# AWS Certified Cloud Practitioner (CLF-C01)

## Exam Guide

ExamCorgi.com

# AWS Guide

## Exam

### Summary

- 90 minutes
- 65 questions (approximately)
- Question Format:
    - Multiple-choice: Has one correct response and three incorrect responses (distractors)
    - Multiple-response: Has two/three correct responses out of five options
- Unanswered questions are scored as incorrect; there is no penalty for guessing. So always worthwhile to put some answer in.
- It is a pass or fail exam
- Each section of the examination has a specific weighting, so some sections have more questions than others
- Your results for the examination are reported as a scaled score from 100 through 1000, with a minimum passing score of 700.
- The examination uses a compensatory scoring model, which means that you do not need to "pass" the individual sections, only the overall examination
- Renewal period is every two years

*Domain 1: Cloud Concepts (26%)*

- 1.1 Define the AWS Cloud and its value proposition
  - Describe the basic global infrastructure of the AWS Cloud
  - Explain the six main benefits of the AWS Cloud
- 1.2 Identify aspects of AWS Cloud economics
  - Articulate the financial benefits of the AWS Cloud for your organization's cost management
- 1.3 List the different cloud architecture design principles
  - Differentiate between on-premises, hybrid-cloud, and all-in cloud models
  - Describe the AWS Well-Architected Framework

*Domain 2: Security and Compliance (25%)*

- 2.1 Define the AWS shared responsibility model
- 2.2 Define AWS Cloud security and compliance concepts
- 2.3 Identify AWS access management capabilities
- 2.4 Identify resources for security support

*Domain 3: Technology (33%)*

- 3.1 Define methods of deploying and operating in the AWS Cloud
- 3.2 Define the AWS global infrastructure
- 3.3 Identify the core AWS services, including compute, network, database, and storage services
  - Identify an appropriate solution using AWS Cloud services for various use cases
- 3.4 Identify resources for technology support

*Domain 4: Billing and Pricing (16%)*

- 4.1 Compare and contrast the various pricing models for AWS
  - Explain how to use pricing tools to make cost-effective choices for AWS services
- 4.2 Recognize the various account structures in relation to AWS billing and pricing
- 4.3 Identify resources available for billing support

## Tips

1. **ALWAYS** read the question in its entirely before attempting to answer the question
2. **ALWAYS** read all the answers in their entirety before attempting to answer the question
3. Note when multiple answers are required instead of just one

## Resources

- Practice Questions
    - 500 Questions for Exam Available (See Practice Questions.docx – These were mostly sourced from (https://www.examtopics.com/exams/amazon/aws-certified-cloud-practitioner/))
    - Another resource for past paper and practice questions (https://www.awslagi.com/), although this one is a paid site. I haven't purchased any exam there, but it does look promising.

- Practice Exams
    - These can be scheduled on this page (https://www.certmetrics.com/amazon/candidate/exam_scheduling.aspx) by clicking on the "Schedule Practice Exam with PSI" or "Schedule Practice Exam with Pearson VUE" buttons on the right. (These cost $20 plus tax unfortunately though)

- Webinars Held Routinely

- Concise Exam Notes (http://kayleigholiver.com/aws-cloud-practitioner-preparation-exam-notes/)

- AWS Whitepapers (https://aws.amazon.com/whitepapers)

# AWS

## General

### *What is Cloud Computing?*

The on-demand delivery of compute, databases storage, applications and other IT resources through a cloud services platform via the internet with pay-as-you-go pricing.

### *6 Advantages of Cloud Computing*

1. Trade capital expense for variable expense
2. Benefit from massive economies of scale
3. Stop guessing about capacity
4. Increased speed and agility
5. Stop spending money running and maintaining data centres
6. Go global in minutes

### *3 Types of Cloud Computing Deployments*

1. Public Cloud – AWS, Azure, GCP
2. Private Cloud (On Premise) – You manage it in your datacenter. Openstack or Vmware
3. Hybrid – Mix of public and private cloud

(http://kayleigholiver.com/aws-cloud-practitioner-cloud-computing-topics/)

### *Cloud Migration Approaches*

Cloud migration is the process of moving some or all your digital operations to your cloud. There are three main types of cloud migration you can perform:

- on-premises to cloud
- cloud to cloud
- cloud to on-premises

When performing any of these three migration types, there are five methods and strategies you can use. The strategies were first defined in the Gartner "5 Rs" model in 2011. These strategies are:

