October 21, 2021 11:48 PM

# What Is Direct Connect?

AWS Direct Connect is a cloud service solution that **makes it easy to establish a dedicated network connection** from your premises to AWS.



## Private Connectivity

Using AWS Direct Connect, you can establish private connectivity between AWS and your data center or office.

In many cases, you can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than internet-based connections.

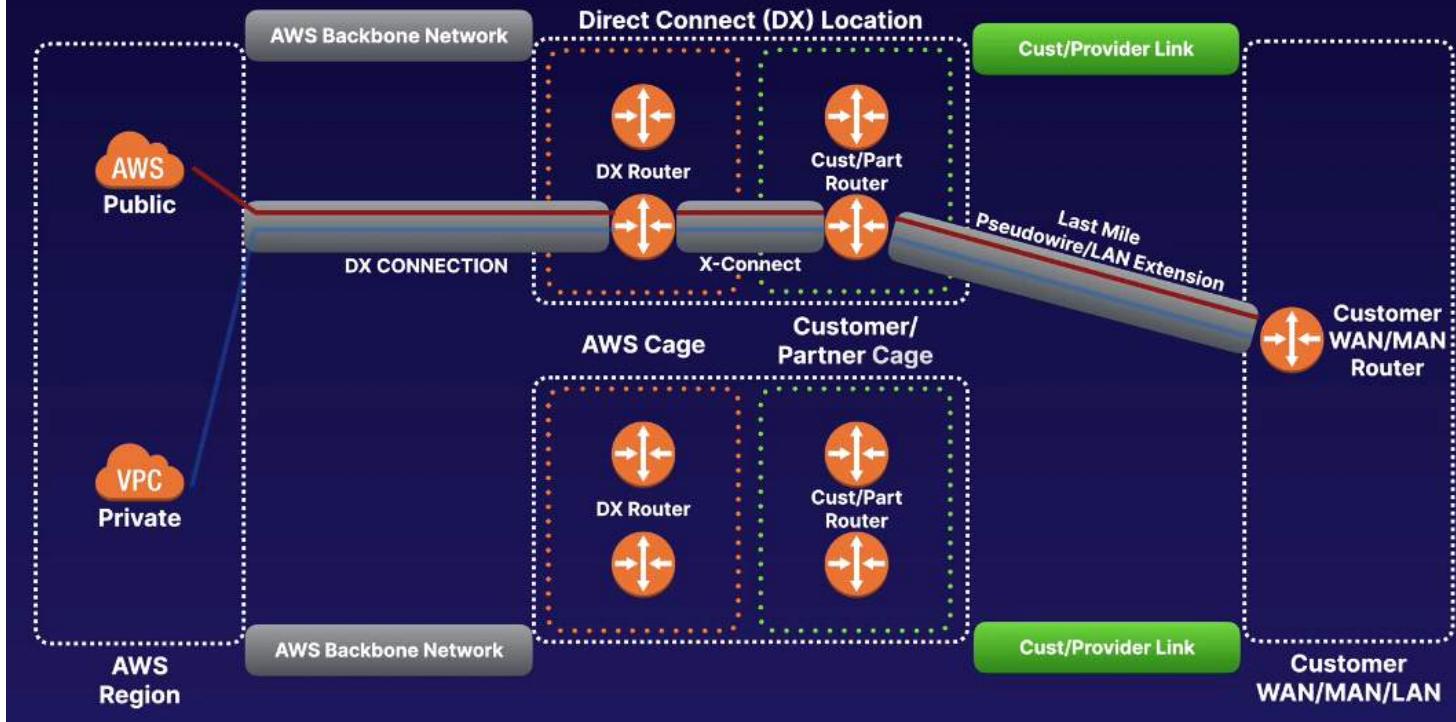
## 2 Types of Direct Connect Connection



**Dedicated Connection:** A physical Ethernet connection associated with a single customer. Customers can request a dedicated connection through the AWS Direct Connect console, the CLI, or the API.



**Hosted Connection:** A physical Ethernet connection that an AWS Direct Connect Partner provisions on behalf of a customer. Customers request a hosted connection by contacting a partner in the AWS Direct Connect Partner Program, who provisions the connection.



## VPNs vs. Direct Connect

VPNs allow private communication, but it still traverses the public internet to get the data delivered. While secure, it can be painfully slow.

### DIRECT CONNECT IS:

- ✓ Fast
- ✓ Secure
- ✓ Reliable
- ✓ Able to take massive throughput

# Direct Connect Exam Tips

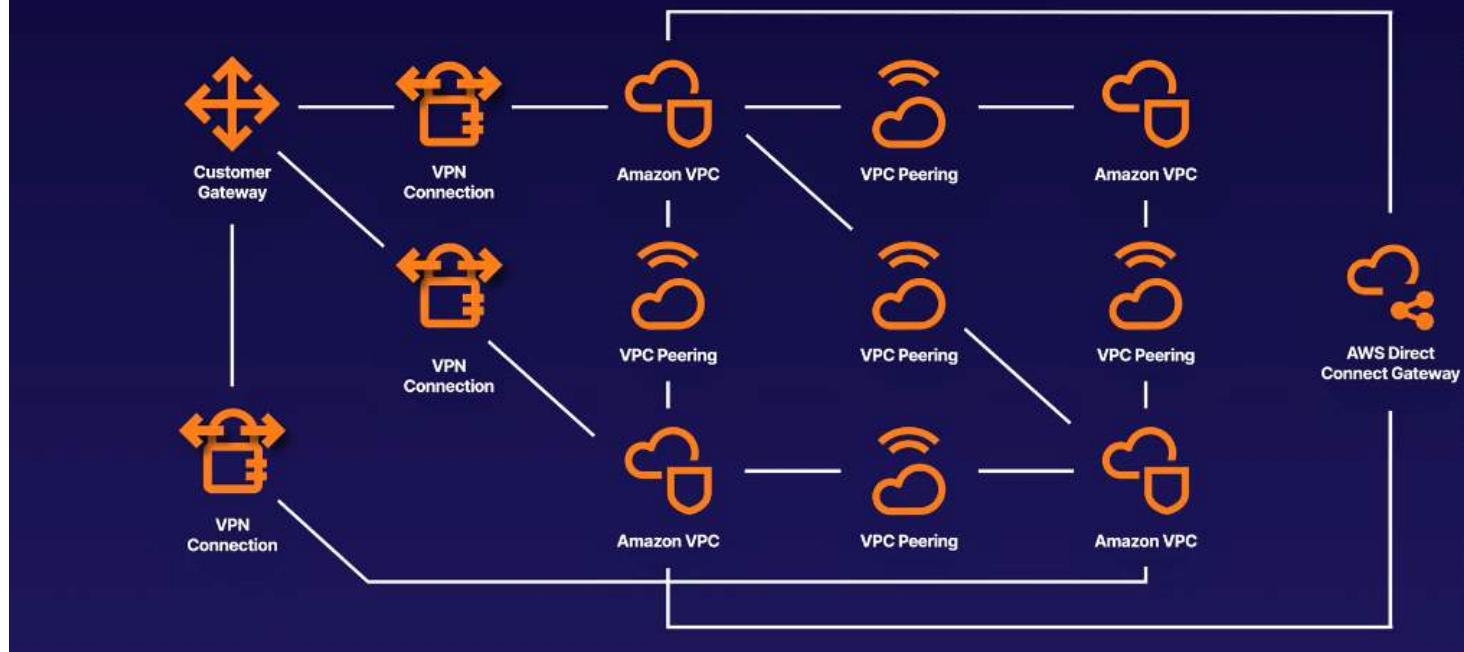
- ✓ Direct Connect directly connects your data center to AWS
- ✓ Useful for high-throughput workloads (e.g., lots of network traffic)
- ✓ Helpful when you need a stable and reliable secure connection

# Transit Gateway - Simplifying Networks with Transit Gateway

Thursday, October 21, 2021 11:54 PM

## Network Architecture Diagram

A CLOUD



## Transit Gateway

AWS Transit Gateway connects VPCs and on-premises networks through a central hub. This simplifies your network and puts an end to complex peering relationships. It acts as a cloud router — each new connection is only made once.



## Transit Gateway Facts

- ✓ Allows you to have transitive peering between thousands of VPCs and on-premises data centers.
- ✓ Works on a hub-and-spoke model.
- ✓ Works on a regional basis, but you can have it across multiple regions.
- ✓ You can use it across multiple AWS accounts using RAM (Resource Access Manager).

## Transit Gateway Exam Tips

- ✓ You can use route tables to limit how VPCs talk to one another.
- ✓ Works with Direct Connect as well as VPN connections.
- ✓ Supports IP multicast (not supported by any other AWS service).



## Elastic Block Store

Storage volumes you can attach to your EC2 instances.

Use them the same way you would use any system disk.

- Create a file system.
- Run a database.
- Run an operating system.
- Store data.
- Install applications.

## Throughput Optimized HDD (st1)

 Low-cost HDD volume

- Baseline throughput of 40 MB/s per TB
- Ability to burst up to 250 MB/s per TB
- Maximum throughput of 500 MB/s per volume
- Frequently accessed, throughput-intensive workloads
- Big data, data warehouses, ETL, and log processing
- A cost-effective way to store mountains of data
- Cannot be a boot volume

## Mission Critical

- 1 **Production Workloads**  
Designed for mission-critical workloads.
- 2 **Highly Available**  
Automatically replicated within a single Availability Zone to protect against hardware failures.
- 3 **Scalable**  
Dynamically increase capacity and change the volume type with no downtime or performance impact to your live systems.

## General Purpose SSD (gp2)

- 3 IOPS per GiB, up to a maximum of 16,000 IOPS per volume
- gp2 volumes smaller than 1 TB can burst up to 3,000 IOPS
- Good for boot volumes or development and test applications that are not latency sensitive

## General Purpose SSD (gp3)

- Predictable 3,000 IOPS baseline performance and 125 MiB/s regardless of volume size.
- Ideal for applications that require high performance at a low cost, such as MySQL, Cassandra, virtual desktops, and Hadoop analytics.
- Customers looking for higher performance can scale up to 16,000 IOPS and 1,000 MiB/s for an additional fee.



## Lowest Cost Option

- Baseline throughput of 12 MB/s per TB
- Ability to burst up to 80 MB/s per TB
- Max throughput of 250 MB/s per volume
- A good choice for colder data requiring fewer scans per day
- Good for applications that need the lowest cost and performance is not a factor
- Cannot be a boot volume



IOPS	Throughput
<ul style="list-style-type: none"> <li>• Measures the number of read and write operations per second</li> <li>• Important metric for quick transactions, low-latency apps, transactional workloads</li> <li>• The ability to action reads and writes very quickly</li> <li>• Choose Provisioned IOPS SSD (io1 or io2)</li> </ul>	<ul style="list-style-type: none"> <li>• Measures the number of bits read or written per second (MB/s)</li> <li>• Important metric for large datasets, large I/O sizes, complex queries</li> <li>• The ability to deal with large datasets</li> <li>• Choose Throughput Optimized HDD (st1)</li> </ul>

## General Purpose SSD (gp3)

- Predictable **3,000 IOPS baseline performance** and 125 MiB/s regardless of volume size.
- Ideal for applications that **require high performance at a low cost**, such as MySQL, Cassandra, virtual desktops, and Hadoop analytics.
- Customers looking for higher performance **can scale up** to 16,000 IOPS and 1,000 MiB/s for an additional fee.
- The top performance of gp3 is **4 times faster than max throughput** of gp2 volumes.

## Provisioned IOPS SSD (io1)

Up to 64,000 IOPS per volume.  
50 IOPS per GiB.

Use if you need more than 16,000 IOPS.

Designed for I/O-intensive applications,  
large databases, and latency-sensitive  
workloads.



500 IOPS per GiB.  
**Up to 64,000 IOPS.**



99.999% durability  
**instead of up to 99.9%.**



I/O-intensive apps, large  
databases, and latency-  
sensitive workloads.  
**Applications that need  
high levels of durability.**

## Throughput Optimized HDD (st1)

 Low-cost HDD volume

- Baseline throughput of 40 MB/s per TB
- Ability to burst up to 250 MB/s per TB
- Maximum throughput of 500 MB/s per volume
- Frequently accessed, throughput-intensive workloads
- Big data, data warehouses, ETL, and log processing

# Learning EBS: SSD Volumes

Highly available and scalable storage volumes **you can attach to an EC2 instance.**

gp2

## General Purpose SSD

- Suitable for boot disks and general applications
- Up to 16,000 IOPS per volume
- Up to 99.9% durability

gp3

## General Purpose SSD

- Suitable for high performance applications
- Predictable 3,000 IOPS baseline performance and 125 MiB/s regardless of volume size
- Up to 99.9% durability

io1

## Provisioned IOPS SSD

- Suitable for OLTP and latency-sensitive applications
- 50 IOPS/GiB
- Up to 64,000 IOPS per volume
- High performance and most expensive
- Up to 99.9% durability

io2

## Provisioned IOPS SSD

- Suitable for OLTP and latency-sensitive applications
- 500 IOPS/GiB
- Up to 64,000 IOPS per volume
- 99.999% durability
- Latest generation Provisioned IOPS volume

# Autoscaling Group

Thursday, 16 February 2023 10:13 PM

# Existing AWS Infrastructure

Thursday, 16 February 2023 10:14 PM

## Get Familiar with Existing AWS Infrastructure

You work for a software company that has just conducted a successful marketing campaign and expects a huge spike in networking traffic on their website. The company stated that the campaign will run weekly across the whole country, therefore network traffic fluctuations are expected over a long period of time.

The existing infrastructure consists of two EC2 instances and a single Load Balancer. The AWS cloud engineers who created the infrastructure have provided you with a script to install the website (referenced in the following challenge). You will need to create an Auto Scaling Group to prepare for network traffic fluctuations.

Click on the **Open AWS console** link to the right of this text, then use the credentials provided to log in to AWS.

In the top search box, type in and click on **EC2**.

From the navigation panel on the left, select **Instances**, and note that there are two instances with Name tags set to Legacy Instance 1 and Legacy Instance 2.

From the navigation panel on the left, select **Load Balancers**. With the **create-asg-lab-lb** load balancer selected, look at the **Description** tab and verify that the **State** parameter reads **Active**.

Copy and paste the **DNS name** of the Load Balancer into a new browser tab, and observe the website. Keep this tab open for later challenges.

The website on your browser should show **Welcome to Pluralsight Lab! This is Legacy Instance <id>**. If you refresh the connection, the <id> part should change depending on which instance is served the request.

# Create a Launch Configuration

Thursday, 16 February 2023 10:14 PM

## Create a Launch Configuration

Now that you know what the existing infrastructure is like, you can add a Launch Configuration. Launch Configurations are templates for launching EC2 instances, and one will be needed by the Auto Scaling Group.

Near the bottom of the left navigation menu, click **Launch Configurations**.

Click **Create launch configuration**.

**Name** the launch configuration `create-asg-lab-lc`.

In the **Amazon machine image (AMI)** section, paste the AMI ID of `ami-0c2ab3b8efb09f272`, and click the result in the drop-down to select it.

Note: This AMI ID is the Amazon-owned HVM Amazon Linux 2 image.

For **Instance type**, click **Choose instance type**, paste `t2.nano` into the search bar, select the instance type, then click **Choose**.

In the **Additional configuration** section, expand the **Advanced details** sub-section.

In the **User data** section, ensure `As text` is selected, and paste the following code in the text field:

```
#!/bin/bash
sudo yum update -y
sudo amazon-linux-extras install nginx1 -y
sudo echo '
<title>Create an Auto Scaling Group</title>
<h1>Welcome to Pluralsight Labs! This is an ASG instance.</h1>
' > /usr/share/nginx/html/index.html
sudo systemctl start nginx
```

Scroll down to the **Security groups** section, and select the radio button for **Select an existing security group**, as the security group for Legacy Instances is already in place and has a working connection between the instances and the Load Balancer.

Search for and select the checkbox for the security group named **create-asg-lab-sg-instance**.

Note: There should be a warning at the bottom of the Security groups section stating that connection to port 22 will not be possible. That's OK for this lab as you will not attempt to connect to the instances themselves.

Note: When changing infrastructure, try to have as little impact as possible, especially in production environments. Creating additional security groups could create confusion and potential outages in more complex infrastructures.

In the **Key pair (login)** section, from the **Key pair options** drop-down choose **Proceed without a key pair**, then check the box below to acknowledge that you will **not be able to connect** to the instance.

Briefly review the details that you customized, then click **Create launch configuration**.

You will see the new launch configuration in the list named **create-asg-lab-lc**.

Observe the launch configuration details. At this stage you will not be able to modify this existing launch configuration; you can only create a copy of the launch configuration and adjust any discrepancies, or delete it and start over.



# Create an Autoscaling Group

Thursday, 16 February 2023 10:14 PM

Now that the launch configuration is created, you can create an Auto Scaling Group to automatically manage instance count required for the marketing campaigns.

At the bottom of the left-hand menu, click **Auto Scaling Groups**.

Click **Create Auto Scaling group**.

Give the Auto Scaling group a **Name** of `create-asg-lab-asg`.

In the **Launch template** section, click **Switch to launch configuration**, then select the launch configuration that you created in the previous challenge, **create-asg-lab-lc**.

