- **DevOps is** a combination of **Development and Operations--is a workflow that emphasizes communication between software developers and IT professionals** managing production environments.
- while automating the deployment of software and infrastructure changes.

- **Demand is high for these jobs as more enterprises**--including **Adobe, Amazon, and Target--turn** to **DevOps practices to deliver software** and **security updates more rapidly, both internally and to customers.**

What are AWS:

- **Amazon Web Services (AWS)** is a highly available, secure cloud services platform that offers more than 100 cloud applications.
- Providing a pay-as-you-go system removes the requirement for capital to be provided upfront.
- It helps in controlling, auditing, and managing identity, configuration, and usage.

How Has AWS Become So Successful?

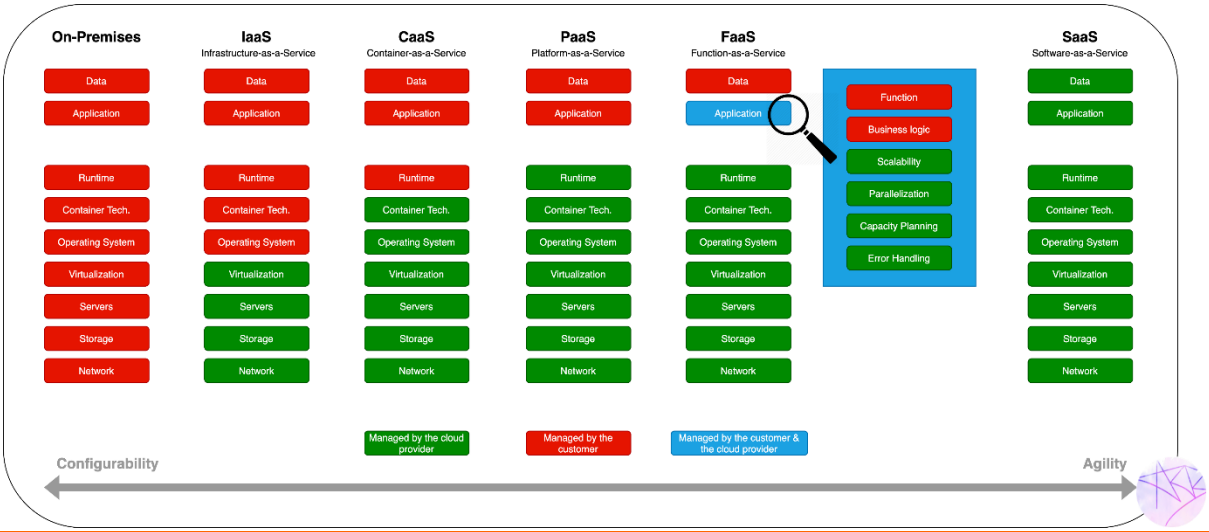
- **Security:** AWS provides a secure and durable platform that provides end-to-end security and storage.
- **Experience:** The skills and infrastructure management born from Amazon's many years of experience can be very valuable.
- **Flexibility:** It allows users to select the operating systems, language, database, and other services as per their requirements.
- **Easy to use:** AWS lets you host your applications quickly and securely, regardless of whether it's an existing or new application.
- **Scalable:** The applications you use can be scaled up or down, depending on your requirements.
- **Cost savings:** You only pay for the compute power, storage, and other resources that you use, without any long-term commitments.
- **Scheduling:** This enables you to start and stop AWS services at predetermined times
- **Reliability:** AWS takes multiple backups at servers at multiple physical locations

The difference between public and private subnet are as follows

- **A public subnet** routes 0.0.0.0/0 through an internet gateway (**IGW**).
- EC2 instances within **public subnet could connect to internet through instance public** IP.
- The instances in the public subnet could send outbound traffic to internet.
- However, all incoming request to your instance is blocked by your public subnet.

Private subnet:

- The instance within **private subnet could not connect to internet.** However, the instances could communicate with other instances within the VPC CIDR.
- AWS provides an option to allow the instance within **private subnet to connect to internet through Network Address Translation (NAT) instance or** NAT gateway.
- The traffic in private subnet is routed through NAT in the public subnet.
- You could also restrict the route to 0.0.0.0/0 to make it as a private subnet with no internet access in or out from it.



IaaS — Infrastructure-as-a-Service

There are multiple ways to look at and explain IaaS. Here we will look at the two most prominent ones: business model and online services.

As a business model

- IaaS is a business model that contrary to the classic buying of computer infrastructure ("My server is in my basement"), makes it possible to rent it when required (on demand).
- The IaaS Provider takes care of networking, storage, servers, and virtualization whereas you, as customer, are responsible for the actual application, its data, the runtime, container technology (if required) and the Operating system.

The most important features of using this approach are:

- Peak loads are absorbed.
- Sudden growth is possible without problems (scalability).
- Unused capacities can be released again immediately.
- There is no need to maintain extra infrastructure for applications that are rarely executed.
- The virtualization technology required for this enables simple software testing on a wide variety of platforms.