- Rehosting – moving applications to the cloud as-is. This is also sometimes referred to as 'Lift and Shift'
- Replatform—moving applications to the cloud without major changes, but taking advantage of benefits of the cloud environment, for example, you may choose to modify theyou're your application interacts with the database to benefit from automation and a more capable database infrastructure
- Refactor—modifying applications to better support the cloud environment
- Rebuild—rewrite the application from scratch
- Replace—retire the application and replace it with a new cloud-native application

(https://cloud.netapp.com/blog/cvo-blg-cloud-migration-approach-rehost-refactor-or-replatform)

### *3 Types of Cloud Computing*

1. Infrastructure As A Service (IaaS)
    - is a type of cloud computing offering in which a service provider provides the basic building blocks for cloud IT and give access to networking features, computers (virtual or on dedicated hardware), and data storage space.
    - Infrastructure as a Service provides you with the highest level of flexibility and management control over your IT resources and is most similar to existing IT resources that many IT departments and developers are familiar with today.
    - The service provider leaves the running of server instances to the customer, they do not access to what is on your servers
2. Platform as a Service (PaaS)
    - Platform as a Service – is a type of cloud computing offering in which a service provider removes the need for organizations to manage the underlying infrastructure (usually hardware and operating systems) and allow you to focus on the deployment and management of your applications.
    - This helps you be more efficient as you don't need to worry about resource procurement, capacity planning, software maintenance, patching, or any of the other undifferentiated heavy lifting involved in running your application, enabling them to develop, run, and manage business applications with less distraction.
    - Examples of PaaS include Amazon LightSail and AWS Elastic Beanstalk
3. Software As A Service (SaaS)
    - is a type of cloud computing offering in which a service provider provides you with a completed product that is run and managed by the service provider.
    - In most cases, people referring to Software as a Service are referring to end-user applications.
    - With a SaaS offering you do not have to think about how the service is maintained or how the underlying infrastructure is managed; you only need to think about how you will use that particular piece of software.
    - A common example of a SaaS application is web-based email where you can send and receive email without having to manage feature additions to the email product or maintaining the servers and operating systems that the email program is running on.

### *Regions, Availability Zones & Edge Locations*

- Region is a geographical area that has two or more Availability Zones. Each Region is completely independent.
- Availability Zone (AZ) is an area with either one or more discrete Data Centres (building filled with servers), each with redundant power, networking, and connectivity, housed in separate facilities. If there are more than one data centre, they are counted as one AZ because they are located close together. Each Availability Zone is isolated, but the Availability Zones in a Region are connected through low-latency links.
- Edge Locations are endpoints used for caching content. They are located in most of the major cities around the world and are specifically used by CloudFront to distribute AWS content closer to end-users to reduce latency.

### *"Why bother with multi-region architectures?*

There are three reasons why you would want to have a multi-region architecture:

- Improve latency for end-users - The closer your backend origin is to end-users, the better the experience. Content Delivery Networks (CDN) like Amazon CloudFront have successfully been used to speed up the delivery of content, especially static content (e.g., images, videos, JavaScript libraries, etc.) to end-users across the globe. Using a globally-distributed network of caching servers, static content is served as if it was local to consumers, thus improving the delivery of that static content. However, even if CloudFront solves the problem for much of your content, some more dynamic calls still need to be done on the backend, and it could be far away, adding precious milliseconds to the request. By using a multi-region architecture you reduce the physical distance between your most distant users and the resources they are trying to access, thus improving latency.
- Large scale disaster recovery using AWS regions - Most organizations try to implement High Availability (HA) instead of Disaster Recovery (DR) to guard them against any downtime of services. In case of HA, we ensure there exists a fallback mechanism for our services. The service that runs in HA is handled by hosts running in different availability zones but in the same geographical region. This approach, however, does not guarantee that our business will be up and running in case the entire region goes down. DR takes things to a completely new level, wherein you need to be able to recover from a different region that's separated by over 250 miles. Our DR implementation is an Active/Passive model, meaning that we always have minimum critical services running in different regions, but a major part of the infrastructure is launched and restored when required.
- Business requirements – Perhaps for regional compliance that require data and services to be regionally hosted, or perhaps for an entirely different business reason, companies will opt to rollout in multiple regions.