Click **Next**.

In the **Network** section, for **VPC** select **create-asg-lab-vpc**, and for **Availability Zones and subnets** select **create-asg-lab-subneta** and **asg-lab-subnetb**.

Click **Next**.

At the **Configure advanced options** page, select the checkbox for **Attach to an existing load balancer**, and from the **Existing load balancer target groups** drop-down, choose **front-end-tg**.

Click **Next**.

In the **Group size** section, set all three values to 4.

Click **Skip to review**.

Briefly review the details you've customized in these tasks, then click **Create Auto Scaling group**.

You will be redirected back to the **Auto Scaling groups** list, where you will see that the **Instances** value on the Auto Scaling group is **0**. This is because the Auto Scaling group is not yet at its desired size of 4. Wait a minute then press the refresh button and this parameter should go up to **4**.

Navigate back to the **Instances** page; there should be 6 instances: two Legacy Instances and 4 instances without a **Name** tag. You may need to click the Refresh button to see the new instances. You will fix this in the next challenge.

# Simulate a Website Outage

Thursday, 16 February 2023 10:15 PM

## Simulate a Website Outage

Now that the Auto Scaling Group is created you will need to clean up Legacy Instances so that the instance count is only set by the Auto Scaling Group. Differentiating between resources is crucial during troubleshooting, so you will add Name tags for newly created instances and recycle the old ones. The following tasks cause downtime of the website as you will terminate all instances, and observe Auto Scaling Group scale out action.

Ensure you're at the EC2 Instances page.

Select both **Legacy** Instances in the table, and click **Instance state > Terminate instance**. Confirm by clicking **Terminate**.

This will remove the instances from the Load Balancer. Be careful when dealing with dynamic websites and sessions, as some sessions can be broken for clients who are connected to particular instances, but in this case the instances are serving static content; therefore, terminating instances this way is safe. In case of active sessions, remove instances from the load balancer first, then terminate.

From the menu on the left, click **Auto Scaling Groups**, then select the checkbox for **create-asg-lab-asg**.

Scroll down to the **Tags** section and click **Edit**.

Click **Add tag**. Give the tag a **Key of Name** and a **Value of ASG Instance**.

Click **Update**.

From the menu on the left, click **Instances**.

Select all four of the instances which are in the **Running Instance state**, then **Terminate** them.

All Legacy instances and instances created by the Auto Scaling Group are terminated at this point, and this effectively simulates an outage of the website.

The load balancer's DNS name should return 500 errors, like a 503 or 504, since it does not have any instances to send HTTP requests to.

Wait 10 minutes or so for the Auto Scaling group to realize that all instances are missing, and for all four **ASG Instance** EC2s to be created, and achieve a **state of Running**.

Refresh the Instances page every few minutes. The Auto Scaling group might not realize that all instances are missing at once; therefore, it might scale out one instance at a time, but eventually all four will be present. This depends on the Load Balancer Health Check configuration.

Once all four new instances are present, you can validate whether the website works again: the website should say, **Welcome to Pluralsight Lab! This is an ASG Instance!**

# Create a Scaling Policy

Thursday, 16 February 2023 10:15 PM

## Create a Scaling Policy

Now you know that the Auto Scaling will recreate any terminated instances to keep a static amount of instances. At the moment, four instances created by the Auto Scaling Group will run forever. Some of them might not be needed at all times during low traffic periods. Therefore, you need to create a Scaling Policy to automatically adjust instance count.

Click on **Auto Scaling Groups** in the left menu, then select the **create-asg-lab-asg** ASG.

Click on the **Automatic scaling** tab, and then click **Create dynamic scaling policy**.

Create the scaling policy using the following values:

- Scaling policy name: **create-asg-lab-sp**
- Target value: **75**
- Instances need: **20**

Leaving everything else as default, and click **Create**.

If you wait a while now, you will realize that nothing is happening. The Auto Scaling Group still maintains the four instances. This is because the minimal amount of instances for the Auto Scaling Group is still set to four; therefore, the Auto Scaling group will still keep this number of instances at all times.

In the **Auto Scaling groups** section, click **Edit** to change **create-asg-lab-asg**.

Set the **Minimum capacity** value to **2** and click **Update**.

This configuration will effectively terminate two instances, since the CPU utilization will be below 50%. When the Scaling Policy detects more than 50% of CPU utilization across those two instances, it'll add one instance. If CPU utilization is still above 50%, it'll add a fourth instance. However, if CPU utilization is still above 50% with four instances present, the Scaling Policy will not add any more as the maximum number of instances is 4, set by Max parameter.

Note: In this lab, it can take 10 to 25 minutes for the two instances to be terminated. You are not required to wait for this to happen. So now the company is able to conduct marketing campaigns all over the world and the website is ready to handle the additional load by scaling out, and also save money by scaling in during periods of decreased traffic. If the amount of traffic causes all four instances to be overloaded, the Max parameter can be set to a higher number causing Auto Scaling group to provision even more instances. Even though this deployment looks very basic it happens to be the standard building block of elastic infrastructures used all over the world for a huge variety of frontend and backend applications.

# VPC

Thursday, 16 February 2023 10:17 PM

VPC

Subnet

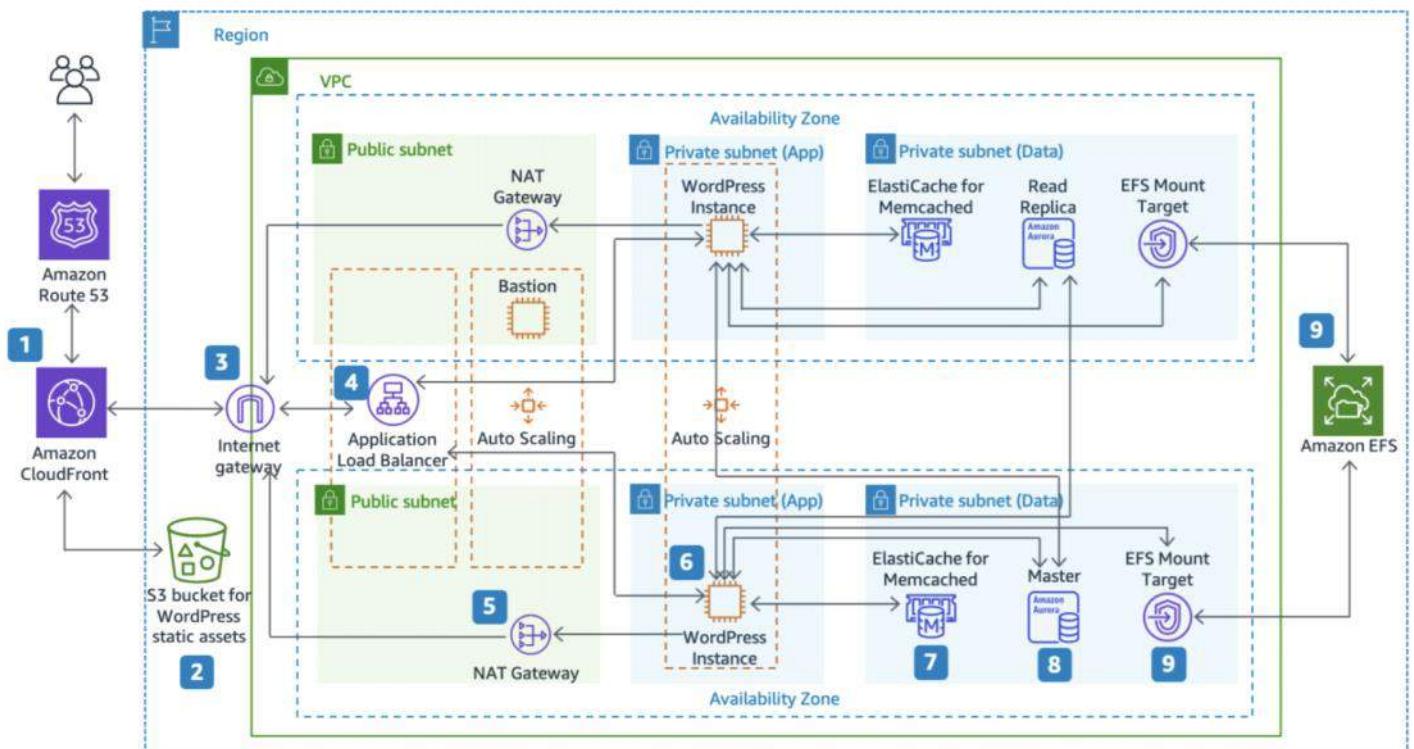
Route

Route - Subnet Association

Internet GW

Route - edit route - 0.0.0.0/0

- Create VPC - make it secure, private env
- Create corresponding subnets (private for App and DB) (Public for Web)
- Create route tables & do subnet associations (helping you traffic being routed through your VPC)
- Create Internet GW (we need to establish the connectivity) & attached to VPC - allows communication between resources in your VPC and the internet.
- Create NAT GW in public subnet- is in each Availability Zone that enable EC2 instances in private subnets (App and Data) to access the internet.
- Make changes in route table so that connectivity to the internet will be established from your internet gateway for the public subnet &NAT gw for private subnets
- Create jump server & App server



# EC2 instance access to an Amazon S3 bucket

Thursday, 16 February 2023 10:21 PM

1. Create an AWS Identity and Access Management (IAM) profile role that grants access to Amazon S3.
2. Attach the IAM instance profile to the instance.
3. Validate permissions on your S3 bucket.
4. Validate network connectivity from the EC2 instance to Amazon S3.
5. Validate access to S3 buckets.

# Features

Monday, 14 November 2022 3:55 PM

Service platform of redhat, providing many features like auto-scaling  
Rich command line toolset, Support Multi databases and language.

Other:

1. Multi-Environment support.
2. One-click deployment.
3. Responsive web console.
4. Remote application debugging.
5. Standardized workflow for developers.
6. Rest API support.
7. Support for remote SSH login to the application.
8. In-built database services.
9. Automatic application scaling.
10. The facility of support for release management and continuous integration.
11. IDE integration.

# Overview

Wednesday, 15 February 2023 9:05 PM

- Cloud development platform as a service (PaaS) hosted by Red Hat
  - User friendly platform: Create, test and run application and finally deploy them on cloud
  - Capable of managing application written in different languages
  - Based on virtualization
- 
- Provide managed hardware and network resources for all kinds of development and testing

## Service Plan:

- Free | Limited plan to three gears with 1GB space for each
- Bronze | Three gears and expand up to 16 gears with 1GB space per gear
- Silver | 16 gear plan of bronze, Storage capacity of 6GB with no additional cost

## Features:

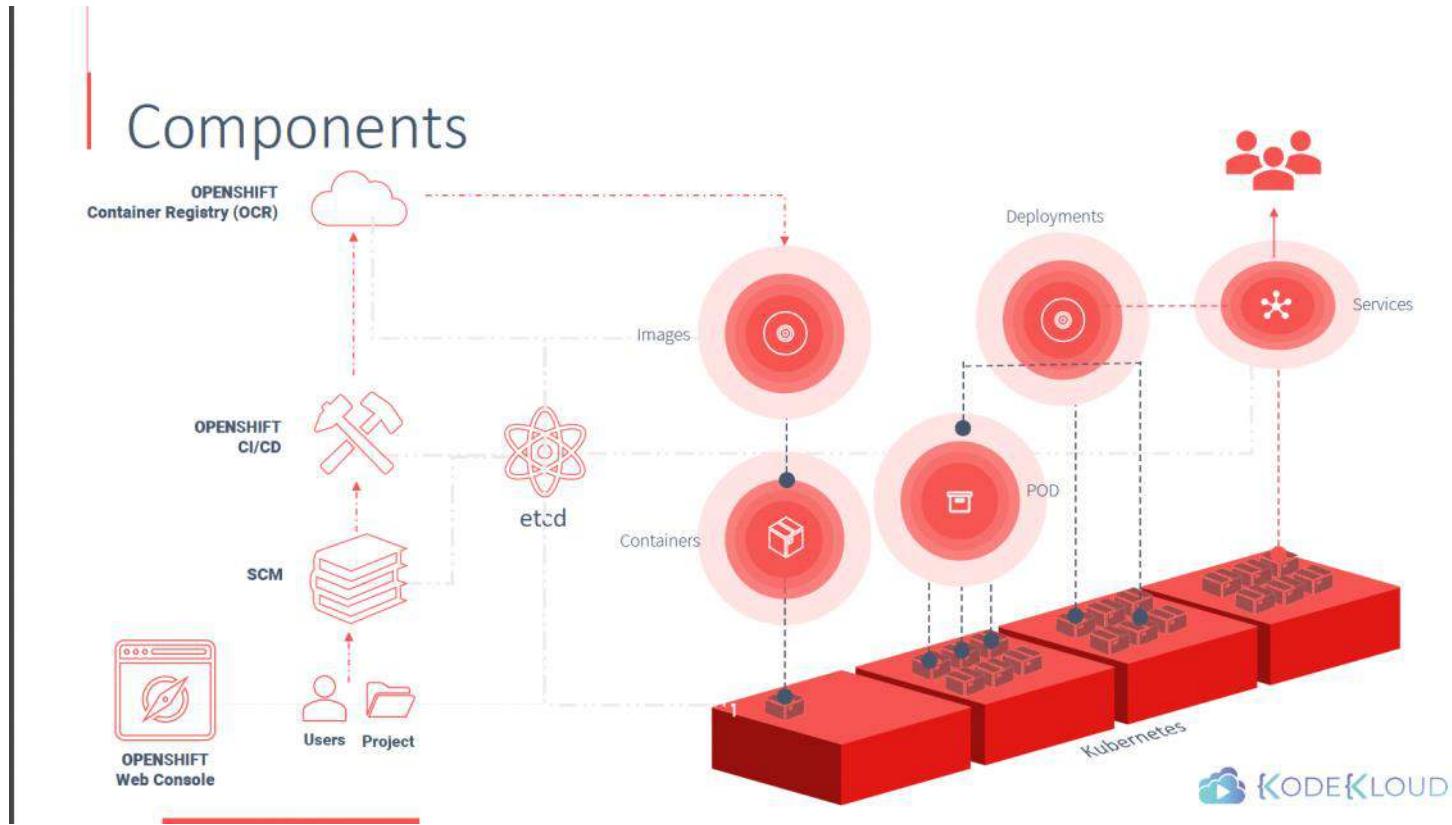
- Multiple Language Support
- Multiple Database Support
- Extensible Cartridge system
- Source Code Version Management
- One-Click Deployment
- Multi Environment Support
- Standardized Developers' workflow

## Tools

- SCM
- Pipeline
- Registry
- Software defined Networking
- API
- Governance

# Components Management tool

Wednesday, 15 February 2023 10:54 PM



## Management tool

CLI

API

Web Console

# Openshift types

Wednesday, 15 February 2023 9:20 PM

**Cartridges:** They were the focal point of building a new application starting from the type of application the environment requires to run them and all the dependencies satisfied in this section.

**Gear:** It can be defined as the bear metal machine or server with certain specifications regarding the resources, memory, and CPU. They were considered as a fundamental unit for running an application.

**Application:** These simply refer to the application or any integration application that will get deployed and run on OpenShift environment.

**OpenShift Origin:** This was the community addition

**OpenShift Online:** It is a public PaaS as a service hosted on AWS.

**OpenShift Enterprise:** This is the hardened version of OpenShift with ISV and vendor licenses.

OpenShift Container Platform

- OpenShift Container Local
- OpenShift Container Lab

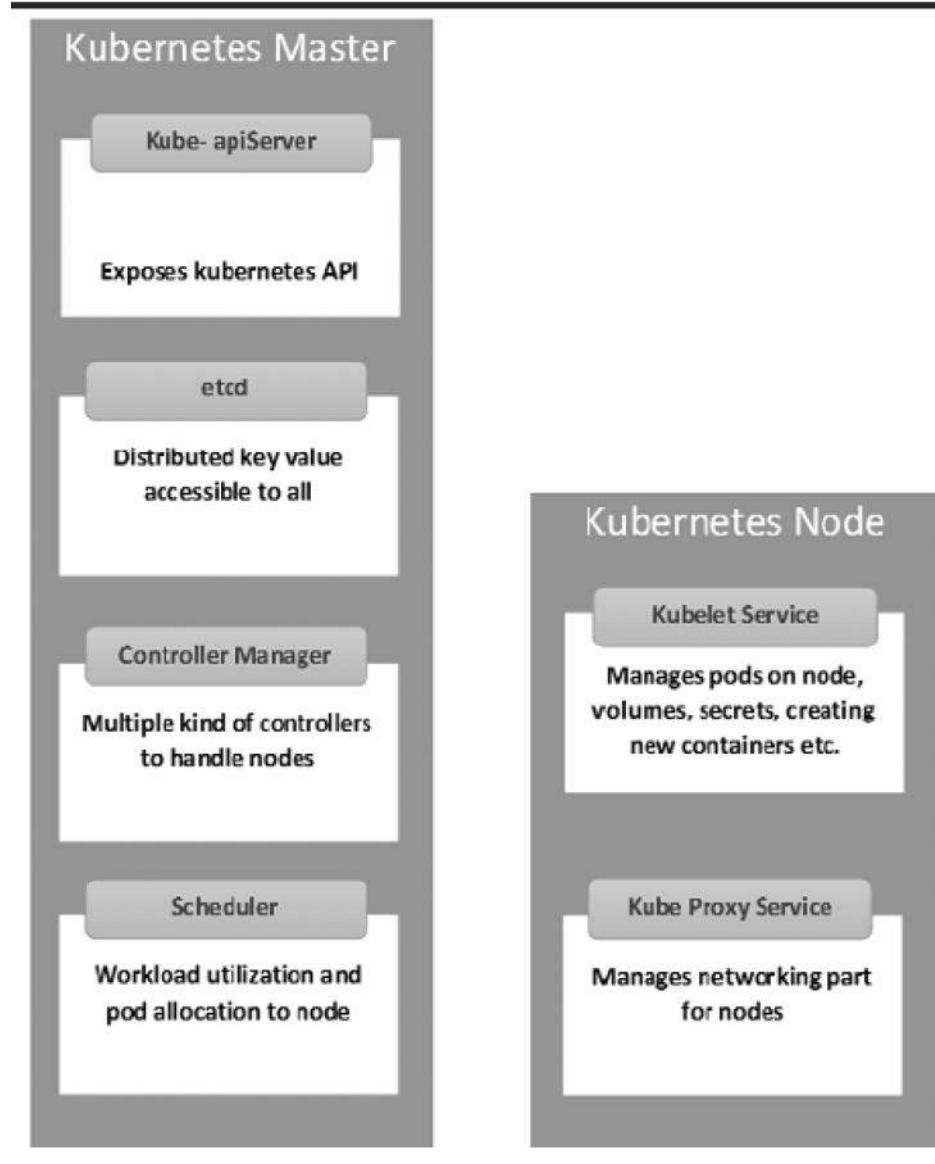
OpenShift Dedicated (Customer choice of hosting on Cloud)

- Extensible and open
- Portability
- Orchestration
- Automation

# Architecture

Wednesday, 15 February 2023 9:39 PM

Layered System: Tightly bound with layers using Kubernetes and Docker cluster.