As Online Services

You can also view IaaS as online services that use certain cloud orchestration technologies to manage creation of virtual machines, selecting hypervisor, allocating storage volume, providing usage information for billing purposes etc.

The cloud orchestration technology used by the most public clouds are proprietary but there are multiple open-source solutions (OpenStack, Apache Cloud Stack etc.) available as well for native clouds.

It should be noted here that VM is not the only infrastructure available as a managed service. There are other resources available as well such as a raw block storage, file or object storage, firewalls, load balancers, IP addresses, virtual local area networks (VLANs) etc.

Primary use cases: single tenancy, confidential computing, more configurability required (specific kernels etc.)
Examples: EC2, S3, Google Compute Engine, GCP Global/Regional Load Balancer, Azure Load Balancer, Azure VM, Azure Storage etc.
Payment structure: Most of the clouds offer pay-per-second option. Furthermore, there are discounts available in case you want to reserve VM instances for 1- or 3-year period. There is also the possibility of cost saving through usage of so-called spot / pre-emptible instances.

These instances are available at a highly discounted price for workloads which are interruptible, since such instances can be claimed back by the IaaS vendor at any time causing your workload to be.

CaaS — Container-as-a-Service

- **What are containers?**
- Containers are an alternative to hypervisors and use several concepts from the Unix world (Control groups, Union Filesystem, Namespaces and Processes) to run your applications securely.
- Containers run in isolated partitions of a single Linux kernel running directly on the physical hardware.
- Linux Cgroups (Control Groups) and namespaces are the underlying Linux kernel technologies used to isolate, secure, and manage the containers.
- Containerisation offers higher performance than virtualization/hypervisors, because there is no hypervisor overhead.
- Also, container capacity auto-scales dynamically with computing load, which eliminates the problem of over-provisioning and enables usage-based billing.

What is CaaS?

- CaaS lets users deploy and manage containerized applications by offering specific products that either provide a simple way to run your single-container deployments, especially for running your simple microservices or provide a managed container orchestration platform like Kubernetes for running more complex multi-container deployments.
- The container orchestration platforms provide services like service discovery, container scheduling, container networking, monitoring etc. In either case, the container runtime and other container related services are provided by the CaaS provider.

Primary use case: Containerized applications
Examples: Elastic Kubernetes Service, Google Kubernetes Engine, Azure Kubernetes Service, Azure Container Instance, Cloud Run, AWS Fargate, ECS etc.

C4 and C5 instances run on different hardware systems.
C4 runs on Haswell chips, while C5 runs on Skylake with the Nitro system. In some situations, such as governmental clouds, all resources have to be vetted and approved.

VPN:
https://www.computerhope.com/jargon/i/ip.htm#classes
Json Lint: it is used for the validator and reformatting for Json code(it's checking for Json code syntax, formatting) https://jsonlint.com
What is Lambda?
- Run functions on demand without the servers
- Supports many languages (python, java, c#, go, ruby, js)
- Adhoc tasks or completely serverless high TPS applications
- Pay per invocation, duration, memory
- Built in metrics with AWS CloudWatch

Integration with other AWS services:
Rest APIs, Data processing, Message buffering & Processing, Message processing, workflow (Step function) Orchestration, Change detection

EBS (Elastic Block store)

- Creating one EBS volume like Gp1
- Attach to the instance
- Goto terminal, check with command lsblk ------ showing all disk space and available disks
- Mounting the drive mkfs -t ext4 /dev/xvdf/
- Create one directory mkdir sanju
- mount /dev/xvdf /sanju/
- lsblk -----→ in that changed the disk name
- ls

- cd /sanju
- ls
- create one text file like nano sanju.txt --→ give some content
- cd ..
- umount /dev/xvdf
- lsblk
- deattach the volume from server in AWS console
- attach same EBS volume to another server
- enter terminal go to root
- create one dir mkdir sanju
- mount /dev/xvdf /sanju/
- cd sanju
- ls
- Ebs Snapshots are used for the backup the data
- Copying data from one available zone to another availability zone

AWS Identity and Access Management (IAM)

AWS Identity and Access Management (IAM) is a web service for securely controlling access to AWS services. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which AWS resources users and applications can access.

- Securely control individual and group access to your AWS resources
- IAM Policies are JSon documents used to describe permissions within AWS.
- IAM users, applications and services may assume IAM Roles
- Uses an IAM Policy for Permissions
- Manage the IAM Users and their Access
- Manage IAM roles and their permissions
- Manage federated users and their permissions

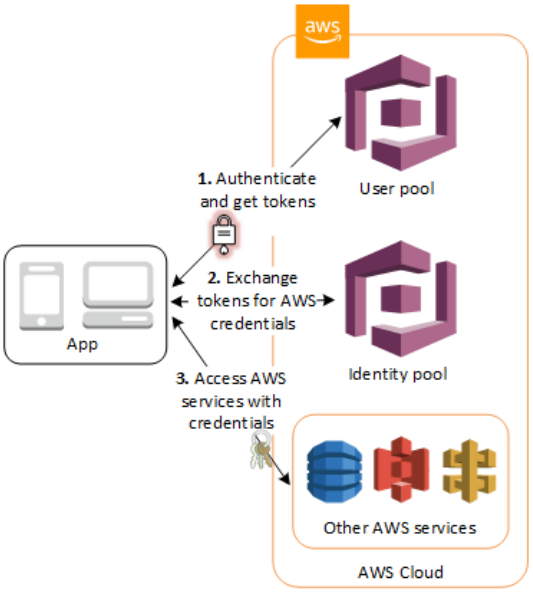
You can work with AWS Identity and Access Management (**IAM**) in any of the **following ways.**