### *CapEx & OpEx*

- CapEx (capital expenditure) is defined as business expenses incurred in order to create long-term benefits in the future, such as purchasing fixed assets like a building or equipment. Some examples of IT items that fall under this category would be whole systems and servers, printers and scanners, or air conditioners and generators. You buy these items once and they benefit your business for many, many years. Maintenance of such items is also considered CapEx, as it extends their lifetime and usefulness. Capex can also be defined as Total Cost of Ownership (TCO).
- OpEx (operating expenditure), the expenses to run day-to-day business, like services and consumable items that get used up and are paid for according to use. This includes printer cartridges and paper, electricity, and even yearly services like website hosting or domain registrations. These things are necessary for your business's success but are not considered major long-term investments like CapEx items.
- The cloud allows you to trade high initial CapEx (such as data centers and physical servers) for a variable OpEx model, and only pay for IT as you consume it. Plus, the variable OpEx expenses are much lower than what you would pay to do it yourself because of the massive economies of scale that AWS has created.

### *Scalability*

Scalability is the ability of a software system to increase workload size without application service interruption or performance impact.

### *Elasticity*

In cloud computing, elasticity is defined as "the degree to which a system is able to adapt to workload changes by provisioning and de-provisioning resources in an autonomic manner, such that at each point in time the available resources match the current demand as closely as possible

Some cloud solutions can also be automatically adjusted to meet these needs. This means you can set them up to scale up or down automatically based on certain conditions, like when your cloud solution is has too many resources of which some are being under-utilised or if you have too few resources and your solution is running out of processing power.

### *Horizontal vs Vertical Scaling*

Horizontal scaling means scaling by adding more machines to your pool of resources (also described as "scaling out"), whereas vertical scaling refers to scaling by adding more power (e.g. CPU, RAM) to an existing machine (also described as "scaling up").

### *Scaling vs Elasticity*

Scalability is a characteristic of a **software architecture** related to serving higher amount if workload, where elasticity is a characteristic of the **physical layer** below, entirely related to hardware budget optimizations.

### *Fault Tolerance*

Fault tolerance refers to the ability of a system (computer, network, cloud cluster, etc.) to continue operating without interruption when one or more of its components fail.

The objective of creating a fault-tolerant system is to prevent disruptions arising from a single point of failure, ensuring business continuity of mission-critical applications or systems.

A fault tolerant system must be able to handle such failures and seamlessly recover without any long term negative impacts to business operations.

### *Availability*

Availability refers to the percentage of time that the infrastructure, system or a solution remains operational under normal circumstances in order to serve its intended purpose. For cloud infrastructure solutions, availability relates to the time that the datacenter is accessible or delivers the intended IT service as a proportion of the duration for which the service is purchased. The mathematical formula for Availability is as follows:

Percentage of availability = (total elapsed time – sum of downtime)/total elapsed time

An SLA (Service-level agreement) of 99.999 percent availability (the famous five nines), the yearly service downtime could be as much as 5.256 minutes.

The numbers portray a precise image of the system availability, allowing organizations to understand exactly how much service uptime they should expect from IT service providers.

True high availability means that a resource is available from at least three different availability zones, however AWS currently only guarantees that a resource can be reached at two different availability zones.

### *Fault Tolerance vs Availability*

A fault tolerant system is a critical system that requires being operational with zero downtime, while a highly available system can tolerate short interruptions in service. Both high availability and fault tolerance require the ability to detect failure.

A fault tolerant system typically has significantly higher costs than a highly available system. The higher costs are due to having physical redundancy added to the system. Redundancy is achieved by having mirrored components up and running, so failure of a component will not mean failure of the entire system.

### *Reliability*

Reliability refers to the probability that the system will meet certain performance standards in yielding correct output for a desired time duration.

Reliability can be used to understand how well the service will be available in context of different real-world conditions. For instance, a cloud solution may be available with an SLA commitment of 99.999 percent, but vulnerabilities to sophisticated cyber-attacks may cause IT outages beyond the control of the vendor.

A common metric is to calculate the Mean Time Between Failures (MTBF). MTBF represents the time duration between a component failure of the system. Similarly, organizations may also evaluate the Mean Time To Repair (MTTR), a metric that represents the time duration to repair a failed system component such that the overall system is available as per the agreed SLA commitment

### *Availability vs Reliability*

The measurement of Availability is driven by **time loss** whereas the measurement of Reliability is driven by the **frequency and impact** of failures. Mathematically, the Availability of a system can be treated as a function of its Reliability. In other words, Reliability can be considered a subset of Availability.