# Container Registry

Wednesday, 15 February 2023 10:07 PM

- Inbuilt storage unit of redhat for storing docker images
- Comes with user interface to view images in OpenShift internal storage.
- Capable of holding images with specified tags.

# Deployment Strategies

Monday, 14 November 2022 3:59 PM

Instruments for modifying or upgrading an application and it doesn't required downtime.

Blue-Green Deployment Strategy

cartridges?

Monday, 14 November 2022 4:11 PM

# CheatSheet

Thursday, 16 February 2023 11:12 AM

## Login and Configuration

Firstly, let's check the most common commands for Login and Configuration in OpenShift:

```
#login with a user  
oc login https://192.168.99.100:8443 -u developer -p developer
```

```
#login as system admin  
oc login -u system:admin
```

```
#User Information  
oc whoami
```

```
#View your configuration  
oc config view
```

```
#Update the current context to have users login to the desired namespace:  
oc config set-context `oc config current-context` --namespace=<project_name>
```

## Basic Commands

Secondly, here is a list of the basic commands to manage Pods and create applications with Templates:

```
#Create a new app from a GitHub Repository  
oc new-app https://github.com/sclorg/cakephp-ex
```

```
#New app from a different branch  
oc new-app --name=html-dev nginx:1.10-https://github.com/joe-speedboat/openshift.html.devops.git#mybranch
```

```
#Create objects from a file:  
oc create -f myobject.yaml -n myproject
```

```
#Delete objects contained in a file:  
oc delete -f myobject.yaml -n myproject
```

```
#Create or merge objects from file  
oc apply -f myobject.yaml -n myproject
```

```
#Update existing object  
oc patch svc mysvc --type merge --patch '{"spec":{"ports":[{"port": 8080, "targetPort": 5000 }]}'>
```

```
#Monitor Pod status  
watch oc get pods
```

```
#Get a Specific Item (podIP) using a Go template  
oc get pod example-pod-2 --template='{{.status.podIP}}'
```

```
#Gather information on a project's pod deployment with node information  
oc get pods -o wide
```

```
#Hide inactive Pods  
oc get pods --show-all=false
```

```
#Display all resources  
oc get all,secret,configmap
```

```
#Get the Openshift Console Address  
oc get -n openshift-console route console
```

```
#Get the Pod name from the Selector and rsh in it  
POD=$(oc get pods -l app=myapp -o name) oc rsh -n $POD
```

```
#Exec single command in pod  
oc exec $POD $COMMAND
```

```
#Copy from local folder byteman-4.0.12 in Pod wildfly-basic-1-mrlt5 under the folder /opt/wildfly  
oc cp ./byteman-4.0.12 wildfly-basic-1-mrlt5:/opt/wildfly
```

## Image Streams

Here is how to list and import ImageStreams on OpenShift

```
#List available IS for openshift project  
oc get is -n openshift
```

```
#Import an image from an external registry  
oc import-image --from=registry.access.redhat.com/jboss-amq-6/amq62-openshift -n openshift jboss-amq-62:1.3 --confirm
```

```
#List available IS and templates  
oc new-app --list
```

## Templates Management

Next, here is how to process Templates:

```
# Deploy resources contained in a template  
oc process -f template.yaml | oc create -f -
```

```
#List parameters available in a template  
oc process --parameters -f .template.yaml
```

## Setting environment variables

Then, here is how to set environment variables on Deployment Configs/Build Configs and list them:

```
# Update deployment 'registry' with a new environment variable  
oc set env dc/registry STORAGE_DIR=/local
```

```
# List the environment variables defined on a build config 'sample-build'  
oc set env bc/sample-build --list
```

```
# List the environment variables defined on all pods  
oc set env pods --all --list
```

```
# Import environment from a secret  
oc set env --from=secret/mysecret dc/myapp
```

## WildFly application example on OpeShift

Here is how to bootstrap a **WildFly** application on OpenShift using a legacy Image Stream:

```
# Create WildFli Image Stream  
oc create -f https://raw.githubusercontent.com/wildfly/wildfly-s2i/wf-26.0/imagestreams/wildfly-centos7.json
```

```
# Create WildFly app from GitHub Repo  
$ oc new-app wildfly:26.0~https://github.com/fmarchioni/ocpdemos --context-dir=wildfly-basic --name=wildfly-basic
```

```
#Expose Service with a Route  
oc expose service wildfly-basic
```

Then, here is how to use Helm to bootstrap an application from an Helm Chart:

```
#Add WildFly Helm Chart to the Repository  
helm repo add wildfly https://docs.wildfly.org/wildfly-charts/
```

```
# Image Streams and Configuration in the file sampleapp.yaml  
helm install sample-app wildfly/wildfly -f sampleapp.yaml
```

## Create app from a Project with Dockerfile

Next, here is how to create an app from a Dockerfile using a Binary Build:

```
oc new-build --binary --name=mywildfly -l app=mywildfly
```

```
oc patch bc/mywildfly -p '{"spec":{"strategy":{"dockerStrategy":{"dockerfilePath":"Dockerfile"}}}}'
```

```
oc start-build mywildfly --from-dir=. --follow
```

```
oc new-app --image-stream=mywildfly
```

```
oc expose svc/mywildfly
```

# How to manage Nodes

```
#Get Nodes list  
oc get nodes  
  
#Check on which Node your Pods are running  
oc get pods -o wide  
  
#Schedule an application to run on another Node  
oc patch dc myapp -p '{"spec":{"template":{"spec":{"nodeSelector":{"kubernetes.io/hostname": "ip-10-0-0-74.acme.compute.internal"}}}}}'  
  
#List all pods which are running on a Node  
oc adm manage-node node1.local --list-pods  
  
#Add a label to a Node  
oc label node node1.local mylabel=myvalue  
  
#Remove a label from a Node  
oc label node node1.local mylabel-
```

# How to manage storage

```
#create a PersistentVolumeClaim (+update the DeploymentConfig to include a PV + update the DeploymentConfig to attach a volumemount into the specified mount-path)  
  
oc set volume dc/file-uploader --add --name=my-shared-storage \  
-t pvc --claim-mode=ReadWriteMany --claim-size=1Gi \  
--claim-name=my-shared-storage --claim-class=ocs-storagecluster-cephfs \  
--mount-path=/opt/app-root/src/uploaded \  
-n my-shared-storage
```

```
#List storage classes  
oc -n openshift-storage get sc
```

# Build Management

```
#Manual build from source  
oc start-build ruby-ex
```

```
#Manual build from source and follow logs  
oc start-build ruby-ex -F
```

```
#Stop a build that is in progress  
oc cancel-build <build_name>
```

```
#Changing the log level of a build:  
oc set env bc/my-build-name BUILD_LOGLEVEL=[1-5]
```

# How to manage Deployments

```
#Manual deployment  
$ oc rollout latest ruby-ex
```

```
#Pause automatic deployment rollout  
oc rollout pause dc $DEPLOYMENT
```

```
# Resume automatic deployment rollout  
oc rollout resume dc $DEPLOYMENT
```

```
#Define resource requests and limits in DeploymentConfig  
oc set resources deployment nginx --limits(cpu=200m,memory=512Mi) --requests(cpu=100m,memory=256Mi)
```

```
#Define livenessProbe and readinessProbe in DeploymentConfig  
oc set probe dc/nginx --readiness --get-url=http://:8080/healthz --initial-delay-seconds=10  
oc set probe dc/nginx --liveness --get-url=http://:8080/healthz --initial-delay-seconds=10
```

```
#Scale the number of Pods to 2  
oc scale dc/nginx --replicas=2
```

```
#Define Horizontal Pod Autoscaler (hpa)
oc autoscale dc $DC_NAME --max=4 --cpu-percent=10
```

## Managing Routes

```
#Create route
$ oc expose service ruby-ex
```

```
# Create Route and expose it through a custom Hostname
oc expose serviceruby-ex --hostname
```

```
#Read the Route Host attribute
oc get route my-route -o jsonpath --template="{.spec.host}"
```

## Managing Services

```
#Make a service idle. When the service is next accessed will automatically boot up the pods again:
$ oc idle ruby-ex
```

```
#Read a Service IP
oc get services rook-ceph-mon-a --template='{{.spec.clusterIP}}'
```

## Clean up resources

```
#Delete all resources
oc delete all --all
```

```
#Delete resources for one specific app
$ oc delete services -l app=ruby-ex
$ oc delete all -l app=ruby-ex
```

```
#CleanUp old docker images on nodes
#Keeping up to three tag revisions 1, and keeping resources (images, image streams and pods) younger than sixty minutes:
oc adm prune images --keep-tag-revisions=3 --keep-younger-than=60m
```

```
#Pruning every image that exceeds defined limits:
oc adm prune images --prune-over-size-limit
```

## Openshift Container Platform Troubleshooting

```
#How to inspect all resources in a namespace (produces resources tree in YAML files)
oc adm inspect ns/mynamespace
```

```
#run cluster diagnostics
oc adm diagnostics
```

```
#Collect must-gather
oc adm must-gather
```

```
#Check status of current project
oc status
```

```
#Get events for a project
oc get events --sort-by='lastTimestamp'
```

```
# get the logs of the myrunning-pod-2-fdthn pod
oc logs myrunning-pod-2-fdthn<br />
```

```
# follow the logs of the myrunning-pod-2-fdthn pod
oc logs -f myrunning-pod-2-fdthn<br />
```

```
# tail the logs of the myrunning-pod-2-fdthn pod
oc logs myrunning-pod-2-fdthn --tail=50
```

```
#Check the integrated Docker registry logs:
oc logs docker-registry-n-{xxxxx} -n default | less
```

## Security

```

#Create a secret from the CLI
oc create secret generic oia-secret --from-literal=username=myuser
--from-literal=password=mypassword

# Use secret in deployment env
oc set env deployment/ --from secret/oia-secret

# You can also mount the Secret on a Volume
oc set volumes dc/myapp --add --name=secret-volume --mount-path=/opt/app-root/
--secret-name=oia-secret

```

## Managing user roles

```

oc adm policy add-role-to-user admin oia -n python
oc adm policy add-cluster-role-to-user cluster-reader system:serviceaccount:monitoring:default
oc adm policy add-scc-to-user anyuid -z default

```

## Misc commands

```

#Manage node state
oc adm manage node <node> --schedulable=false

#List installed operators
oc get csv

#Export in a template the IS, BC, DC and SVC
oc export is,bc,dc,svc --as-template=app.yaml

#Show user in prompt
function ps1(){
  export PS1='[\u@\h$(oc whoami -c 2>/dev/null|cut -d/ -f3,1)) \w]\$ '
}

#backup openshift objects

oc get all --all-namespaces --no-headers=true | awk '{print $1","$2}' | while read obj
do
  NS=$(echo $obj | cut -d, -f1)
  OBJ=$(echo $obj | cut -d, -f2)
  FILE=$(echo $obj | sed 's//-/g;s/-/g')
  echo $NS $OBJ $FILE; oc export -n $NS $OBJ -o yaml > $FILE.yaml
done

```

# About me

Sabtu, 25 Februari 2023 11:13 PTG

I'm responsible for designing, building, and maintaining the technical infrastructure that supports the software applications.

It's include anything from the server hardware to the network infrastructure to the storage solutions.

I have strong technical skills and be able to communicate effectively with other members of the engineering team.

Platform Engineering is the process of designing, building, and maintaining a platform that can be used to develop, test, and deploy software applications. A platform can be either a physical or virtual machine, or a set of software components that are used to build, test, and deploy applications.

# Best Practice

Sabtu, 25 Februari 2023 11:25 PTG

1. Define the business goals and objectives that the platform is meant to achieve.
2. Conduct a comprehensive analysis of the existing IT landscape to identify gaps and areas of improvement.
3. Put together a detailed plan for the platform implementation, including timelines, milestones, and deliverables.
4. Assign dedicated resources to manage and oversee the platform implementation.
5. Test and validate the platform before going live.

## vmem

Friday, 3 February 2023 6:10 PM

- While creating snapshot if we select option “Snapshot Virtual machine’s memory”, then the vmem file gets created which stores all the contents of virtual machine memory.

# HA & FT

Selasa, 21 Februari 2023 12:10 PTG

Vsphere HA service can be enabled on cluster level and allows to reduce downtime by restarting VMs on another ESxi host in case of host failure.

FT can be enabled on VMs to assure 100 % uptime for services and applications running in VM.  
FT can be enabled only for critical applications.

# Master node

Selasa, 21 Februari 2023 12:11 PTG

Master monitors the state of slave host. It also monitors state of all powered on machines. If VM fails it makes sure that it restarts on other host in cluster and similarly if host fails it will restart all VMs on other hosts.

we enable HA service on cluster , FDM agent (or HA Agent) gets installed on every host in a cluster. After that master slave election takes place. Out of all host one will become master and others will act as slave.

# RDM

Selasa, 21 Februari 2023 12:13 PTG

RDM stands for Raw Device Mapping. Storage LUN we can directly map to the VM.

RDM mapping file contains the information about the location and locking state of mapped device.

# vVols

Selasa, 21 Februari 2023 12:14 PTG

vVols are similar to the virtual disk of machine.

# Linked Clone

Selasa, 21 Februari 2023 12:15 PTG

Linked clone is a copy of VM that shares virtual disk with parent VM.

The original VM is called parent.

# VMKernel adaptor

Selasa, 21 Februari 2023 12:15 PTG

VMKernel adapter is used to assign ip address to ESXi host and we can also enable different services such as replication traffic, provisioning , Vmotion , FT and other services on this adapter.

# Hypervisor

Selasa, 21 Februari 2023 12:20 PTG

## #1. What is VMKernel, and why is it important?

VMkernel is a virtualization interface between a Virtual Machine and the ESXi host, which stores VMs. It is responsible for allocating all available resources of the ESXi host to VMs such as memory, CPU, storage, etc. It's also controlled special services such as vMotion, Fault tolerance, NFS, traffic management, and iSCSI. To access these services, the VMkernel port can be configured on the ESXi server using a standard or distributed vSwitch. Without VMkernel, hosted VMs cannot communicate with the ESXi server.

## #2. What are the hypervisor and their types?

A hypervisor is a virtualization layer that enables multiple operating systems to share a single hardware host. Each operating system or VM is allocated physical resources such as memory, CPU, storage, etc., by the host. There are two types of hypervisors.

- Hosted hypervisor (works as application i.e VMware Workstation)
- Bare-metal (is virtualization software i.e VMvisor, [Hyper-V which is installed directly onto the hardware and controls all physical resources](#)).

## #3. What is Virtualization?

The process of creating virtual versions of physical components, i.e., Servers, Storage Devices, Network Devices on a physical host, is called virtualization. Virtualization lets you run multiple virtual machines on a single physical machine which is called ESXi host.

## #4. What are the different types of virtualization?

There are 5 basic types of virtualization.

- Server virtualization: consolidates the physical server, and multiple OS can be run on a single server.
- Network Virtualization: Provides complete reproduction of physical network into a software-defined network.
- Storage Virtualization: Provides an abstraction layer for physical storage resources to manage and optimize virtual deployment.
- Application Virtualization: increased mobility of applications and allows migration of VMs from a host to another with minimal downtime.
- [Desktop Virtualization: virtualize desktop to reduce cost and increase service](#)

## #5. What is VMware FT?

FT stands for Fault Tolerance very prominent component of VMware vSphere. It provides continuous availability for VMs when an ESXi host fails. It supports up to 4 vCPUs and 64 GB memory. FT is very bandwidth-intensive, and 10GB NIC is recommended to configure it. It creates a complete copy of an entire VM such as storage, compute, and memory.