- **AWS Management Console**
- **AWS Command Line Tools (CLI)**
- **AWS SDKs**

Amazon Cognito

- Amazon Cognito provides authentication, authorization, and user management for your web and mobile apps.
- Your users can sign in directly with a username and password, or through a third party such as Facebook, Amazon, Google, or Apple.

- Amazon Cognito provides token handling through the Amazon Cognito User Pools Identity SDKs for JavaScript, Android, and iOS. See Getting Started with User Pools and Using Tokens with User Pools.
- The two main components of Amazon Cognito are **user pools and identity pools.**
- **User pools** are user directories that provide sign-up and sign-in options for your app users.
    - **A user pool is a user directory in Amazon Cognito**. With a user pool, your users can sign into 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 identity providers.
    - **Whether your users sign in directly or through a third party**, all members of the user pool have a directory profile that you can access through a Software Development Kit (SDK).
- **Identity pools** enable you to grant your users access to other AWS services. You can use identity pools and user pools separately or together.
    - **Identity pools** provide AWS credentials to grant your users access to other AWS services. To enable users in your user pool to access AWS resources, you can configure an identity pool to exchange user pool tokens for AWS credentials.



AWS Managed Microsoft AD:

You can deploy an AWS Managed Microsoft AD (Enterprise Edition) directory across AWS Regions. Once configured, AWS automatically replicates your directory data in multiple Regions, so everything stays in sync.

Amazon S3:

- Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers.
- An Amazon S3 (**Simple Storage Service**), bucket is a public cloud storage resource available in Amazon Web Services' (AWS) an object storage offering. Amazon S3 buckets, which are like file folders, store objects, which consist of data and its descriptive metadata.
- An S3 customer first creates a bucket in the AWS region of his or her choice and gives it a globally unique name.
- AWS charges customers for storing objects in a bucket and for transferring objects in and out of buckets. Bucket pricing varies by region.vc
- S3 performance remains the same regardless of how many buckets an individual creates. Each AWS account can create 100 buckets, though more are available by requesting a service limit increase.

Lifecycle configuration:

To manage your objects so that they are stored cost effectively throughout their lifecycle, configure their lifecycle. A lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects. Lifecycle rules run once per day.

AWS CloudTrail:

Amazon S3 is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon S3. CloudTrail captures a subset of API calls for Amazon S3 as events, including calls from the Amazon S3 console and from code calls to the Amazon S3 APIs.

Awe S CloudTrail Log Monitoring – Share log files between accounts, monitor CloudTrail log files in real time by sending them to CloudWatch Logs, write log processing applications in Java, and validate that your log files have not changed after delivery by CloudTrail.

Logging with Amazon S3:

You can record the actions that are taken by users, roles, or AWS services on Amazon S3 resources and maintain log records for auditing and compliance purposes.

Amazon CloudWatch Alarms – Watch a single metric over a time that you specify and perform one or more actions based on the value of the metric relative to a given threshold over several time periods.

Backup vaults: Backup vaults are containers where your backups are stored. You can have one default vault or multiple vaults where backups can be stored.

Backup plans: Backup plans define your backup requirements, including backup schedules, backup retention rules and lifecycle rules.

Protected resources: Resources backed up by AWS Backup

Access points:

- Amazon S3 Access Points simplify managing data access at scale for shared datasets in S3.
- Access points are named network endpoints that are attached to buckets that you can use to perform S3 object operations

Amazon Elastic File System (Amazon EFS):

- Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources.
- It is built to scale on demand to petabytes without disrupting applications, growing, and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth.
- Amazon EFS is designed to provide massively parallel shared access to thousands of Amazons EC2 instances, enabling your applications to achieve high levels of aggregate throughput and IOPS with consistent low latencies.

Benefits and features:

Dynamic elasticity: Amazon EFS automatically and instantly scales your file system storage capacity up or down as you add or remove files without disrupting your applications, giving you the storage, you need – when you need it

Fully managed: Amazon EFS is a fully managed service providing shared file system storage for general purpose workloads.

Scalable performance: Amazon EFS is designed to provide the throughput, IOPS, and low latency needed for general purpose workloads.

Shared file storage: Amazon EFS provides secure access for thousands of connections. Amazon EC2 instances and on-premises servers can simultaneously access a shared Amazon EFS file system using a traditional file permissions model, file locking capabilities, and hierarchical directory structure via the NFSv4 protocol.

Jenkins:

Jenkins provides hundreds of plugins to support building, deploying, and automating any project.