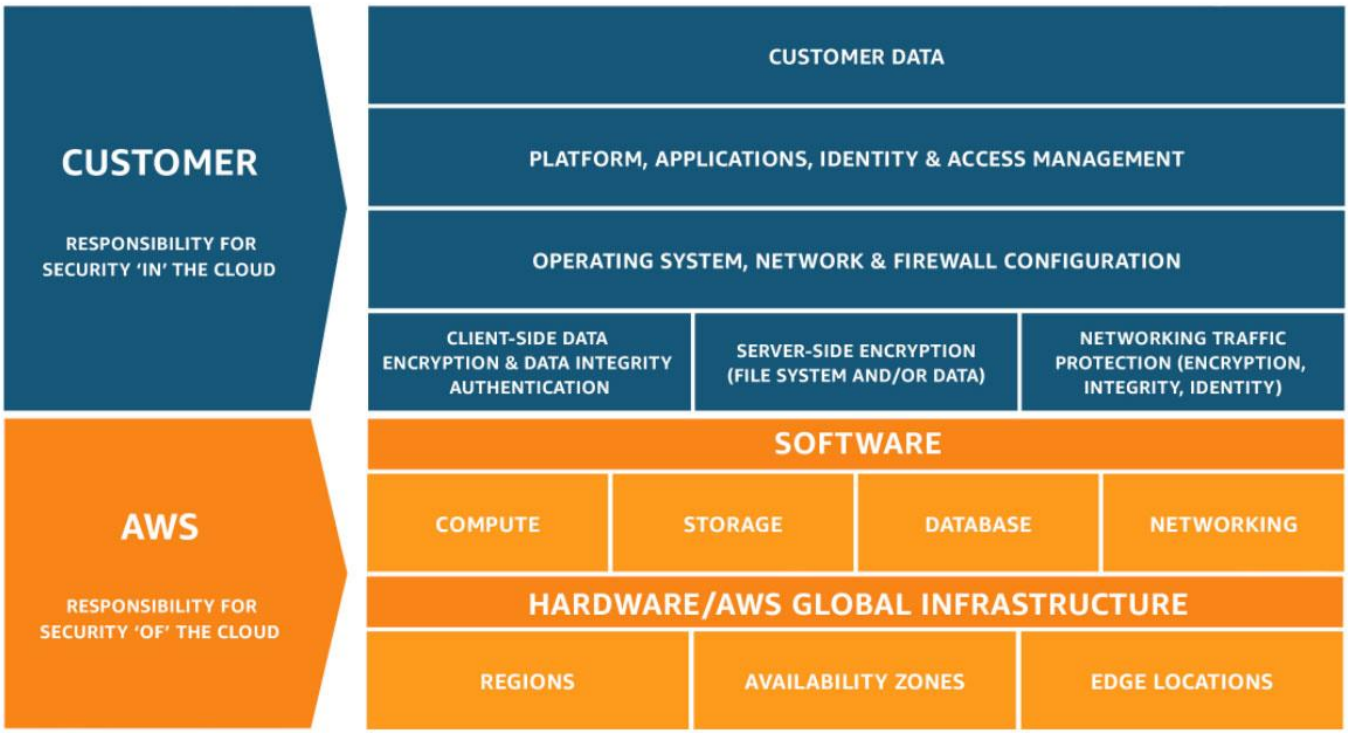
### *Agility*

Agile is a time boxed, iterative approach to software delivery that builds software incrementally from the start of the project, instead of trying to deliver it all at once near the end.

The requirements might need to change. We are not talking about growth here but a change of way of doing things. May be they started with a static webpage and it turned out they now need a database instead. This is not elasticity. They don't need more computing power, they need an agile solution that can change overtime.

Agility is the practice of “building in” the ability to change quickly and inexpensively. The cloud not only makes these other practices practical but provides agility on its own. Infrastructure can be provisioned in minutes instead of months, and de-provisioned or changed just as quickly.

*Shared Responsibility Model*




Security and Compliance is a shared responsibility between AWS and the customer. This shared model can help relieve the customer's operational burden as AWS operates, manages and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the service operates. The nature of this shared responsibility also provides the flexibility and customer control that permits the deployment. As shown in the chart above, this differentiation of responsibility is commonly referred to as Security "of" the Cloud versus Security "in" the Cloud.