## #6. How many vCPUs can be used for a VM in FT in VMware vSphere 7.0?

In VMware vSphere 7.0, there can be up to 8 vCPUs with the VMware vSphere Enterprise Plus license.

## #7. What is the name of the technology used by VMware FT?

vLockstep technology is used by VMware FT

## #8. What is Fault Tolerant Logging?

The communication between two ESXi hosts is called FT logging when FT is configured between them. The pre-requisition of configuring FT is to configure the VMKernel port.

## #9. Will the FT work if the vCenter Server goes down?

vCenter Server is only required to enable Fault Tolerance on a VM. Once it is configured, vCenter is not required to be online for FT to work. FT failover between primary and secondary will occur even if the vCenter is down.

## #10. What is the main difference between VMware HA and FT?

The main difference between VMware HA and FT is:

HA is enabled per cluster, and VMware FT is enabled per VM.

HA, VMs will be re-started and powered-on on another host in case of a host failure, while in FT, there is no downtime because the second copy will be activated in case of host failure.

# Networking

Selasa, 21 Februari 2023 12:24 PTG

## #11. What is virtual networking?

A network of VMs running on a physical server that is connected logically with each other is called virtual networking.

## #12. What is vSS?

vSS stands for Virtual Standard Switch is responsible for the communication of VMs hosted on a single physical host. it works like a physical switch that automatically detects a VM which wants to communicate with another VM on the same physical server.

## #13. What is vDS?

vDS stands for Virtual Distributed Switch acts as a single switch in a whole virtual environment and is responsible for providing central provisioning, administration, and monitoring of the virtual network.

## #14. How many maximum standard ports per host are available?

4096 ports per host are available either in a standard switch or distributed switch.

## #15. What are the main benefits of a distributed switch (vDS)?

vDS can provide:

- The central administration for a virtual data center
- Central provision, and
- Monitoring

## #16. What is the VMKernel adapter, and why is it used?

VMKernel adapter provides network connectivity to the ESXi host to handle network traffic for vMotion, IP Storage, NAS, Fault Tolerance, and vSAN. For each type of traffic, such as vMotion, vSAN, etc. separate VMKernel adapter should be created and configured.

## #17. What is the main use of port groups in data center virtualization?

You can segregate the network traffic using port groups such as vMotion, FT, management traffic, etc.

## #18. What are the three-port groups configured in ESXi networking?

- Virtual Machine Port Group – Used for Virtual Machine Network
- Service Console Port Group – Used for Service Console Communications
- VMKernel Port Group – Used for VMotion, iSCSI, NFS Communications

## #19. What is VLAN, and why use it in virtual networking?

A logical configuration on the switch port to segment the IP Traffic where each segment cannot communicate with other segments without proper rules is called VLAN. Every VLAN has a proper number called VLAN ID.

#### **#20. What is VLAN Tagging?**

The practice of inserting VLAN ID into a packet header to identify which VLAN packet belongs to is called VLAN tagging.

#### **#21. What are the three network security policies/modes on vSwitch?**

- Promiscuous mode
- MAC address change
- Forged transmits

#### **#22. What is the promiscuous mode on vSwitch?**

Promiscuous mode is a security policy that can be defined at the virtual switch or portgroup level in vSphere ESX/ESXi. A virtual machine, Service Console, or VMkernel network interface in a portgroup that allows the use of promiscuous mode can see all network traffic traversing the virtual switch.

By default, a guest operating system's virtual network adapter only receives frames that are meant for it. Placing the guest's network adapter in promiscuous mode causes it to receive all frames passed on the virtual switch that is allowed under the VLAN policy for the associated portgroup. This can be useful for intrusion detection monitoring or if a sniffer needs to analyze all traffic on the network segment.

#### **#23. What is MAC address changes network policy?**

The security policy of a virtual switch includes a MAC address change option. This option affects the traffic that a virtual machine receives.

When the Mac address changes option is set to Accept, ESXi accepts requests to change the effective MAC address to a different address than the initial MAC address.

When the Mac address changes option is set to Reject, ESXi does not honor requests to change the effective MAC address to a different address than the initial MAC address. This setting protects the host against MAC impersonation.

#### **#24. What is the Forged transmits network policy?**

The Forged transmits option affects traffic that is transmitted from a virtual machine.

When the Forged transmits option is set to Accept, ESXi does not compare source and effective MAC addresses.

# Virtual Storage (Datastore)

Selasa, 21 Februari 2023 12:30 PTG

#32. What is a **datastore**?

A datastore is a storage location where virtual machine files are stored and accessed. Datastore is based on a file system which is called VMFS, NFS.

#33. What is the **.vmx** file?

It is the configuration file of a VM

#34. What information **.nvram** file store?

It stores BIOS-related information of a VM.

#35. What **.vmdk** file do and used?

vmdk is a VM disk file and stores data of a VM. It can be up to 62 TB in size in the vSphere 5.5 and onward versions.

#36. How many **disk types** are in VMware?

There are three disk types in vSphere.

- **Thick Provisioned Lazy Zeroes:** every virtual disk is created by default in this disk format. Physical space is allocated to a VM when a virtual disk is created. It can't be converted to a thin disk.
- **Thick Provision Eager Zeroes:** this disk type is used in VMware Fault Tolerance. All required disk space is allocated to a VM at the time of creation. It takes more time to create a virtual disk compare to other disk formats.
- **Thin provision:** It provides an on-demand allocation of disk space to a VM. When data size grows, the size of a disk will grow. Storage capacity utilization can be up to 100% with thin provisioning.

#37. What is **Storage vMotion**?

It is similar to traditional vMotion; in Storage vMotion, a virtual disk of a VM is moved from one datastore to another. During Storage vMotion, virtual disk types think provisioning disk can be transformed to thin-provisioned disk.

# VSAN

Selasa, 21 Februari 2023 12:32 PTG

## VSAN Interview Questions

### #53. What is vSAN?

Virtual SAN is software-defined storage first introduced in vSphere 5.5 and is fully integrated with vSphere. It aggregates locally attached storage of ESXi hosts, which are part of a cluster, and creates a distributed shared solution.

### #54. What is cold migration?

To move a powered-off VM from one host to another is called cold migration.

### #55. What is Storage vMotion?

To move a powered-on VM from one datastore to another is called Storage vMotion.

### #56. What are the different configuration options for VSAN?

There are two configuration options for vSAN:

- Hybrid: Uses both flash-based and magnetic disks for storage. Flash are used for caching, while magnetic disks are used for capacity or storage.
- All-Flash: Uses flash for both caching and for storage

### #57. Are there VSAN ready nodes available in the market?

Yes, vSAN-ready, such as VxRail 4.0 and 4.5, are available in the market. VxRail is the combination of min 3 servers that are part of a cluster and can scale up to 64 servers.

### #58. How minimum servers/hosts are required to configure vSAN?

To configure a vSAN, you should have a minimum of 3 ESXi hosts/servers in the form of a vSAN cluster. If one of the servers fails, a vSAN cluster will fail.

### #59. How many maximum ESXi hosts allowed for vSAN?

64 hosts are max allowed to configure a vSAN cluster.

### #60. How many disk groups and max magnetic disks are allowed in a single disk group?

A maximum of 5 disk groups are allowed on an ESXi host, which is a part of a vSAN cluster, and a maximum of 7 magnetic and 1 SSD per disk group is allowed.

### #61. How many types of storage can we use in our virtual environment?

- Direct Attached Storage
- Fiber Channel (FC)
- iSCSI
- Network Attached Storage (NAS)

### #62. What is NFS?

Network File System (NFS) is a file-sharing protocol that ESXi hosts use to communicate with a NAS device. NAS is a specialized storage device that connects to a network and can provide file access services to ESXi hosts.

### #63. What is Raw Device Mapping (RDM)?

Raw Device Mapping (RDM) is a file stored in a VMFS volume that acts as a proxy for a raw

physical device. RDM enables you to store virtual machine data directly on a LUN. RDM is recommended when a VM must interact with a real disk on the SAN.

#### #64. What is iSCSI storage?

An iSCSI SAN consists of an iSCSI storage system, which contains one or more storage processors. TCP/IP protocol is used to communicate between host and storage array. An iSCSI initiator is configured with the ESXi host. An iSCSI initiator can be hardware-based, either dependent or independent, and software-based is known as an iSCSI software initiator.

#### #65. What is the format of iSCSI addressing?

It uses TCP/IP to configure.

#### #66. What are iSCSI naming conventions?

iSCSI names are formatted in two different ways:

- the iSCSI qualified name (IQN)
- extended unique identifier (EUI)

### vApp Interview Questions

#### #67. What is vApp?

vApp is a container or group where more than one VM can be package and manage multi-tiered applications for specific requirements; for example, Web server, database server, and application server can be configured as a vApp and can be defined their power-on and power-off sequence.

#### #68. What settings can be configured for vApp?

We can configure several settings for vApp, such as CPU and memory allocation, and IP allocation policy, etc.

### Miscellaneous Interview Questions

#### #69. What is VMware Tanzu?

VMware Tanzu is the suite or portfolio of products and solutions that allow its customers to Build, Run, and Manage Kubernetes-controlled container-based applications. This technology is introduced in VMware vSphere 7.0.

#### #70. What is VMware DRS?

DRS stands for Distributed Resource Scheduler, which automatically balances available resources among various hosts by using clusters or resource pools. With the help of HA, DRS can move VMs from one host to another to balance the available resources among VMs.

#### #71. What are share, limit, and reservation?

**Share:** A value that specifies the relative priority or importance of a VM access to a given resource.

**Limit:** Consumption of a CPU cycle or host physical memory that cannot cross the defined value (limit).

**Reservation:** This value defines in the form of CPU or memory and must be available for a VM to start.

#### #72. What are the alarms why we use them?

An alarm is a notification that appears when an event occurs. Many default alarms exist for many inventory objects. Alarms can be created and modified using vSphere Web Client;

**#73.** What are the hot-pluggable devices that can be added while VM is running?

We can add HDDs and NIC while VM is running.

**#74.** What is a Template?

When a VM is converted into a format that can be used to create a VM with pre-defined settings is called a template. An installed VM can be converted into a template, but it cannot be powered on.

**#75.** What is Snapshot?

To create a copy of a VM with the timestamp as a restore point is called a snapshot. Snapshots are taken when an upgrade or software installation is required. For better performance, a snapshot should be removed after a particular task is performed.

**#76.** How to convert a physical machine into a VM?

Three steps are required to convert a physical machine to a VM:

- An agent needs to be installed on the Physical machine
- VI client needs to be installed with Converter Plug-in
- A server to import/export virtual machines

**#77.** What is vMotion, and what is the main purpose of using it in a virtual environment?

It is a very prominent feature of VMware vSphere used to live migrate running VMs from one ESXi host to another without any downtime. Datastores and ESXi hosts can both be used while vMotion.

**#78.** What is the difference between a clone and a template?

A clone is a copy of a virtual machine. Cloning a VM will save time if multiple VMs with the same configurations are required to configure. While a template is a master copy of an image created from a VM, which can be later used to create many clones. After converting a VM to a template, it can't be powered-on or edited.

**#79.** What monitoring method is used in vSphere HA?

- Network Heartbeat
- Datastore Heartbeat

**#80.** How is the master host elected in vSphere HA?

When HA is enabled in a cluster, all hosts take part in a selection process to be selected as a master host. A host which has the highest number of datastores mounted will be selected as a master host. All other hosts will remain slave hosts.

**#81.** What is the purpose of VMware Tools?

It is a suite of utilities that are used to enhance the performance of a VM in the form of graphics, mouse/keyboard movement, network card, and other peripheral devices.

**#82.** What is VMware DPM?

Stands for Distributed Power Management is a feature of VMware DRS that is used to monitor required resources in a cluster. When the resources are decreases due to low usage, VMware DPM consolidates workloads and shut down the hosts which are not being used, and when resources are increased it automatically power on the un-used hosts.

**#83.** What is the ESXi Shell?

It is a command-line interface. It is used to run the repair and diagnostics of ESXi hosts. It can be accessed via DCUI, vCenter Server enables/disable, and via [SSH](#).

#### #84. How to run ESXTOP on the ESXi host?

To run ESXTOP on an ESXi host, we'll need two pre-requisites:

- Install vSphere Client on a host where you want to configure
- Enable SSH from DCUI by using the “Troubleshooting Options” link

#### #85. What is VMware vCenter Enhanced Linked Mode and How It Works?

VMware vCenter Server Enhanced Linked Mode (ELM) is one of the vSphere advanced features that allows connecting multiple vCenter Servers to provide a single interface where you can view, search, and manage permissions, replications of roles, policies, and licenses between multiple vCenter Servers.

It allows you to simplify enterprise virtual environments deployed in the same or multiple sites with multiple vCenter Server while deploying vCenter Server as VCSA or Windows Servers.

# VCenter

Selasa, 21 Februari 2023 12:33 PTG

## #25. What are the main components of vCenter Server architecture?

vCenter Server provides a centralized platform for management, operation, resource provisioning, and performance evaluation of virtual machines and hosts.

When you deploy the vCenter Server Appliance, vCenter Server, the vCenter Server components, and the authentication services are deployed on the same system.

The following components are included in the vCenter Server appliance deployments:

- The authentication services contain vCenter Single Sign-On, License service, Lookup Service, and VMware Certificate Authority.
- The vCenter Server group of services contains vCenter Server, vSphere Client, vSphere Auto Deploy, and vSphere ESXi Dump Collector. The vCenter Server appliance also contains the VMware vSphere Lifecycle Manager Extension service and the VMware vCenter Lifecycle Manager.

## #26. What are PSC and its components?

PSC stands for Platform Services Controller, first introduced in version 6 of [VMware vSphere](#), which handles infrastructure security functions. It has three main components.

- Single Sign-On (SSO)
- VMware Certificate Authority (CA)
- Licensing service

## #27. What are the two main deploying methods of PSC?

You can install PSC in VMware vSphere 6.7 in two ways:

- Embedded
- External

But, in VMware vSphere 7.0, we can install PSC only in Embedded mode; External PSC deployment has been deprecated in VMware vSphere 7.0 or onwards.

## #28. What are the different types of vCenter Server deployment?

It has two deployment types till VMware vSphere 6.7.

- Embedded Deployment
- External deployment

In VMware vSphere 7.0 and onwards, External PSC has been deprecated. We can only install PSC in Embedded mode.

## #29. What is vRealize Operation (vROP)

vROP provides the operation dashboards for performance analytics, capacity optimization, and monitoring the virtual environment.

## #30. What is vCloud Suite?

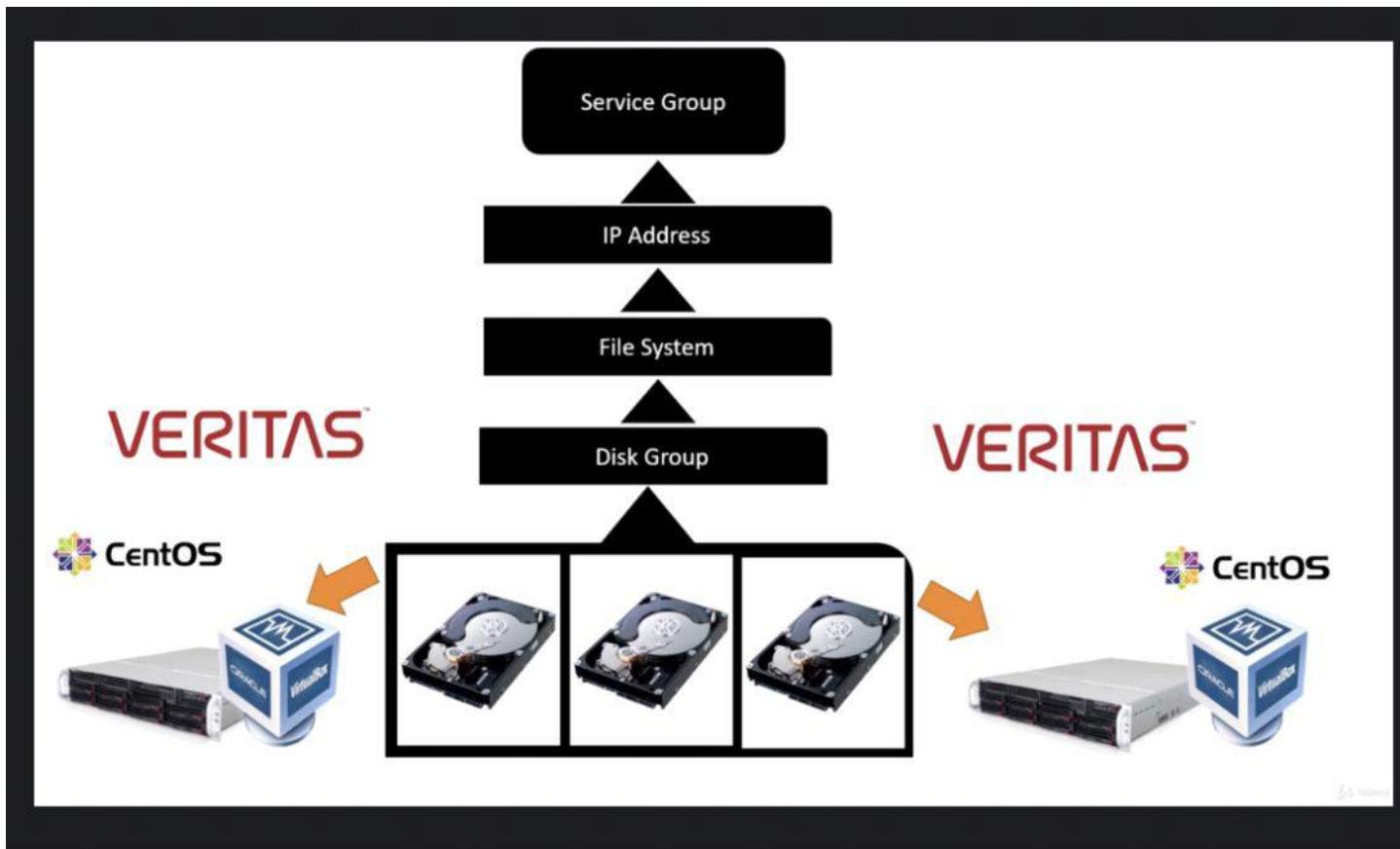
vCloud Suite combines multiple VMware components to give a complete set of cloud infrastructure capabilities in a single package, including virtualization, software-defined datacenter services, disaster recovery, application management, etc.