VPC:

- **A bastion host** is a server whose purpose is to provide access to a private network from an external network, such as the Internet. Because of its exposure to potential attack, a bastion host must minimize the chances of penetration.
- **The bastion hosts** provide secure access to Linux instances located in the private and public subnets of your virtual private cloud (VPC).

Df -kh

Cat /etc/fstab
mount -a

Amazon DynamoDB

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, IoT, and many other applications.

Create Tables:

- Create DynamoDB tables with a few clicks. Just specify the desired read and write throughput for your table, and DynamoDB handles the rest.
- DynamoDB is a schema-less database that only requires a table name and primary key. The table's primary key is made up of one or two attributes that uniquely identify items, partition the data, and sort data within each partition.

Add and query items:

Once you have created a DynamoDB table, use the AWS SDKs to write, read, modify, and query items in DynamoDB.

Monitor and manage tables

Using the AWS Management Console, you can monitor performance and adjust the throughput of your tables, enabling you to scale seamlessly.

Amazon API Gateway

- create, maintain, and secure APIs at any scale
- Amazon API Gateway helps developers to create and manage APIs to back-end systems running on Amazon EC2, AWS Lambda, or any publicly addressable web service. With Amazon API Gateway, you can generate custom client SDKs for your APIs, to connect your back-end systems to mobile, web, and server applications or services.

Choose an API type:

HTTP API:

Build low-latency and cost-effective REST APIs with built-in features such as OIDC and OAuth2, and native CORS support.

Works with the following: Lambda, HTTP backends

WebSocket API:

Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards.

Works with the following: Lambda, HTTP, AWS Services

REST API:

Develop a REST API where you gain complete control over the request and response along with API management capabilities.

Works with the following: Lambda, HTTP, AWS Services

REST API(PVT):
Create a REST API that is only accessible from within a VPC.
Works with the following: Lambda, HTTP, AWS Services
Amazon Elastic Cache:

- Amazon Elastic Cache allows you to seamlessly set up, run, and scale popular open-source compatible in-memory data stores in the cloud.
- Build **data-intensive apps or boost the performance** of your existing databases by **retrieving data from high throughput and low latency in-memory data stores**.
- Amazon Elastic Cache is a popular choice for real-time use cases like Caching, Session Stores, Gaming, Geospatial Services, Real-Time Analytics, and Queuing.

A computer cluster:
A computer cluster is a set of loosely or tightly connected computers that work together so that, in many aspects, they can be viewed as a single system. Unlike grid computers, computer clusters have each node set to perform the same task, controlled, and scheduled by software

Document DB:
mongo --ssl host docdb-2020-02-08-14-15-11.cluster.region.docdb.amazonaws.com:27107 --sslCAFile rds-combined-ca-bundle.pem --username demoUser --password

1. Create AWS DocumentDB Cluster with 1 node on t2.micro
2. Deploy Amazon Linuz T2. Micro Instance
3. **Execute following commands**

```
echo -e "[mongodb-org-3.6] \nname=MongoDB
Repository\nbaseurl=https://repo.mongodb.org/yum/amazon/2013.03/mongodb-
org/3.6/x86_64/\ngpgcheck=1 \nenabled=1
\ngpgkey=https://www.mongodb.org/static/pgp/server-3.6.asc" | sudo tee
/etc/yum.repos.d/mongodb-org-3.6.repo
 sudo yum install -y mongodb-org-shell
```

4. Download the CA certificate for Amazon DocumentDB

```
  wget https://s3.amazonaws.com/rds-downloads/rds-combined-ca-bundle.pem
  cat rds-combined-ca-bundle.pem
```

5. Connect to cluster with following commands

```
  mongo --ssl --host docdb-2020-12-16-14-03-39.cluster-cegt01khj2di.us-east-
1.docdb.amazonaws.com:27017 --sslCAFile rds-combined-ca-bundle.pem --username awsuser -
-password <insertYourPassword>

  rs0:PRIMARY>  show dbs
  rs0:PRIMARY>  show collections
  rs0:PRIMARY>  db.helo.insertMany([
        {"_id": 1, "name" : "Matt", "status": "active", "level": 12, "score":202},
```

{"_id": 2, "name" : "Frank", "status": "inactive", "level": 2, "score":9},
{"_id": 3, "name" : "Karen", "status": "active", "level": 7, "score":87},
{"_id": 4, "name" : "Katie", "status": "active", "level": 3, "score":27, "status": "married", "emp": "yes", "kids": 3}
])

```
rs0:PRIMARY> db.helo.find({name: "Katie"})
rs0:PRIMARY> show DB OR Show collections
rs0:PRIMARY> db.helo.find({name: "Matt"})
rs0:PRIMARY> db.helo.find()  ----------> this command showing document
```

Verify with https://docs.mongodb.com/manual/core/databases-and-collections/

MongoDB:

MongoDB stores data records as documents (specifically BSON documents) which are gathered together in collections. A database stores one or more collections of documents

```
rs0:PRIMARY> show dbs
rs0:PRIMARY> use dns
rs0:PRIMARY> use test
rs0:PRIMARY> show tables
rs0:PRIMARY> show collections
```

AWS CloudFormation:

AWS CloudFormation is a service that helps you model and set up your Amazon Web Services resources so that you can spend less time managing those resources and more time focusing on your applications that run in AWS.