Also, note that the customer:

- assumes responsibility and management of the guest operating system (including updates and security patches), other associated application software as well as the configuration of the AWS provided security group firewall.
- should carefully consider the services they choose as their responsibilities vary depending on the services used, the integration of those services into their IT environment, and applicable laws and regulations.
- is responsible for data configuration (i.e. encrypting data at rest and in transit)

### *Inherited, Shared and Customer Specific Controls*

- Inherited Controls
    - Controls which a customer fully inherits from AWS and so does not have to worry about. Examples include:
        - Physical controls
        - Environmental controls
- Shared Controls
    - Controls which apply to both the infrastructure layer and customer layers, but in completely separate contexts or perspectives. In a shared control, AWS provides the requirements for the infrastructure and the customer must provide their own control implementation within their use of AWS services. Examples include:
        - Patch Management – AWS is responsible for patching and fixing flaws within the infrastructure, but customers are responsible for patching their guest OS and applications.
        - Configuration Management – AWS maintains the configuration of its infrastructure devices, but a customer is responsible for configuring their own guest operating systems, databases, and applications.
        - Awareness & Training - AWS trains AWS employees, but a customer must train their own employees.
- Customer Specific
    - Controls which are solely the responsibility of the customer based on the application they are deploying within AWS services. Examples include:
        - Customer data
        - Service and Communications Protection or Zone Security which may require a customer to route or zone data within specific security environments.

### *What is Throughput?*

Throughput is a measure of how many units of information a system can process in a given amount of time.

High-throughput computing (HTC) involves running many independent tasks that require a large amount of computing power. With HTC, users can run many copies of their software simultaneously across many different computers. What could have taken weeks before on one computer now takes mere hours on a HTC cluster.

### *Loose Coupling*

- As application complexity increases, a desirable attribute of an IT system is that it can be broken into smaller, loosely coupled components. This means that IT systems should be designed in a way that reduces interdependencies—a change or a failure in one component should not cascade to other components.
- Your infrastructure also needs to have well defined interfaces that allow the various components to interact with each other only through specific, technology- agnostic interfaces. Modifying any underlying operations without affecting other components should be made possible.
- Subareas of loose coupling include the coupling of classes, interfaces, data, and services.
- Loose coupling is when each of the components of a system has, or makes use of, little or no knowledge of the definitions of other separate components.
- Loose coupling between services can also be done through asynchronous integration, which involves one component that generates events and another that consumes them. The two components do not integrate through direct point-to-point interaction, but usually through an intermediate durable storage layer. This approach decouples the two components and introduces additional resiliency. So, for example, if a process that is reading messages from the queue fails, messages can still be added to the queue to be processed when the system recovers.

*Data Recovery Methods*

☞ Backup and Restore: a simple, straightforward, cost-effective method that backs up and restores data as needed. Keep in mind that because none of your data is on standby, this method, while cheap, can be quite time-consuming.

☞ Pilot Light: The idea of the pilot light is an analogy that comes from gas heating. In that scenario, a small flame that's always on can quickly ignite the entire furnace to heat up a house. In this DR approach, you simply replicate part of your IT structure for a limited set of core services so that the pilot light environment seamlessly takes over in the event of a disaster. A small part of your infrastructure is always running simultaneously syncing mutable data (as databases or documents), while other parts of your infrastructure are switched off and used only during testing. Unlike a backup and recovery approach, you must ensure that your most critical core elements are already configured and running in the pilot light environment. When the time comes for recovery, you can rapidly provision a full-scale production environment around the critical core.

☞ Warm Standby: The term warm standby is used to describe a DR scenario in which a scaled-down version of a fully functional environment is always running in the cloud. A warm standby solution extends the pilot light elements and preparation. It further decreases the recovery time because some services are always running. By identifying your business-critical systems, you can fully duplicate these systems on in the warm standby environment and have them always on.

☞ Hot Standby: Also known as a Multi-Site Solution, this method fully replicates your company's data/applications between two or more active locations and splits your traffic/usage between them. If a disaster strikes, everything is simply rerouted to the unaffected area, which means you'll suffer almost zero downtime. However, by running two separate environments simultaneously, you will obviously incur much higher costs.