**#31.** What is the basic security step to secure vCenter Server and users?

Authenticate vCenter Server with Active Directory. By using this, we can assign specific roles to users and can also efficiently manage the virtual environment.

# VCS

Friday, 3 February 2023 1:44 PM



# VCS Task

Friday, 3 February 2023 10:25 AM

## **(b) Coming to Veritas Volume Manager**

(1) We get requests from production, database, QA people like  
creating volumes,  
file system creation,  
increase and (or) decrease the volume sizes,  
provide permissions,  
redundancy,  
put the volume into cluster to provide high availability,

(2) sometimes destroy or remove the volumes, backup and restore whenever necessary,

(3) We also get some troubleshooting issues like  
volume not started,  
volume not accessible,  
file system crashed,  
mount point deleted,  
disks failed,  
volume manager daemons are not working,  
configuration files missed, crashed,  
disk groups not deporting and not importing,  
volume started but users are unable to access file systems on those volumes..etc.,

## **(c) Coming to Veritas Cluster :**

(1) We get requests like node adding, resource adding, service group adding, adding service groups and resources to existing service groups, mount points adding, adding NIC cards, IP addresses, adding volumes, disk groups, freezing and unfreezing services groups and also

# Shared disk - Volume Group

Monday, 30 January 2023 7:35 PM

Volume groups also known as disk groups

Shared storage in the back end in order for VCS to work

# Start stop VCS

Tuesday, 31 January 2023 1:02 PM

## Starting and stopping VCS

This section describes how to start and stop the VCS.

### To start VCS

- On each node, start VCS:

```
# hastart
```

### To stop VCS

- On each node, stop VCS:

```
# hastop -local
```

You can also use the command **hastop -all** to stop the VCS cluster on all the nodes in cluster at the same time;

# Network

Monday, 30 January 2023 7:59 PM

Node1

Primary IP: **192.168.100.85**

Private IP 1: 192.168.0.101

Private IP 2: 192.168.0.102

Node2

Primary IP: **192.168.100.86**

Private IP 1: 192.168.0.201

Private IP 2: 192.168.0.202

VCS Service Group IP

Service Group IP: **192.168.100.150**

Gateway: 192.168.100.1

## **Cluster information verification:**

Cluster Name: vcs-cluster

Cluster ID Number: 57451

Private Heartbeat NICs for node1:

link1=enp0s8

link2=enp0s9

Low-Priority Heartbeat NIC for node1:

link-lowpri1=enp0s3

Private Heartbeat NICs for node2:

link1=enp0s8

link2=enp0s9

Low-Priority Heartbeat NIC for node2:

link-lowpri1=enp0s3

# Cmd - main.cf logs

Tuesday, 31 January 2023 11:30 AM

Default location: `/opt/VRTSvcs/bin`

```
# ./hastatus -sum
```

The `main.cf` file contains the configuration of the entire cluster and is located in the directory `/etc/VRTSvcs/conf/config`.

```
# haconf -dump -makero  (Dumps in memory configuration to main.cf and makes it read-only)  
# haconf -makerw
```

## syntax of the main.cf file

```
# hacf -verify /etc/VRTSvcs/conf/config
```

## log file

VCS cluster engine logs is located at `/var/VRTSvcs/log/engine_A.log`

```
# hamsg engine_A
```

# Service Group

Tuesday, 31 January 2023 11:45 AM

Service groups can be one of the 3 type :

1. **Failover** – Service group runs on one system at a time.
2. **Parallel** – Service group runs on multiple systems simultaneously.
3. **Hybrid** – Used in replicated data clusters (disaster recovery setups). SG behaves as Failover within the local cluster and Parallel for the remote cluster.

# Cleanup default service group

Tuesday, 31 January 2023 11:38 AM

```
# cd /etc/VRTSvcs/conf/config/  
  
# hagrp -resources ClusterService  
webip  
csgnic  
  
# haconf -makerw  
  
# hares -delete csgnic  
  
# hares -delete webip  
  
# hagrp -delete ClusterService  
  
# haconf -dump -makero  
  
# hastatus -sum  
  
-- SYSTEM STATE  
-- System      State      Frozen  
  
A node1      RUNNING      0  
A node2      RUNNING      0
```

# Create Service Group

Tuesday, 31 January 2023 11:46 AM

```
# hastatus -sum  
# haconf -makerw  
# hagrp -add SG1  
# hagrp -modify SG1 SystemList node1 0 node2 1  
# hagrp -modify SG1 AutoStartList node1 node2  
# haconf -dump -makero
```

# Assign IP to Service Group

Tuesday, 31 January 2023 11:50 AM

Assign IP to SG and create dependencies

```
# hastatus -sum  
  
# haconf -makerw  
  
# hares -add SG-ip IP SG1  
  
# hares -modify SG-ip Enabled 1  
  
# hares -modify SG-ip Device enp0s3  
  
# hares -modify SG-ip Address "192.168.100.150"  
  
# hares -modify SG-ip NetMask "255.255.255.0"  
  
# hares -add SG-nic NIC SG1  
  
# hares -modify SG-nic Device enp0s3  
  
# hares -modify SG-nic Enabled 1  
  
[root@node1 config]# hares -link SG-ip SG-nic  
  
[root@node1 config]# haconf -dump -makero
```

# Storage

Tuesday, 31 January 2023 12:04 PM

# Shared storage Volumes for VCS

Tuesday, 31 January 2023 12:00 PM

```
# vxdisk list
DEVICE    TYPE    DISK    GROUP    STATUS
node1_disk_0 auto:LVM - - LVM [REDACTED]
node1_disk_1 auto:none - - online invalid
node1_disk_2 auto:none - - online invalid
node1_disk_3 auto:none - - online invalid
```

```
[root@node1 config]# vxdisksetup -i node1_disk_1
[root@node1 config]# vxdisksetup -i node1_disk_2
[root@node1 config]# vxdisksetup -i node1_disk_3
```

```
[root@node1 config]# vxdisk list
DEVICE    TYPE    DISK    GROUP    STATUS
node1_disk_0 auto:LVM - - LVM [REDACTED]
node1_disk_1 auto:cdsdisk - - online
node1_disk_2 auto:cdsdisk - - online
node1_disk_3 auto:cdsdisk - - online
```

On Node 2

```
[root@node2 ~]# vxdctl enable
```

```
[root@node2 ~]# vxdisk list
DEVICE    TYPE    DISK    GROUP    STATUS
node2_disk_0 auto:cdsdisk - - online
node2_disk_1 auto:LVM - - LVM [REDACTED]
node2_disk_2 auto:cdsdisk - - online
node2_disk_3 auto:cdsdisk - - online
```

# DiskGroup in VCS

Tuesday, 31 January 2023 12:05 PM

```
# vxdg init DG1 disk1=node1_disk_1
[root@node1 config]# [REDACTED]
[root@node1 config]# vxdisk list
DEVICE      TYPE      DISK      GROUP      STATUS
node1_disk_0 auto:LVM   -        -          LVM[REDACTED]
node1_disk_1 auto:cdsdisk disk1    DG1       online
node1_disk_2 auto:cdsdisk -        -          online[REDACTED]
node1_disk_3 auto:cdsdisk -        -          online
```

On Node 2

```
# vxdctl enable
# vxdisk list
DEVICE      TYPE      DISK      GROUP      STATUS
node2_disk_0 auto:cdsdisk -        (DG1)     online
node2_disk_1 auto:LVM   -        -          LVM[REDACTED]
node2_disk_2 auto:cdsdisk -        -          online
node2_disk_3 auto:cdsdisk -        -          online
```

# Filesystem on VCS

Tuesday, 31 January 2023 12:09 PM

```
# mkdir /testvcsfs  
# vxassist -g DG1 make testvcsfs_lv 100M  
# mkfs -t vxfs /dev/vx/rdsk/DG1/testvcsfs_lv
```

# Adding resource to VCS control

Tuesday, 31 January 2023 12:16 PM

```
[root@node1 config]# haconf -makerw
```

```
[root@node1 config]# hares -add DG1 DiskGroup SG1  
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
```

```
[root@node1 config]# hares -modify DG1 DiskGroup DG1
```

```
[root@node1 config]# hares -add testvcsfs_lv Mount SG1  
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
```

```
[root@node1 config]# hares -modify testvcsfs_lv BlockDevice /dev/vx/dsk/DG1/testvcsfs_lv
```

```
[root@node1 config]# hares -modify testvcsfs_lv FSType vxfs  
[root@node1 config]# hares -modify testvcsfs_lv FsckOpt "%-y"  
[root@node1 config]# hares -modify testvcsfs_lv MountPoint /testvcsfs
```

```
[root@node1 config]# hares -link testvcsfs_lv DG1
```

```
[root@node1 config]# hagrp -enableresources SG1
```

```
[root@node1 config]# haconf -dump -makero
```

```
[root@node1 config]# hares -online testvcsfs_lv -sys node1
```

```
[root@node1 config]# hastatus -sum
```

```
[root@node1 config]# df -h /testvcsfs/  
Filesystem           Size  Used Avail Use% Mounted on  
/dev/vx/dsk/DG1/testvcsfs_lv 100M  3.2M  91M  4% /testvcsfs
```

# Failover

Tuesday, 31 January 2023 12:26 PM

```
# hagrp -switch SG1 -to node2
```

Node 2

```
# df -h /testvcsfs/
Filesystem           Size  Used Avail Use% Mounted on
/dev/vx/dsk/DG1/testvcsfs_lv 100M  3.2M  91M  4% /testvcsfs
[root@node2 ~]# 
```

[root@node2 ~]# hastatus -sum

```
-- SYSTEM STATE
-- System      State      Frozen
```

A node1	RUNNING	0
A node2	RUNNING	0

```
-- GROUP STATE
```

Group	System	Probed	AutoDisabled	State
B SG1	node1	Y	N	OFFLINE
B SG1	node2	Y	N	ONLINE

# How to

Friday, 3 February 2023 1:54 PM

## How to list all the resource dependencies

To list the resource dependencies :

```
# hares -dep
```

## How to enable/disable a resource ?

```
# hares -modify [resource_name] Enabled 1      (To enable a resource)
# hares -modify [resource_name] Enabled 0      (To disable a resource)
```

## How to list the parameters of a resource

To list all the parameters of a resource :

```
# hares -display [resource]
```

### Service group operations

## How to add a service group(a general method) ?

In general, to add a service group named SG with 2 nodes (node01 and node02) :

```
haconf -makerw
hagrp -add SG
hagrp -modify SG SystemList node01 0 node02 1
hagrp -modify SG AutoStartList node02
haconf -dump -makero
```

## How to check the configuration of a service group – SG ?

To see the service group configuration :

```
# hagrp -display SG
```

## How to bring service group online/offline ?

To online/offline the service group on a particular node :

```
# hagrp -online [service-group] -sys [node]      (Online the SG on a particular node)
# hagrp -offline [service-group] -sys [node]      (Offline the SG on particular node)
```

The -any option when used instead of the node name, brings the SG online/offline based on SG's failover policy.

```
# hagrp -online [service-group] -any  
# hagrp -offline [service-group] -any
```

## How to switch service groups ?

The command to switch the service group to target node :

```
# hagrp -switch [service-group] -to [target-node]
```

## How to freeze/unfreeze a service group and what happens when you do so ?

When you freeze a service group, VCS continues to monitor the service group, but does not allow it or the resources under it to be taken offline or brought online. Failover is also disabled even when a resource faults. When you unfreeze the SG, it starts behaving in the normal way.

To freeze/unfreeze a Service Group temporarily :

```
# hagrp -freeze [service-group]  
# hagrp -unfreeze [service-group]
```

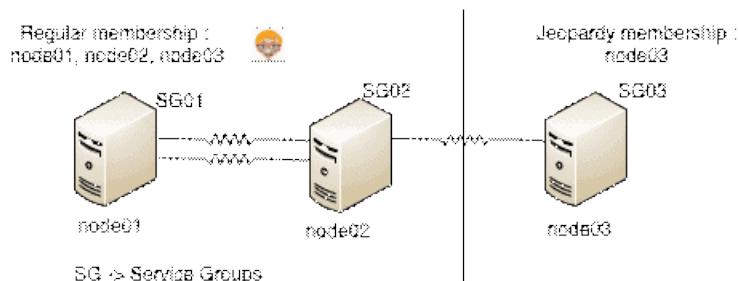
To freeze/unfreeze a Service Group persistently (across reboots) :

```
# hagrp -freeze -persistent[service-group]  
# hagrp -unfreeze [service-group] -persistent
```

Communication failures : Jeopardy, split brain

## What's a Jeopardy membership in vcs clusters

When a node in the cluster has only the last LLT link intact, the node forms a regular membership with other nodes with which it has more than one LLT link active and a Jeopardy membership with the node with which it has only one LLT link active.



Effects of jeopardy : (considering example in diagram above)

1. Jeopardy membership formed only for node03
2. Regular membership between node01, node02, node03
3. Service groups SG01, SG02, SG03 continue to run and other cluster functions remain unaffected.
4. If node03 faults or last link breaks, SG03 is not started on node01 or node02. This is done to avoid data corruption, as in case the last link is broken the nodes node02 and node01 may think that node03 is down and try to start SG03 on them. This may lead to data corruption as same service group may be online on 2 systems.
5. Failover due to resource fault or operator request would still work.

## Overview

Tuesday, 7 February 2023 12:03 AM

# An overview of Cloud Security Basics

This course is the first in the University of Minnesota's 4-part Cloud Security specialization.

We start the course with a deceptively simple and secure web service and we address the problems arising as we improve it. The improvements take us through a series of architectural steps. The improvements take us through a series of architectural steps. Each step makes strategic choices that both improve functionality and open security vulnerabilities. In response, each step introduces essential server security measures.

As we work through these scenarios, we look at relevant security incidents, the vulnerabilities behind them, their impact on affected services, and how we assess the properties of vulnerabilities. Fortunately, there are lots of incidents to choose from.

Coursework consists of the inevitable multiple choice quizzes, plus experience writing three types of cloud-related security documents.

- Service outline: you describe an online service in terms of what it does, who uses it, who provides it, and the impact of security failures.
- Vulnerability assessment: you assign a standard set of characteristics to a chosen vulnerability and justify your choices.
- Basic security plan: you create a high-level design of a cloud service and describe the security measures it requires.

## About the Instructor: Rick Smith

Twenty-five years ago, I was hired to develop security software for the US Department of Defense. Our goal was to build a truly secure computer. We failed.

I've been working in cybersecurity ever since. It's fascinating if complicated. It is constantly changing. I've written three books on cybersecurity; my textbook is releasing its third edition.

We can't make security 100% foolproof, but we can make the bad guys work harder for less benefit.

## About Cybersecurity

First of all, we try to be proactive. We try to stop problems before they occur.

Effective security controls balance efficiency with safety. They block attacks, and ideally, they don't get in the way of the real work.

We also plan for trouble. We know some security controls will fail.

Some organizations try to combine their compliance and security operations. While it's true that they have overlapping requirements, they pursue different goals. It's easy to be compliant without being secure. And vice versa.

According to Ron Joyce, who used to be the top hacker at the US National Security Agency, the NSA can get into just about anyone's network. They do it by understanding the network better than the people who built it and the ones who operate it.

It's hard to really know a network these days. People bring in personal gadgets and connect them to your network. Some people add their own switches and routers.

Not even the NSA and CIA are safe from hackers. Consider the stories of Manning and Snowden.