AWS CloudFormation provides a common language to describe and provision all the infrastructure resources in your environment in a safe, repeatable way.

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-anatomy.html

AWS CloudFormation Anatomy:

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-anatomy.html

How it works:

- Code your infrastructure using the CloudFormation template language in **the YAML or JSON format or start from many available sample templates**.
- Use AWS CloudFormation via the browser console, command line tools, or APIs to create a stack based on your template code.
- AWS CloudFormation provisions and configures the stacks and resources you specified in your template.

Stack: A stack is **a collection of AWS resources that you can manage as a single unit**. All the resources in a stack are defined by the stack's AWS CloudFormation template.

Stack Sets enables you to create, update, or delete stacks across multiple accounts and regions with a single operation

Change sets allow you to preview how proposed changes to a stack might impact your running resources, making changes to your stack only when you decide.
Run **Drift detection** to **identify configuration changes between your live resources and the template**. Drifts will be detected on stacks and resources.

- Only resources which currently support drift detection are displayed here. To view all your stack resources

Resource types: Over 500 resource types are supported by CloudFormation, covering over 100 AWS services.
Benefits:

- Simplify infrastructure management
- Quickly replicate your infrastructure
- Easily control and track changes to your infrastructure

In your AWS CloudFormation template, enter **Retain as the DeletionPolicy** for the resources that you want to keep when the stack is deleted. In the following example JSON and YAML template snippets, the Retain policy is specified for security groups.
https://aws.amazon.com/premiumsupport/knowledge-center/delete-cf-stack-retain-resources/
Stack sets:
Set up permissions for all users of the administrator account to perform stack set operations in all target accounts:
https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/stacksets-prereqs-self-managed.html
AWSCloudFormationStackSetAdministrationRole.
https://s3.amazonaws.com/cloudformation-stackset-sample-templates-us-east-/AWSCloudFormationStackSetAdministrationRole.yml

Terraform

What is the Terraform: It is IAC (Infrastructure as a code)
Two Types of approaches
Impetrative approach

- AWS CLI
- Bash Scripts

Declarative approach: Developing, changing infrastructure safely

- Terraform
- Cloudformation
- ARM
- Deployment manager

1. Developed by Hashicorp

2. Terraform Enterprise(Paid Version), Open source
3. Hashicorp having multiple tools in that main tools are(Terrafrom, Vault, consul)
4. Terraform – Infrastructure Orchestration tool, introduced in 2014 but after 2 years everyone using
5. Packer- Image Building tool
6. Vagrant – like VMware workstation or VirtualBox
7. Value- encrypted keys

https://registry.terraform.io/providers/hashicorp/aws/latest/docs/resources/kms_alias

Terraform commands:

Terraform init Prepare your working directory for other commands All plugins (e.g. for AWS) that are referenced in the found files are then downloaded and initialized.
Terraform validate Check whether the configuration is valid
Terraform plan Show changes required by the current configuration you can now see what changes would be made to the infrastructure
Terraform apply Create or update infrastructure
Terraform destroy Destroy previously created infrastructure If you want to shut down and dismantle all configured resources, this is possible
console Try Terraform expressions at an interactive command prompt
fmt Reformat your configuration in the standard style (formatting the code syntax)
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
untaint	Remove the 'tainted' state from a resource instance
version	Show the current Terraform version
workspace	Workspace management

ssh-keygen ---------showing public key

Different IAC (Infrastructure as Code) Tools:

- Puppet
- Chef
- Ansible
- Terraform

Github:

Ls command showing all list in your computer of User

Suppose do you want to change the drive user to d drive in **windows**

Step1: Cd \

Step2: Cd /d (path address like F:\AWS_Harsha_Online clasees)

In Bush

Step1: Cd\

Sep2: Cd "F:\AWS_Harsha_Online clasees"

Step3: Git init

Step4: Git remote add origin <gitlink>

Step5: Git remote -v

Step6: Git add

Step7: Git commit -m "first commit"