## *Well Architected Framework*

The AWS Well-Architected Framework helps you understand the pros and cons of decisions you make while building systems on AWS. By using the Framework you will learn architectural best practices for designing and operating reliable, secure, efficient, and cost-effective systems in the cloud. It provides a way for you to consistently measure your architectures against best practices and identify areas for improvement.

The process for reviewing an architecture is a constructive conversation about architectural decisions, and is not an audit mechanism. We believe that having well-architected systems greatly increases the likelihood of business success.

## *What is an AWS Solutions Architect (SA)?*

AWS Solutions Architects are individuals with years of experience architecting solutions across a wide variety of business verticals and use cases. Collectively AWS SAs have helped design and review thousands of customers' architectures on AWS. From this experience, they have identified best practices and core strategies for architecting systems in the cloud

They are responsible for building and integration of computer systems and information for meeting specific needs. Typically, this involves the integration of hardware and software for meeting the customer-defined purpose. Examination of current systems and architecture is also one of their responsibilities. They work with technical and business staff for recommending solutions for more effective systems.

The main duties of an AWS Solutions Architect are:

- Use technology to find a solution to business problems
- Decide which platform, framework or tech-stack should be used for the creation of a solution
- Designing the appearance of the application, what modules to use and the interaction between those modules
- Plan for scaling for future and it's the maintenance of the system
- Determine the risk associated with third-party platforms or frameworks

## *What is the AWS Well-Architected Tool?*

The AWS Well-Architected Tool (AWS WA Tool) is a service in the cloud that provides a consistent process for you to review and measure your architecture using the AWS Well-Architected Framework. The AWS WA Tool provides recommendations for making your workloads more reliable, secure, efficient, and cost-effective.

### *Components, Workloads and Architecture*

- A **component** is the code, configuration and AWS Resources that together deliver against a requirement. A component is often the unit of technical ownership, and is decoupled from other components.
- The term **workload** is used to identify a set of components that together deliver business value. A workload is usually the level of detail that business and technology leaders communicate about.
- We think about **architecture** as being how components work together in a workload. How components communicate and interact is often the focus of architecture diagrams.

### *General Cloud Design Principles*

1. Stop guessing your capacity needs
    - If you make a poor capacity decision when deploying a workload, you might end up sitting on expensive idle resources or dealing with the performance implications of limited capacity. With cloud computing, these problems can go away. You can use as much or as little capacity as you need, and scale up and down automatically.
2. Test systems at production scale:
    - In the cloud, you can create a production-scale test environment on demand, complete your testing, and then decommission the resources. Because you only pay for the test environment when it's running, you can simulate your live environment for a fraction of the cost of testing on premises.
3. Automate to make architectural experimentation easier:
    - Automation allows you to create and replicate your workloads at low cost and avoid the expense of manual effort. You can track changes to your automation, audit the impact, and revert to previous parameters when necessary.
4. Allow for evolutionary architectures:
    - Allow for evolutionary architectures. In a traditional environment, architectural decisions are often implemented as static, onetime events, with a few major versions of a system during its lifetime. As a business and its context continue to evolve, these initial decisions might hinder the system's ability to deliver changing business requirements. In the cloud, the capability to automate and test on demand lowers the risk of impact from design changes. This allows systems to evolve over time so that businesses can take advantage of innovations as a standard practice.
5. Drive architectures using data:
    - In the cloud, you can collect data on how your architectural choices affect the behaviour of your workload. This lets you make fact based decisions on how to improve your workload. Your cloud infrastructure is code, so you can use that data to inform your architecture choices and improvements over time.
6. Improve through game days:
    - A game day simulates a failure or event to test systems, processes, and team responses. The purpose is to actually perform the actions the team would perform as if an exceptional event happened. These should be conducted regularly so that your team builds "muscle memory" on how to respond. Your game days should cover the areas of operations, security, reliability, performance, and cost.
    - In AWS, your game days can be carried out with replicas of your production environment using AWS CloudFormation. This enables you to test in a safe environment that resembles your production environment closely.
    - Test how your architecture and processes perform by regularly scheduling game days to simulate events in production. This will help you understand where improvements can be made and can help develop organizational experience in dealing with events.