## The Specialization

The Cloud Security specialization contains these four courses:

- Cloud Security Basics
- Cloud Data Security
- Cloud Application Security
- Cloud Capstone Project

# Introduction

Wednesday, 15 March 2023 10:16 PM

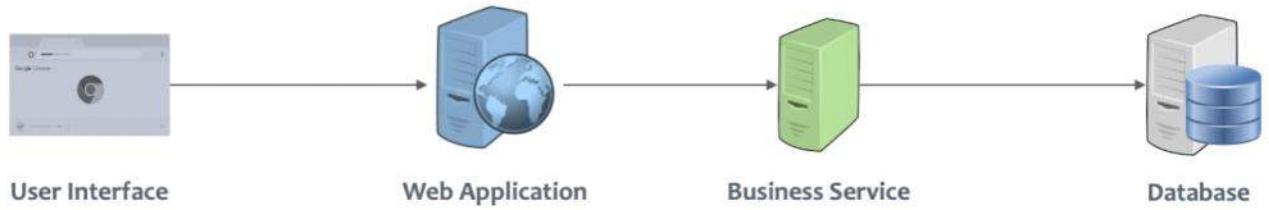
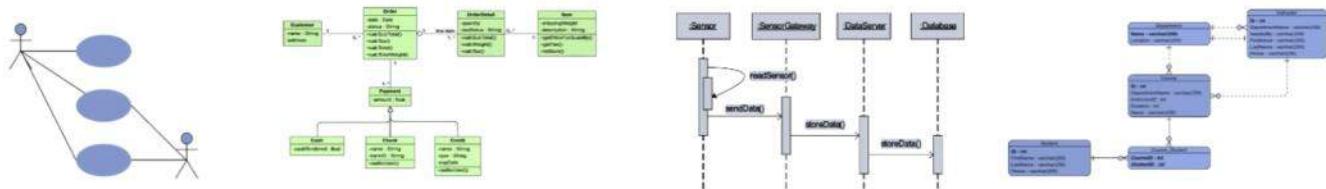
Functional Requiring  
Used Cases

How Schema is going to be

How is code design

Different Technology and languages

# Designing & Developing Functionality



# Developer To Architect

✓ Performance

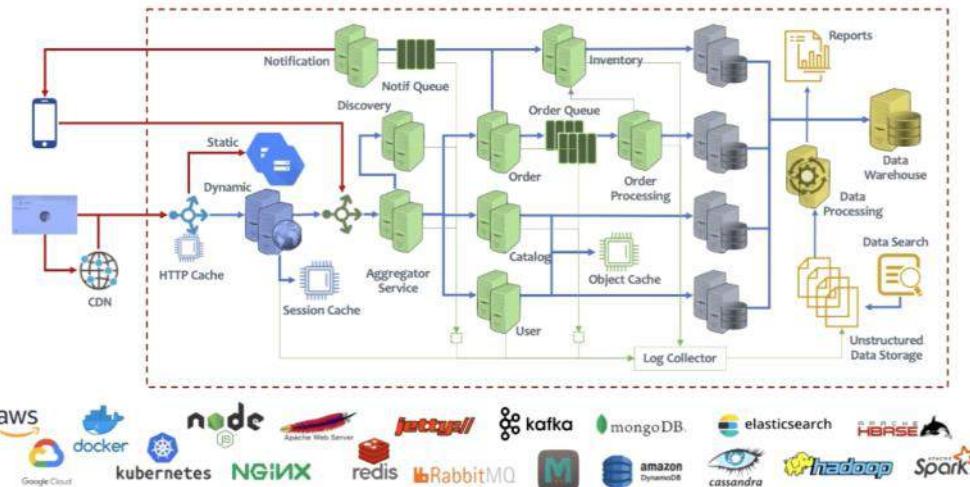
✓ Scalability

✓ Reliability

✓ Security

✓ Deployment

✓ Technology Stack



# Performance

Wednesday, 15 March 2023 10:25 PM

## System Performance

- Understanding Performance
  - Problems
  - Measurement
  - Principles
- Latency
  - CPU
  - Memory
  - Network
  - Disk
- Concurrency
  - Locking
    - Pessimistic
    - Optimistic
  - Coherence
- Caching
  - Static Data
  - Dynamic Data

Udemy

# Introduction

Isnin, 6 September 2021 2:21 PTG

I have been working in IT industry with almost 9 years of experience in numerous sectors from banking, telecommunication and other private sectors. My skills are in Linux / Unix systems / Automation / Scripting in Bash / Cloud such as AWS and AliCloud and Variety of DevOps opensource tools.

Presently I'm working with Capgemini and handling banking project for United Overseas Bank, I'm part of Managing Test Environment Service) where I handle build, deployment, testing and move to operation. Helping with automation task using bash shell scripting.

Before that I worked for Entomo Malaysia and Involved in Mysejahtera Project that's mobile application for the Government of Malaysia to facilitate contact tracing efforts in response to the COVID-19 pandemic in Malaysia.

33 million users are using this application. I worked together with Developers to implement the project based on CI/CD using Variety of Opensource tools. such as Docker swarmkit , Jenkins, GIT, Ansible Mongo, Maria, ElasticSearch and many more are there.

Previously I worked for Zebra Technologies. Where i was SME for multiple IoT products & solution own by Zebra and such as Data services, SmartPack, Location Solutions, FetchRobotics etc.

- Focusing on solving customer issues, like Escalations, responding to incidents, handling technical issues.
- automate the tasks to reducing the manual work.
- Keeps things lives like software, hardware, middleware

Before that I was working with Lenovo in Global Managed Services team where I was providing IT operations services for complex solutions like SAP HANA / BWA applications servers, Automation task and managed GPFS Cluster.

Prior to Lenovo

Dxc - I was in Unix project services and Where I was responsible for private Cloud infra support such as Centrica for (British Gas). I have worked on various Migration & new Implementation Projects in HP. for Multiple clients global such as Affin Bank, PepsiCo, BPOST, ALU, NOKIA, SuperPartner, Ericsson.

Where I have hands-on experience with Solaris, HP-UX, Containexitors, Solaris Zones, VCS Cluster, HPSA tools.

I have started my career with Soft Solvers Company that provides CRM solution where I have migrated their CRM web application and MySQL DB to AWS from Singtel Cloud hosting.

## Qualification

In Addition to this, i have a Hardware Networking & Software Engineering professional Diplomas I hold various professional certification and technical training including:

AWS solution architect, Redhat, Vmware 6.5. Some other technical trainings as well.

I'm also doing BCA (Bachelor in Computer Application)

I'm a skilled professional in Linux/Unix, Networking, AWS, CI/CD pipeline (Jenkins), Docker, GitHub, Terraform, Ansible Then Project Management tools Like Microsoft D365 , IBM Remedy, SalesForce, Snow, Jira and Agile Methodology workflow. Knowledge on Scrum & Waterfall.

## Personal

I like machines gadgets and to learn new technology.

From that I love to cook and I play some Indian classical instruments as well.

In outdoor activities, I go for fishing, hiking & scuba diving.

Now I am looking forward to take my career to next level , For my career growth and learn new work culture and environment, enhanced the new skills and technologies.

on DevOps Cloud and Solution architect to follow my dream - more deep technical automation Infrastructure skills.

## Family Background

In my family, Mom Dad and younger brother. I just recently got married in Feb. My wife is graduated in B-Tech but not currently working.

# Current Project

Sunday, 29 May 2022 11:07 AM

I as a devops engineer work with software developers to ensure smooth code releases.

We use software engineering practice that focus on the purpose of automating the project at every stage.

This helps to improve the understanding of technology stack that is used in our production environment.

Our mainly focus on team communication, resource management and team work that help resolving the issues at production env to stability of the application.

To increase the quality of product in current project I'm using

Gitlab labmirror- for version control system

I am using Jenkins For continues integration that is integrated with many other open source tools such as Gitlab labmirror- for version control system

The build tools - jdk, maven

Configuration management & deployment tool : Ansible

Containerization & Orchestration: Docker, Swarmpit & AKS

Continues Monitoring: Prometheus, Zabbix and have Grafana dashboard

For log visualization - ELK

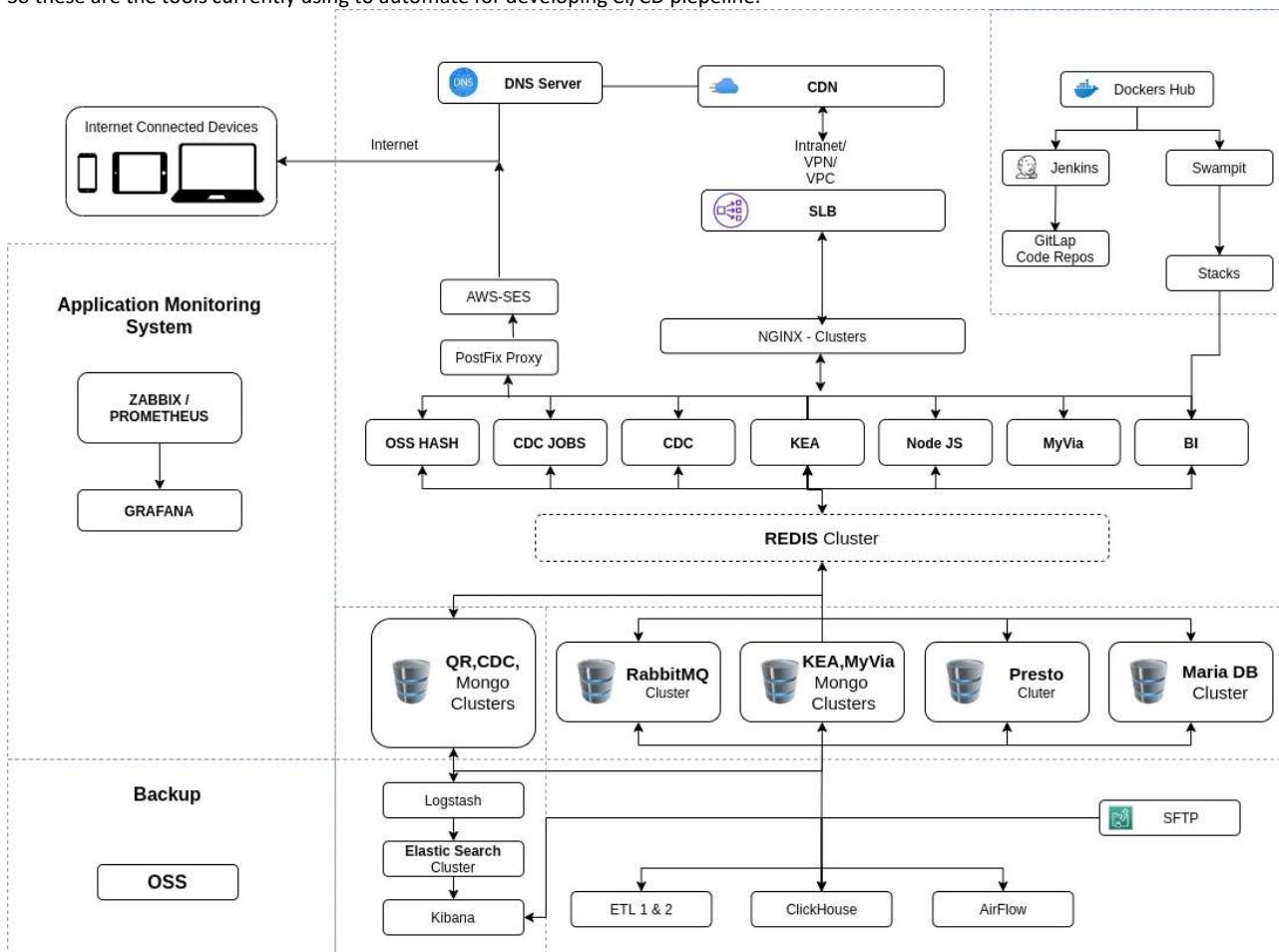
Backend platform is on Alibaba Cloud.

Using managed service such as CDN, SLB, OSS backup, Apsara Redis caching, MongoDB then MariaDB

Some services we are using from AWS as well like Route53, CodeCommit.

For docker images - using Docker Hub

So these are the tools currently using to automate for developing CI/CD pipeline.



# Tech / Prob / Communication

Selasa, 21 Februari 2023 4:05 PTG

*Technical expertise*

*to build and maintain complex systems, identify potential problems before they arise and knowledge to develop solutions quickly and efficiently.*

*Problem-solving skill*

*Which gives me to think critically and solve difficult challenges in order to ensure that the system runs smoothly and efficiently.*

*strong communication*

*to communicate effectively with other engineers, developers, and stakeholders in order to ensure that everyone has a clear understanding of the project goals and objectives.”*

# Server Troubleshooting

Selasa, 21 Februari 2023 4:41 PTG

*When troubleshooting a problem with a server,*

*I like to start by gathering as much information as possible.*

*such as system logs and any error messages that are being generated.*

*Once I have an idea of what is causing the issue, I can then begin to look into potential solutions.*

*I also make sure to document my process along the way so that if there is a need for further investigation or debugging, I have all the necessary information on hand.*

*Depending on the complexity of the issue*

# Design Deployment

Selasa, 21 Februari 2023 4:42 PTG

*I have experience in designing and deploying cloud-based solutions.*

*I have worked on a variety of projects, ranging from small scale web applications to mid-size app systems. My most recent project was MySejhatera app platform that ran entirely in the cloud.*

*I started by creating an architecture  
that would support scalability and high availability.*

*This included setting up multiple servers across different regions for redundancy and load balancing.*

*I also created automated deployment scripts to ensure rapid and consistent deployments.*

*Finally, I set up monitoring tools to track performance metrics and alert us when something went wrong.”*

# Scalability

Selasa, 21 Februari 2023 5:15 PTG

*Scalability is an important factor to consider when designing an application platform.*

*System can handle increased workloads without compromising performance or availability.*

*Scalability also allows for more efficient use of resources, as it enables applications to scale up and down depending on demand.*

*This helps reduce costs associated with maintaining a large infrastructure and keeps users happy by providing them with a reliable service.*

*Finally, scalability makes it easier to add new features and functionality to the platform, allowing businesses to stay competitive in their industry.”*

# App Performance

Selasa, 21 Februari 2023 5:16 PTG

*I have experience optimizing applications for performance.*

*I am responsible for ensuring that our applications are running optimally and efficiently. To do this, I use various tools to monitor the application's performance and identify any areas of improvement.*

*Once identified, I work with developers to implement changes that will improve the overall performance of the application. This includes things like reducing load times, improving memory usage, and increasing throughput.*

*I also have experience using profiling tools such as New Relic and AppDynamics to analyze code execution paths and identify potential bottlenecks in the system. By utilizing these tools, I can quickly pinpoint where improvements need to be made and provide actionable insights on how to optimize the application. Finally, I stay up-to-date with industry trends and best practices so that I can continue to make sure our applications are running at peak performance.”*

# Tight deadline

Selasa, 21 Februari 2023 4:46 PTG

*"I recently had to manage a large project with a tight deadline.*

*The project was to build an automated deployment pipeline for a web application. I did setting up the infrastructure on AliCloud, configuring the CI/CD pipelines using Jenkins, and deploying the application on Docker Swarmpit using Ansible.*

*I created a detailed timeline to let me keep on track and assigned tasks to each team member.*

*Having regular meetings to review progress and address any issues that arose.*

*I made sure to communicate regularly so everyone involved in project were aware of our progress.*

*Finally, I worked closely with the development team to ensure that all code changes were tested and deployed in a timely manner.*

*Thanks to my careful planning and management, we successfully completed the project within the allotted timeframe. This experience has taught me the importance of staying organized and communicating effectively when managing projects with tight deadlines."*

# If hired

Selasa, 21 Februari 2023 4:51 PTG

“If I am hired as a Platform Engineer,

my first approach would be to learn about the company’s current platform and infrastructure.

This includes understanding what technologies are used, how they are integrated, and any existing processes or procedures in place.

Once I have an understanding of the current setup, I can then assess what improvements need to be made and develop a plan for implementing them.

I also believe it is important to understand the company’s goals and objectives when it comes to their platform engineering needs. By understanding these goals, I can create solutions that will help the company reach its desired outcomes. Finally, I would take the time to get to know the team and build relationships with key stakeholders so that I can better understand their individual needs and how I can best support them.”

# Maintenance

Selasa, 21 Februari 2023 5:02 PTG

*I understand the importance of regularly performing maintenance on systems, and I strive to ensure that all my systems are running optimally. To achieve this goal, I have a set schedule for maintenance that I adhere to closely.*

*I typically perform maintenance once every two weeks, but depending on the system, I may adjust this frequency as needed. For example, if I am managing a system with high traffic or more sensitive data, I will increase the frequency of maintenance checks. During these maintenance sessions, I check for any potential issues, such as security vulnerabilities, software updates, or hardware malfunctions. If any problems arise, I take steps to address them immediately.”*

# CI CD

Selasa, 21 Februari 2023 5:12 PTG

*I have experience building automated deployments using CI/CD pipelines.*

*I have built and maintained multiple complex pipelines for various projects, ranging from web applications to mobile apps.*

*My expertise includes setting up build and deployment processes with Jenkins.*

*I am also well-versed in the best practices for creating secure and reliable pipelines. I understand the importance of having a robust testing process that is integrated into the pipeline to ensure quality code. I also make sure to use version control systems such as Git to keep track of changes and enable rollbacks if needed.”*

# Testing Env

Selasa, 21 Februari 2023 5:15 PTG

*"When setting up a staging environment for testing new features,*

*my first step would be to create an isolated copy of the production environment. This will allow me to make changes and test them without affecting the production system.*

*I would then configure the staging environment with all necessary components such as databases, web servers, application servers, etc.*

*Next, I would ensure that the staging environment is configured correctly by running tests on it.*

*This would include verifying that the correct versions of software are installed, that the server configurations are correct, and that the network connectivity is working properly.*

*Once the staging environment is set up and tested, I would deploy the latest version of the codebase into the staging environment. This would allow developers to begin testing the new features in a safe and controlled environment. Finally, I would monitor the performance of the staging environment and provide feedback to the development team if any issues arise."*

# Why you want change

Wednesday, October 13, 2021 9:39 PM

I do enjoy working at my current job but I'm looking for more responsibilities ready for a fresh challenge with a new and exciting company that has ambitious plans for the future. I'm looking to work somewhere new, where my skills qualities and experiences will be put to good use and i see my future with you here at this company.

and i'm fast learner. Whatever i had to learn is now done here and now i really want to growth.