Step8: Git push origin master

Step9: Git help

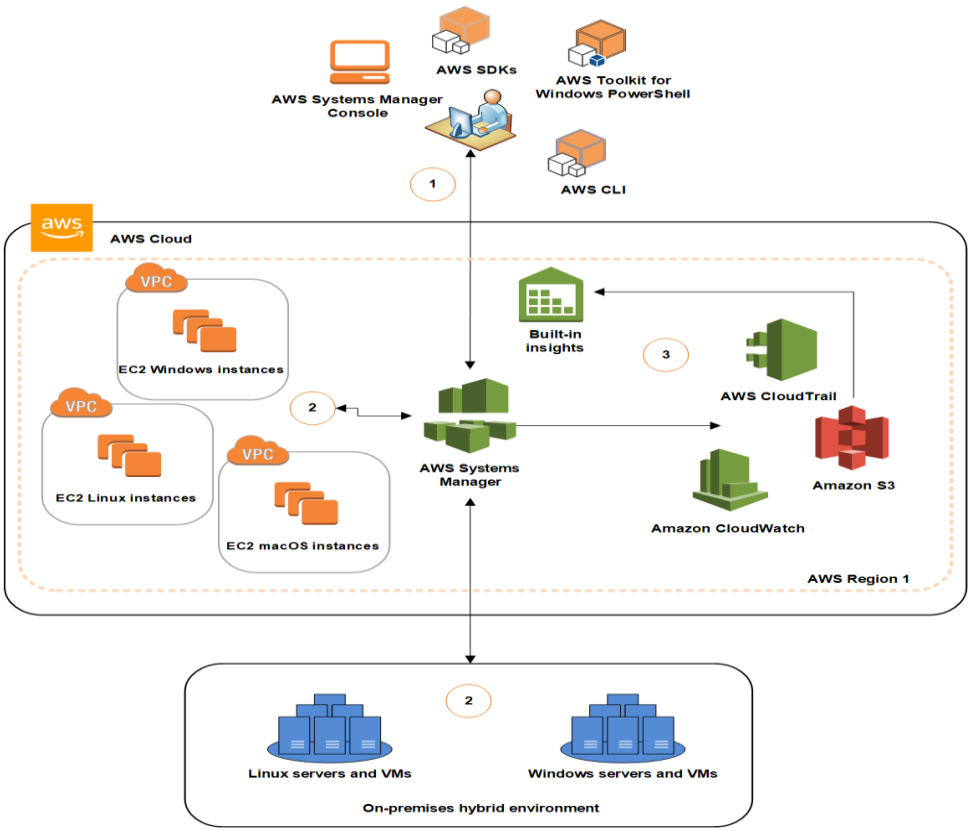
AWS Systems Manager:

AWS Systems Manager is an AWS service that you can use to view and control your infrastructure on AWS. Using the Systems Manager console, you can view operational data from multiple AWS services and automate operational tasks across your AWS resources. Systems Manager helps you maintain security and compliance by scanning your managed instances and reporting on (or taking corrective action on) any policy violations it detects.

Systems Manager capabilities are grouped into the following capability types:

Topics

- Quick Setup
- Operations Management
- Application Management
- Change Management
- Node Management
- Shared Resources



Session Manager:

- Session Manager provides secure and auditable instance management without the need to open inbound ports, maintain bastion hosts, or manage SSH keys.
- Session Manager also makes it easy to comply with corporate policies that require controlled access to instances, strict security practices.
- **Avoid the need to set up or maintain bastion hosts, to open up inbound SSH** or remote PowerShell ports, or to provision or manage SSH keys and certificates.

Run Command enables you to automate common administrative tasks and perform ad hoc configuration changes at scale.

AWS Systems Manager State Manager is a secure and scalable configuration management service that automates the process of keeping your Amazon EC2 and hybrid infrastructure in a state that you define.

- State Manager and Maintenance Windows can perform some similar types of updates on your managed instances.
- Which one you choose depends on whether you need to automate system compliance or perform high-priority, time-sensitive tasks only during periods you specify.

AWS Systems Manager Parameter Store provides secure, hierarchical storage for configuration data management and secrets management.

- A Parameter Store parameter is any piece of data that is saved in Parameter Store data such as passwords, database strings, Amazon Machine Image (AMI) IDs, and license codes as parameter values.
- Administrators who want to be notified when changes have or have not been made to secrets and passwords.

Parameter Store provides support for three types of parameters:

- String,
- String List
- Secure String.

AWS Parameter Store vs. AWS Secrets Manager:

	SSM Parameter Store	AWS Secrets Manager
Store values up to 4096 Characters	√	√
Values can be encrypted using KMS	√	√
Can be referenced in CloudFormation	√	√
Built-in password generator		√
Automated secret rotation		√
No additional costs	√	
Cross-account access		√

- There are no additional charges for using **SSM Parameter Store.**
- Charges are applicable for **Secrets Manager**

Hybrid Activations:

- To set up servers and virtual machines (VMs) in your hybrid environment as managed instances, you create a managed-instance hybrid activation.
- After you complete the activation, you receive an activation code and ID.
- This code/ID combination functions like an Amazon EC2 access ID and secret key to provide secure access to the Systems Manager service from your managed instances.

AWS OpsWorks:

AWS OpsWorks is a configuration management service that helps you build and operate highly dynamic applications and propagate changes instantly.

AWS Control Tower setup:

Set up your well-architected automated landing zone.

Cloud Trail:

- Continuously log your AWS account activity, Use CloudTrail to meet your governance, compliance, and auditing needs for your AWS accounts.

AWS Config: AWS Config provides an inventory of your AWS resources and a history of configuration changes to these resources. You can use AWS Config to define rules that evaluate these configurations for compliance.

- The changes performed -→configuration changes
- Compliance
- It is used for identify changes in configuration, and who did changes find out them

AWS Config:

- AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources.

Cloud Inspector: Amazon Inspector enables you to analyse the behaviour of your AWS resources and helps you identify potential security issues.

- Amazon Inspector tests the network accessibility of your Amazon EC2 instances and the security state of your applications that run on those instances.
- Amazon Inspector assesses applications for exposure, vulnerabilities, and deviations from best practices.
- Amazon Inspector also offers predefined software called an agent that you can optionally install in the operating system of the EC2 instances that you want to assess. T

AWS Service Catalog Tutorial: Use our Administrator Guide to get started with creating portfolios, managing permissions, and setting constraints.

- The AWS Service Catalog API provides programmatic control over all end-user actions as an alternative to using the AWS Management Console.

Cloud watch:

- Use CloudWatch to collect monitoring and operational data in the form of logs, metrics, and events.
- Amazon CloudWatch is a monitoring and observability services
- CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health.
- CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications, and services that run on AWS and on-premises servers.

Fleet Manager: Managed Instances moved to a new Systems Manager capability called Fleet Manager.

Features and benefits

- Centralized management
- Open-source agent
- Broad platform support

Json: Java script object notation

It is a syntax for starting and exchange data. It is a text written with Json

Less Verbose:

Json has more compact style than XML, and it is often more readable. The lightweight approach of JSON can make significant improvements.

Readable: The JSON structure is straightforward and readable. You have an easier time mapping to domain objects, no matter what programming language you're working with.

Structure data: Rather than XML Tree. In some situations, key/value pairs can limit what you can do, but you get a predictable and easy to understand data model. `

Using JSON-

- Json string
- JSON .parse the JSON String

```
{"name": "sanju", "age":23}

//data is in name/value pairs
//data is separated by commas
//curly Braces hold objects
//square brackets hold arrays


// Fundataental
let Person = {name:"sanju", age: 23 city: "hyd"};
console.log(person.name);
```

Web Application Firewall (WAF)

- **AWS WAF** is a web application firewall that helps protect your web applications or APIs against common web exploits that may affect availability, compromise security, or consume excessive resources.
- Set up protection for your Amazon CloudFront distributions, Application Load Balancers, and/or Amazon API Gateway stages in just under 5 minutes.

AWS Shield:

- Managed DDoS protection service.
- Security, Identity, and Compliance
- AWS Shield provides continuous attack detection and automatic mitigations. AWS Shield offers two tiers of protection - Standard and Advanced.
- To help mitigate DDoS attacks, authorize the DRT to access your account.

AWS Firewall Manager:

- Centralized security management, Centrally configure and manage firewall rules across accounts and applications.

- To get started designate an account in your organization as AWS Firewall Manager Administrator account.
- To manage AWS Network Firewall, the AWS Organizations management account must enable AWS Resource Access Manager (AWS RAM).
- The AWS Organizations management account must enable AWS RAM for all member accounts in your organization.

AWS Organizations:

- AWS Organizations is a service that enables you to centrally manage billing, control access, compliance, security and share resources across your AWS accounts.

AWS Certificate Manager:

AWS Certificate Manager (ACM) makes it easy to provision, manage, deploy, and renew SSL/TLS (**Transport Layer Security (TLS**)certificates on the AWS platform.

Resource Groups:

- Find and group your AWS resources by using queries.
- You can create unlimited, single-region groups in your account, use your groups to view group-related insights, and automate tasks on group resources.
- Groups can be based on resource types and tag queries, or AWS CloudFormation stacks.

AWS Resource Access Manager:

- Share AWS resources with other AWS accounts.

Amazon Cognito:

- Amazon Cognito offers **user pools and identity pools.**
- User pools are user directories that provide sign-up and sign-in options for your app users.
- Identity pools provide AWS credentials to grant your users access to other AWS services.

Amazon Macie

- Discover and protect your sensitive data at scale
- Amazon Macie is a data security and data privacy service that uses machine learning to help you identify and protect your sensitive data in AWS.

KMS (Key Management service):

- You can store your KMS customer master keys (CMKs) in a custom key store instead of the standard KMS key store.
- Custom key stores are created using an AWS Cloud HSM **(cloud-based hardware security module)** cluster that you own and manage.
- This provides direct control of the hardware security modules (HSMs) that generate the key material for your CMKs and perform cryptographic operations with them.

AWS Security Hub

- Manage and improve your security posture
- AWS Security Hub provides a consolidated view of your security status in AWS.
- Automate security checks, manage security findings, and identify the highest priority security issues across your AWS environment.