I have thoroughly enjoyed working for my employer and we have achieved some great things whilst i have been there. I will leave on good terms. and i will definitely stay in touch with them.

## Why

"Over the years, I have gained a wide-ranging set of skills & experience in IT that, I believe, make me supportive, professional as to ensure the company achieve their commercial and financial objectives."

I take pride in my work, I take my professional development seriously.

I always focus on how I can add value to the organization by providing secure and innovative solutions based on the needs of the business.

In addition to possessing solid technical knowledge capabilities,

I have good communication, collaboration, and decision-making skills.

## Why work for our Org

DevOps is exciting field and prior to applying of this role, i research into your organization to make sure it is somewhere i want to work for.

I feel my knowledge, my skills and experience will help to achieve company's goal.

## Most Important skills

DevOps role,

three different types of skills: technical, soft and business.

The TECHNICAL SKILLS needed include coding and scripting capabilities, infrastructure knowledge, cloud and testing skills, software security skills, and also an understanding of major DevOps tools and resources from opensource.

SOFT SKILLS required include strong communication, interpersonal and collaboration capabilities, and also the ability to solve problems, be entirely flexible and adaptable in your work, and also the desire to maintain competence through continuous professional development.

BUSINESS SKILLS, it's imperative you have an understanding of how my work fits into the wider, strategic goals of the organization you are working for."

## Strengths & weaknesses

My Technical skills & expertise.

I have built up lots of experince working with variuos sector from linux/Unix to cloud and DevOps strategy.

I feel i can bring a wealth of knowledge and experience to your team.

Other strength is my soft & interpersonal skills so i can fit in team quickly

**"I would say my core strengths are my technical knowledge and expertise. I have built up lots of experience in various DevOps positions over the years, and I feel I can bring a wealth of knowledge and experience to your team."**

Other strengths include my communication and interpersonal skills. This means I can fit into a team quickly, and I will always be unselfish in my work and ensure the needs of the team and the organization always come first.

**Another strength of mine is my level of commercial awareness. I understand that, in order for your business to be successful, I have to excel in the position.**

In respect of my weakness, the only one I have is the fact I have trouble sometimes letting go of projects. I tend to get engrossed in projects and I become quite passionate about them.

**Having said that, I am learning to complete DevOps projects quickly and then move on to the next one, and I will always take onboard constructive feedback from my peers and managers in a positive way as I am someone who is keen to continually grow, learn and develop."**

## Hardest day

**"Perhaps the hardest day I have had as a DevOps Engineer was whilst working on a project for a client in a previous role. This was a cloud-based project and, despite having an initial set brief to work towards, the client continually changed the project specifications."**

Due to the client's unfortunate haphazard approach to the project requirements, the team started to show signs of stress and frustration. I spoke to the team members and explained how important it was that we still provided a high level of service.

**Although it was frustrating to have to continually change our approach to the project, this was our opportunity to dig deep, maintain flexibility and also use patience and resilience to get through the project to a satisfactory conclusion.**

**Although the project was very difficult to work on, we stuck together as a team, adapted as and when required and successfully completed everything the client wanted, on time and to the final requested specification."**

**Q. Why do you want to work for our organization in this DevOps role?**

**"For me, DevOps is a very exciting field to work in, providing of course, you choose the right organization to work for.**

Prior to applying for this DevOps role, I carried out lots of research into your organization to make sure it was somewhere I wanted to work for long-term, and also to make sure I was one hundred percent confident I could contribute positively to your goals and objectives.

**You are clearly an organization that has ambitious, exciting and diverse plans for the future, and I feel my knowledge, my skills and my experience will help you to achieve your goals.**

Finally, one of the influencing factors that made me really want to work for your organization, is the fact you employ lots of talented people. I want to work with a team of like-minded professionals who are all passionate about their work and who are also striving to achieve the same goal. For those reasons, I want to work here and nowhere else."

The more i travel and the more i growth my knowledge and experience that will help me to Expand my network. well i didn't think about why i am going Malaysia, i just saw an opportunity to move and work in Malaysia so i accepted it.

The clients i have worked with are spread in continent. I just worked with them virtually and I have already explored much in Asia and now i want to travel out from Malaysia and work with them to explore their culture, work environment. Malaysia is tropical country full year and haven't experienced working in Cold weather. It's easy to travel around Europe and UK

I 'have been to Netherlands and I like it there  
Can speak English everywhere  
Cycling dutch tradition  
Good org and infra (Stations, Transport)  
The Work life is balanced - 35 hour  
Safety - very low rate crime  
Weather

Rent - 600 - 800/Month  
Food - 500  
Transport - 80 / Monthly Pass  
Entertainment - 300

-----  
Total 1600 - 1800 - Per Month Expense

Nordcloud - IT Services provider company - 2011 - 501 to 1k  
Adyen - Financial Services that was founded in 2006 - 1k - 5k

I'm looking for more responsibilities ready for a fresh challenge with a new and exciting company that has ambitious plans for the future. I'm looking to work somewhere new, where my skills qualities and experiences will be put to good use and i see my future with you here

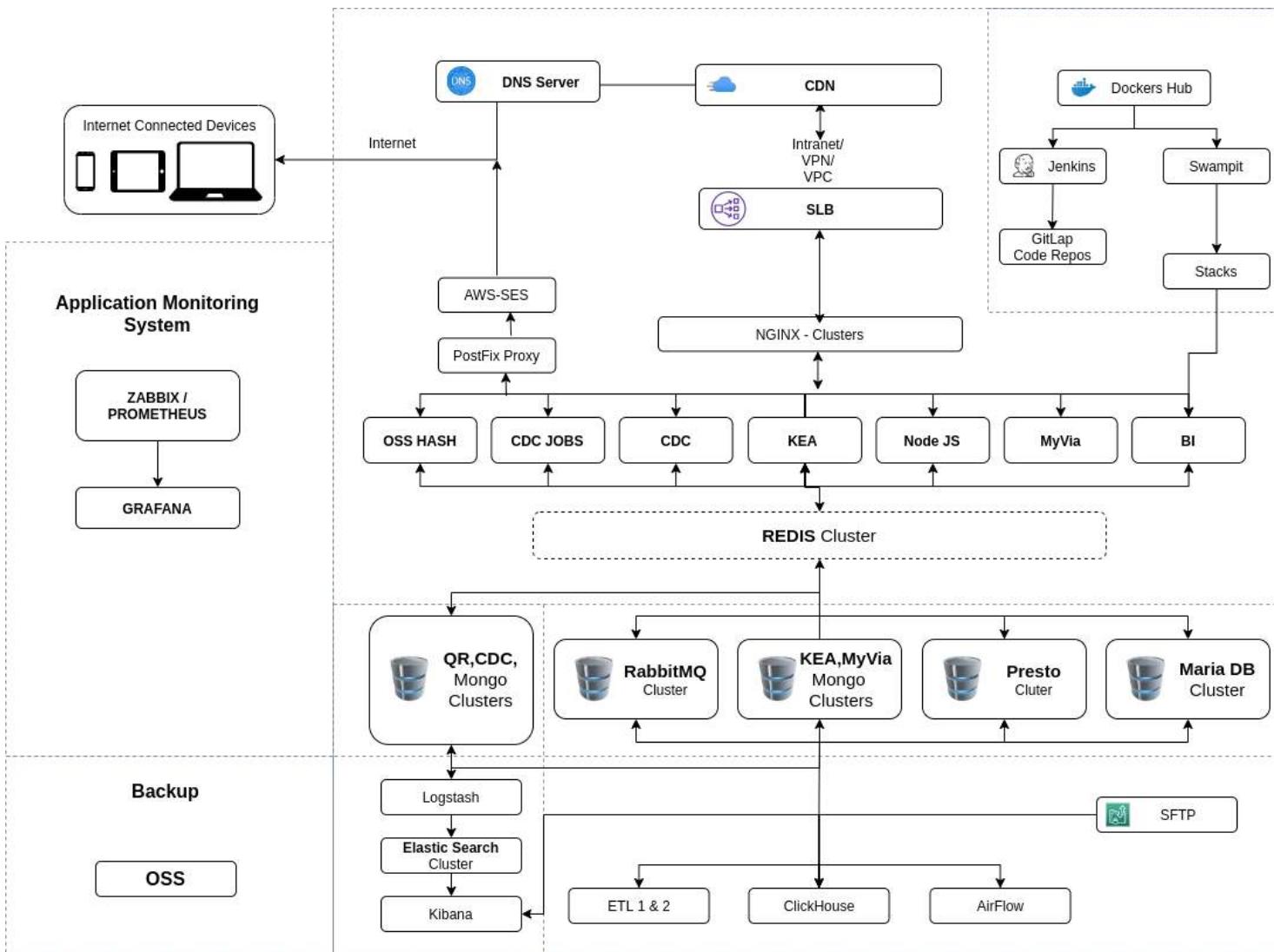
at this company. I have gained a wide-ranging set of skills & experience in IT that, I believe, make me supportive, professional as to ensure the company achieve their commercial and financial objectives. I take pride in my work, I take my professional development seriously. I always focus on how I can add value to the organization by providing secure and innovative solutions based on the needs of the business.

Hi Joseph, I hope you're doing well! I'm interested in the role you posted: DevOps Engineer. Based on my experience as Senior Consultant - DevOps at Capgemini, I believe I could be a good fit. Are you open to a quick chat to discuss the position? I'd love to learn more about it, and share more about my own qualifications. I look forward to hearing from you. Best regards, Malkiat Singh

From <<https://www.linkedin.com/feed/>>

# Project

Wednesday, May 11, 2022 11:46 AM



## Info about mysejahtera - Government Mana , Vaccine, Tracking

MySejahtera is a mobile application developed by the Government of Malaysia to facilitate contact tracing efforts in response to the COVID-19 pandemic in Malaysia.

Web, iOS, Android version for this app under government governance

For this application we highly depended on Open Source technologies.

Frontend app is running on Docker swarmpit application microservice and using DockerHub & Gitlab for code repository.

Backend on alibabacloud platform

Multi flavor databases for specific use cases like MongoDB & MariaDB

Profile on mongo and Maria is managing user data

Redis cluster for Cache service

We are also using Alibaba managed service- ApsaraDB or Mongo & Redis

Then SLB, CDN & OSS for backup purpose.

For DNS service we are using AWS route53

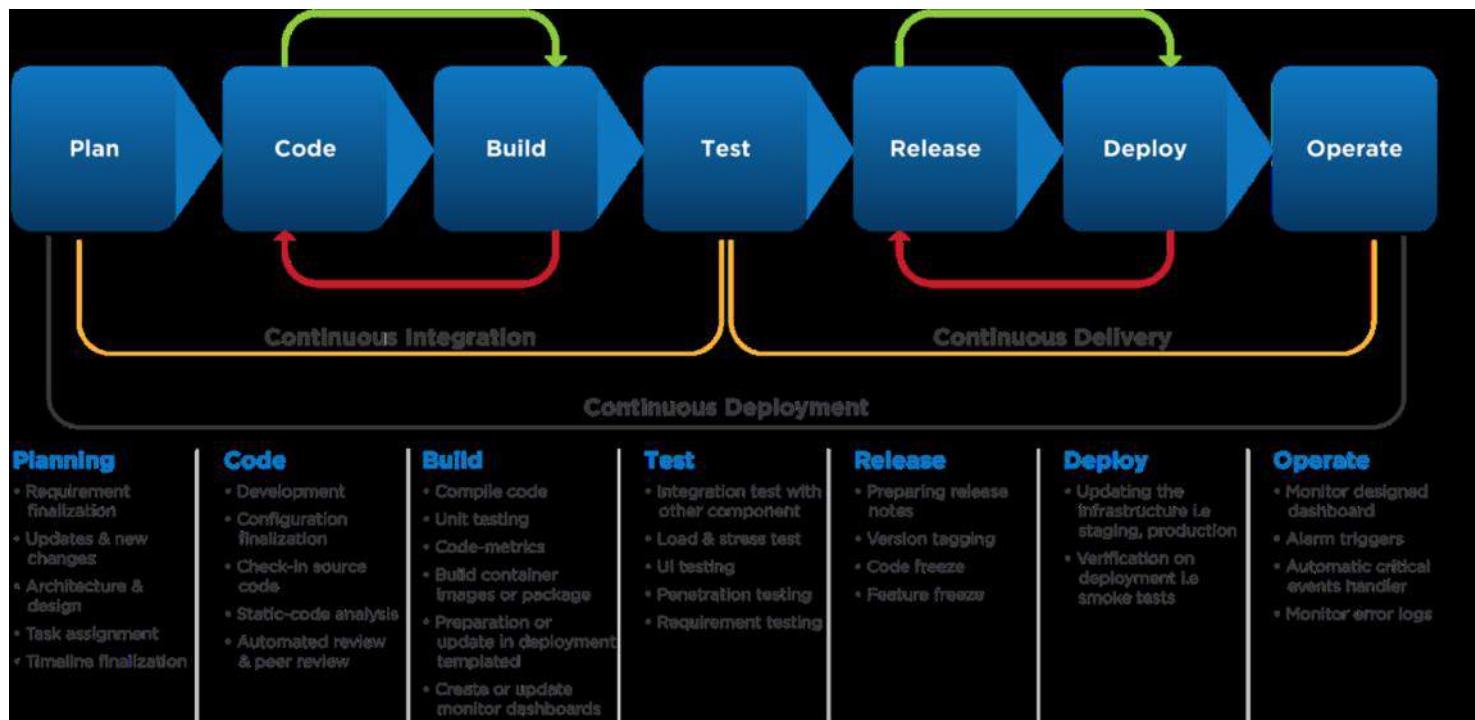
For Monitoring - Zabbix, Prometheus & Grafana  
We are using ELK for log visualization

I'm managing the DevOps and Infra

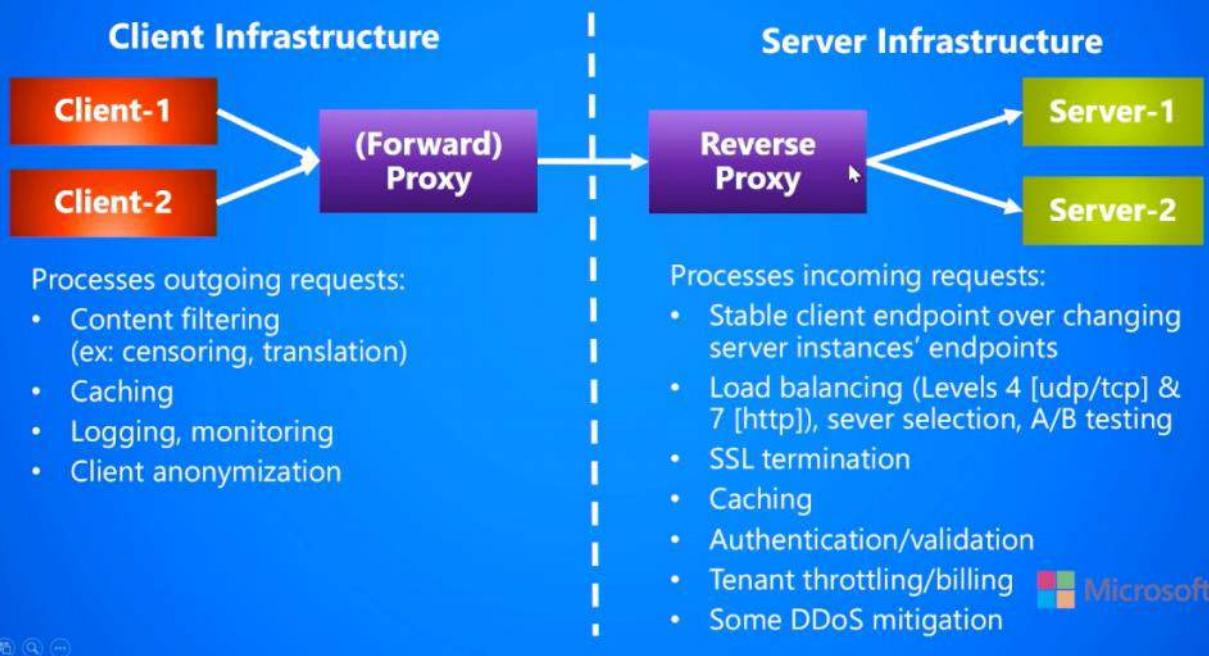
=====

Jenkins  
Pipeline  
CDC build release code

DNS Server - Route53  
CDN - Content delivery service - accelerate the response to users and increase the response rate  
SLB - Distributes network traffic  
Nginx- cluster for web server and reverse proxy  
Docker-Hub -  
Jenkins  
GitLab Code Repo  
Swarmpit  
App Stacks  
Redis Cluster - in-memory data structure store, used as a database cache, and message broker.  
Mongo cluster  
MariaDB  
ELK  
Backup OSS  
Application monitoring  
Zabbix / Prometheus  
Grafana



# Forward & reverse proxies



# Devops domain int qu

Friday, 20 May, 2022 11:32 AM

**Cloudtrail** - to get the audit data who did what in aws resources. It's expensive.

Docker restart - extension restart=always

## DevOps KPI

- Key Performance Indicators
- Deployment Frequency - Jenkins (Prod deployment How freq)
- Percentage of failed deployments - Failer in prod (Where to improve)
- meantime to failer recovery - Microservice (Pod down - Api also down, how fast can it be recover)

## Git Fetch / Pull

- git clone the repo - Fetch will update the repo with latest metadata update with original
- git pull - bring all the changes happened on remote repo

## Hashicorp Vault

- solution to manage secret and Protect Sensitive data.
- easy to intigrate with many other services such as terraform, K8s etc.
- AWS Secret manager

## Git log / what information with it helps you?

- It tells current state of your repo with commits and other info
- you can view this also in nice tree option git log --graph --oneline --decorate

## AWS Aurora - what kind of RDS used

- server less database - managed database - Mysql, PostgresDB
- backup handled by Amazon, data across AZ

## Agent and Controller in Jenkins.

- Controller -Master - Handling of scheduling jobs
- Agent - Runs the jenkins stage / part of jenkins job assigned by controller

## Artifact (Local / Remote)

- storing your final code in zip, jar (Python), wheel(JAVA) format (Compressed)
- Can't be used by anyone else in local.
- Upload artifact in central artifact repository with version title.
- K8s, dockerswarm can download the compressed file and run them

## Jenkins Security

- What kind of security are we speaking off here
- Login authentication - Default is admin, Integrate jenkins with diff LDAP services (Gmail, SOS)
- Handeling Secrets, Roles assign

## LivenessProbe, Used, Failure Threshold?

- Responsible for checking if the POD is in proper running state after deployment.
- Threshold - Given time to check POD status.