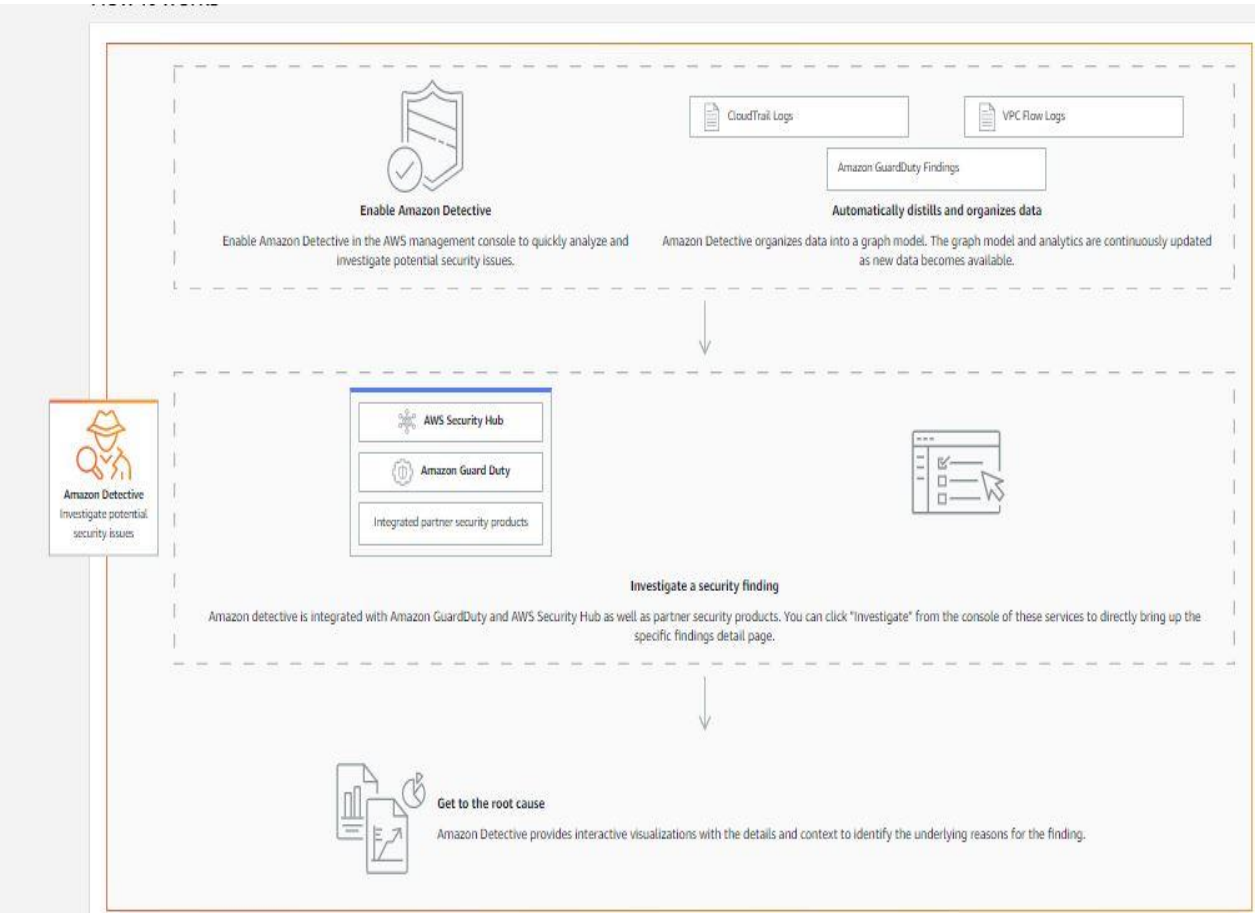
AWS Artifact (Downloading compliance reports):

- Compliance and security in the AWS Cloud
- No cost, self-service portal for on-demand access to AWS compliance reports and for entering select online agreements.
- Security, Identity, Compliance

Amazon Detective:

- Investigate potential security issues
- Amazon Detective makes it easy to investigate, analyze, and quickly identify the root cause of potential security issues or suspicious activities.
- Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to help you visualize and conduct faster and more efficient security investigations.

How it works:



AWS Audit Manager:

- **Continuously audit your AWS usage to simplify how you assess risk and compliance** with regulations and industry standards.
- Audit Manager makes it easier to evaluate if your policies, procedures, and activities, also known as controls, are operating as intended.

- The service offers prebuilt frameworks with controls that are mapped to well-known industry standards and regulations, full customization of frameworks and controls, and automated collection and organization of evidence as defined by each control requirement.
- When it is time for an audit, AWS Audit Manager helps you manage stakeholder reviews of your controls and enables you to build audit-ready reports with much less manual effort.

AWS Signer:

- Ensuring trust and integrity of your code
- AWS Signer is a fully managed code signing service to ensure the trust and integrity of your code for AWS Lambda.
- AWS Signer makes it easy for organizations to validate code against a digital signature to confirm the code is unaltered and from a trusted publisher.

CloudFront:

Amazon CloudFront is a fast-global content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

(OR)

- Amazon CloudFront is a content delivery network offered by Amazon Web Services. Content delivery networks provide a globally- distributed network of proxy servers which cache content, such as web videos or other bulky media, more locally to consumers, **thus improving access speed for downloading the content.**

- It is offering the most advanced security capabilities, including field level encryption and HTTPS support, seamlessly integrated with AWS Shield, AWS Web Application Firewall and Route 53 to protect against multiple types of attacks including network and application layer DDoS attacks.