## Jenkins setup done/Managed

If our setup is scalable and can handle multi build

- 1 Master (Independent Machine EC2), 2 Agent
- Multiple Agent (Auto Scaling group on EC2) Staging
- Multiple Agent (Auto scaling group on EC2) Live
- Pipeline build it spins
- 1 Application
- 5 Services
- Nginx service reload
- Env QA, UAT, Prod
- Java Code

#### Scalabel Setup

- 30 to 40 Builds in given day
- Staging and Live

#### **tmpfs mounts in docker**

- Temp fs and persist in the host memory (Only on Linux)
- Container stops, tmpfs removed, file written wont be persisted
- Useful to temp store sensitive files that you dont want to persist in either the host or the container writable layer

S3 - Ec2 in private - how to access - endpoint

Terraform - Working together with team - prevent from state file to be used by other team members

Difference between running RUN in single and multiple

S3 full permission on group but user from same group have set no permission to S3. Result ?

How you will execute Json query from S3 ?

Terraform destroy single instance from main file.

## SRE (Site Reliability Eng)

Traditional IT role or DevOps

- Focusing on solving customer issues, like Escalations, responding to incidents, handling technical issues.
- Trying to automate the tasks and reducing the manual work.
- Keeps things lives like software, hardware, middleware

System - Servers, Database, Cloud & Virtualization, Network, App & services

Reliability - Service accessible most of time like , gmail, youtube not like ATM or email.

Important - Unhappy customer, online shopping down > Lost revenue, loss of business

When makes systems unreliable?

- Changes in infra
- Changes in platform (e.g. k8s)
- Changes in services & Applications

When makes systems reliable good?

- Change make app better
- Increase business value
- Stay competitive

Dev - Release fast

Ops - Maintain stability



SRE - Tries to automate the process of evaluating the affects the changes will have

## SRE Tasks and Responsibilities



SRE

SRE teams are made up of software engineers

who build and implement software

to improve the reliability of their systems/services



SRE

**1) Automation** - Create automated processes for operational aspects



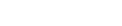
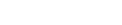
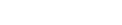
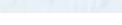
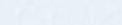
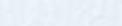
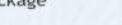
Code it



Test it



Build & Package



# Capgemini

Wednesday, 25 May 2022 4:34 PM

Hi Malkiat,

As discussed, sharing all the details for your reference. Let me know if you would like to refer any of your friends/colleagues.

We at Capgemini hire the best talent to partner with prestigious projects for our client base which includes top class Global Banks and Large MNCs.

As discussed, Capgemini is looking for **Test Environment Manager (TEM)** candidates.

**Role Summary –Key Requirement –DevOps candidates with knowledge of CI/CD and cloud.**

**Job Description-**

- Strong Knowledge on Unix, Linux / Unix/ windows Application support
- Good Knowledge in CI/CD , Devops and SCM tools - Bamboo, Ansible, Git,
- Codefresh, Jenkins, Artifactory, Udeploy, Splunk, Appdynamics, Grafana, ITRS
- Work experience in CI/CD automation, scripting – shell, perl, python..etc
- Excellent experience in troubleshooting, Debugging and Support.
- Experience in OpenShift, Kubernetes, Docker
- Work experience in GCP and AWS cloud
- Test Environment monitoring
- Good Application support experience
- Build and deployment skills – ANT, Maven

## **About Capgemini:**

Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. Present in 44 countries with more than 180,000 employees, the Capgemini Group helps its clients transform in order to improve their performance and competitive positioning.

Our company culture is based on 7 shared values. Honesty and trust allow collaboration; team spirit and modesty sustain it; and the resulting creative freedom and boldness lead to quality results –especially when infused with a sense of fun

In addition to an interesting remuneration package, we offer you a professional and international work environment where you will work on major projects. We provide you with intense professional development and stretch you as much as needed to put your skills into action, learn and progress.

<b>Regular Benefits</b>
15 days Annual leave
Medical Leave - 12 days , Hospitalization Leave - 46 days
Normal Leave (e.g. Paternity leave, funeral leave, compensatory leave)
Maternity Leave - 90 days
Up to RM1,800 dental benefit
Up to RM50k Medical Benefit for self , Kids and Dependent spouse
Up to RM350k GTL (Group Term Life Insurance)
Training and Certification
EPF Employer Contribution: 13% for below 5K 12% for above 5K

To be successful you will need thorough experience with most of:

- AWS (ideally also Azure)
- Azure DevOps CI/CD Pipelines (ideally also Jenkins)
- Terraform Infrastructure as Code (ideally also ARM & Bicep)
- Azure DevOps Code Repository management (ideally also GitHub)
- Security tooling (SAST/DAST/Build pipeline integration)
- AWS Lambda (ideally also Azure Functions)
- Architectural solution design
- ‘Secure by Design’ systems
- Agile working practices (Backlog/Refinement/Prioritisation/Retro)

To help you get there it would be good to have all, or some of:

- Serverless application stacks
- Atlassian’s Jira/Confluence
- Scripting & automation skills (Shell/Python/JavaScript)
- Container-based development
- AWS and/or Azure certification to Architect level
- Understanding of DevSecOps working practices
- Understanding of the full SDLC

# Motivation

Khamis, 9 Mac 2023 7:17 PTG

I am thrilled to have the opportunity to discuss my motivation for joining Deloitte with you. Deloitte's reputation as a leading global professional services firm, combined with its commitment to delivering innovative solutions to clients, makes it an organization that I am eager to be a part of.

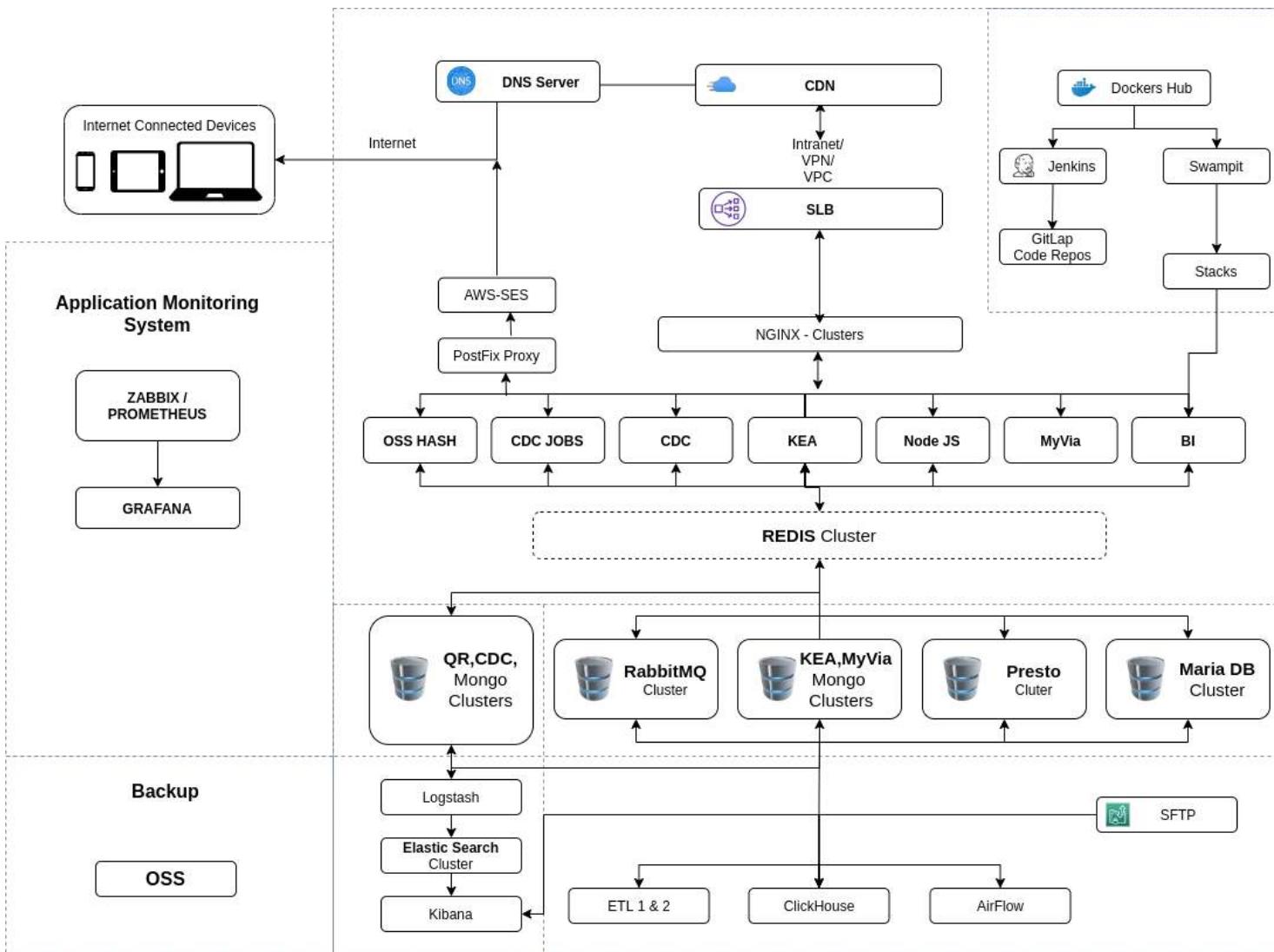
As someone with a strong passion for problem-solving and a track record of success in driving business growth, I am drawn to Deloitte's emphasis on collaboration, creativity, and excellence. I believe that my skills and experience align well with the needs of the company, and I am confident that I can contribute to Deloitte's ongoing success.

Additionally, I am excited about the prospect of working alongside talented professionals and learning from their expertise. I have no doubt that Deloitte's collaborative culture will provide me with ample opportunities to learn and grow, both professionally and personally.

Thank you again for considering me for the role. I look forward to discussing how I can contribute to Deloitte's success and working towards achieving its goals together.

# Project

Wednesday, May 11, 2022 11:46 AM



Info about mysejahtera - Govenement Mana , Vaccine, Tracking

MySejahtera is a mobile application developed by the Government of Malaysia to facilitate contact tracing efforts in response to the COVID-19 pandemic in Malaysia.

Web , Ios, Android version for this app under government governance

For this application we highly depended on Open Source technologies.

Frontend app is running on Docker swamplight application microservice and using DockerHub & Gitlab for code repository.

Backend on alibabacloud platform

Multi flavor databases for specific use cases like MongoDB & MariaDB

Profile on mongo and Maria is managing user data

Redis cluster for Cache service

We are also using Alibaba managed service- ApsaraDB or Mongo & Redis  
Then SLB, CDN & OSS for backup purpose.

For DNS service we are using AWS route53

For Monitoring - Zabbix, Prometheus & Grafana

We are using ELK for log visualization

I'm managing the DevOps and Infra

=====

Jenkins  
Pipeline  
CDC build release code

**DNS Server - Route53**

**CDN** - Content delivery service - accelerate the response to users and increase the response rate

**SLB** - Distributes network traffic

**Nginx**- cluster for web server and reverse proxy

**Docker-Hub** -

**Jenkins**

**GitLab Code Repo**

**Swarmpit**

**App Stacks**

**Redis Cluster** - in-memory data structure store, used as a database cache, and message broker.

**Mongo cluster**

**MariaDB**

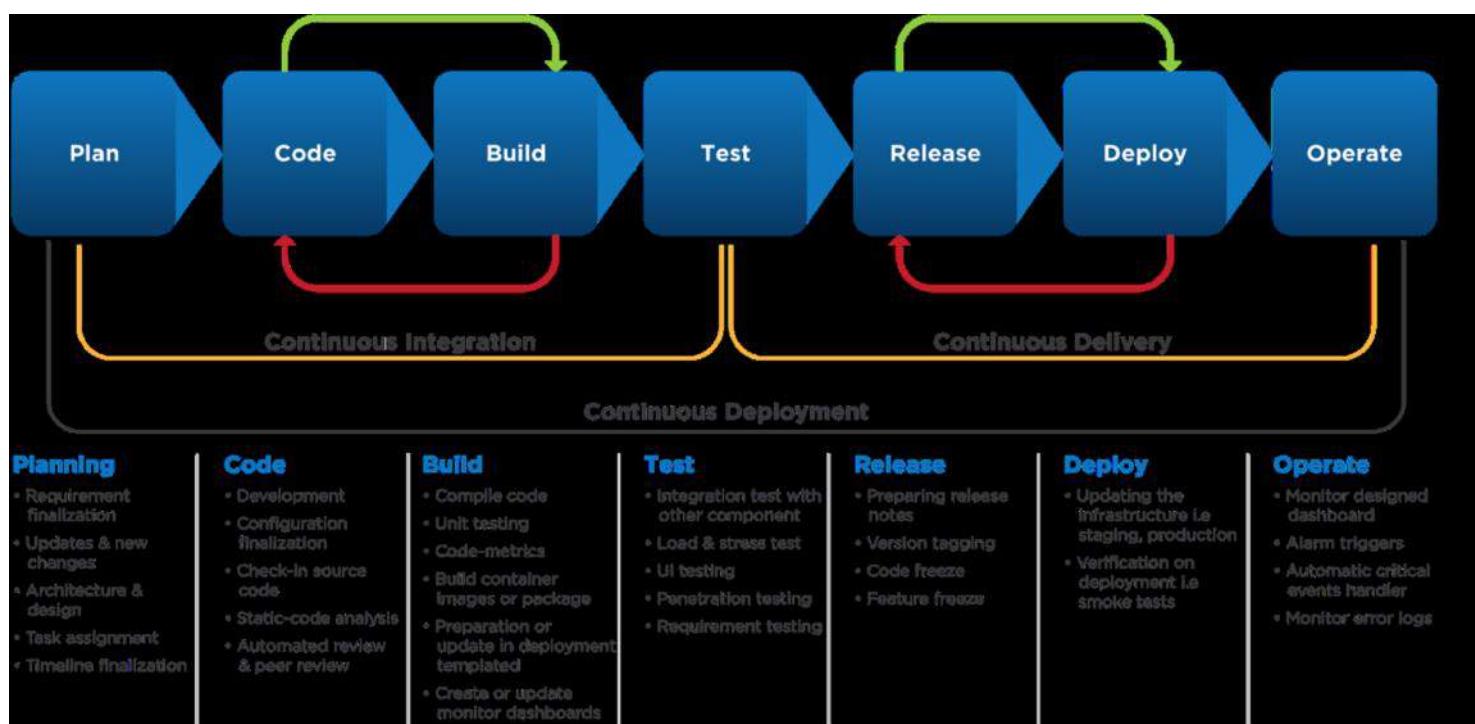
**ELK**

**Backup OSS**

**Application monitoring**

**Zabbix / Prometheus**

**Grafana**



# Forward & reverse proxies