(https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf)

*The Five Pillars of a well Architecture Framework*

- **C**ost optimization
  - The ability to run systems to deliver business value at the lowest price point.
- **R**eliability
  - The ability of a workload to perform its intended function correctly and consistently when it's expected to. This includes the ability to operate and test the workload through its total lifecycle.
- **O**perational Excellence
  - The ability to support development and run workloads effectively, gain insight into their operations, and to continuously improve supporting processes and procedures to deliver business value
- **P**erformance efficiency
  - The ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve.
- **S**ecurity
  - The security pillar encompasses the ability to protect data, systems, and assets to take advantage of cloud technologies to improve your security.

Easily remembered as **CROPS**

*Figure 1: Harvesting some best practices*

When architecting technology solutions, if you neglect the five pillars it can become challenging to build a system that delivers on your expectations and requirements. Incorporating these pillars into your architecture will help you produce stable and efficient systems. This will allow you to focus on the other aspects of design, such as functional requirements.

The AWS Well-Architected Framework documents a set of foundational questions that allow you to understand if a specific architecture aligns well with cloud best practices. The framework provides a consistent approach to evaluating systems against the qualities you expect from modern cloud-based systems, and the remediation that would be required to achieve those qualities. As AWS continues to evolve, and we continue to learn more from working with our customers, we will continue to refine the definition of well-architected.

(https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf)

*Cost Optimization Design Principles & Best Practices*

- **Cost-effective resources:** Using the appropriate instances and resources for your workload is key to cost savings. AWS offers a variety of flexible and cost-effective pricing options to acquire services in a way that best fits your needs.
    - Pay only for the computing resources that you require and increase or decrease usage depending on business requirements, not by using elaborate forecasting. For example, development and test environments are typically only used for eight hours a day during the work week. You can also modify the demand, using a throttle, buffer, or queue to smooth the demand and serve it with less resources resulting in a lower cost, or process it at a later time with a batch service. When designing to modify demand and supply resources, actively think about the patterns of usage, the time it takes to provision new resources, and the predictability of the demand pattern.
    - AWS does the heavy lifting of data center operations like racking, stacking, and powering servers. It also removes the operational burden of managing operating systems and applications with managed services. This allows you to focus on your customers and business projects rather than on IT infrastructure.
- **Measure overall efficiency and expenditure:** The ability to align your organization to an agreed set of financial objectives, and provide your organization the mechanisms to meet them. Measure the business output of the workload and the costs associated with delivering it. Use this measure to know the gains you make from increasing output and reducing costs
- **Optimize over time:** As your requirements change, be aggressive in decommissioning resources, entire services, and systems that you no longer require. As AWS releases new services and features, it's a best practice
- to review your existing architectural decisions to ensure they continue to be the most cost effective. Your organization needs to dedicate time and resources to build capability in this new domain
    - The cloud makes it easier to accurately identify the usage and cost of systems, which then allows transparent attribution of IT costs to individual workload owners and drives efficient usage behaviour. This helps measure return on investment (ROI) and gives workload owners an opportunity to optimize their resources and reduce costs

*Reliability Design Principles and Best Practices*

- **Workload Architecture:** A reliable workload starts with upfront design decisions for both software and infrastructure. Understand that distributed systems rely on communications networks to interconnect components, such as servers or services. Your workload must operate reliably despite data loss or latency in these networks. Components of the distributed system must operate in a way that does not negatively impact other components or the workload, i.e. loose coupling.
    - **Foundations:** Foundational requirements are those whose scope extends beyond a single workload or project. Before architecting any system, foundational requirements that influence reliability should be in place. For example, you must have sufficient network bandwidth, adequate storage space and enough compute capacity.
    - **Scale horizontally:** to increase aggregate workload availability**.** Replace one large resource with multiple small resources to reduce the impact of a single failure on the overall workload. Distribute requests across multiple, smaller resources to ensure that they don't share a common point of failure.
    - **Stop guessing capacity:** A common cause of failure in on-premises workloads is resource saturation, when the demands placed on a workload exceed the capacity of that workload (this is often the objective of denial of service attacks). In the cloud, you can monitor demand and workload utilization, and automate the addition or removal of resources to maintain the optimal level to satisfy demand without over or under-provisioning.
- **Change Management:** Changes to your workload or its environment must be anticipated and accommodated to achieve reliable operation of the workload. Changes include those imposed on your workload, such as spikes in demand, as well as those from within, such as feature deployments and security patches. Using AWS, you can monitor the behaviour of a workload and automate the response to these changes. With monitoring in place, your team will be automatically alerted when KPIs deviate from expected norms. Automatic logging of changes to your environment allows you to audit and identify actions that might have impacted reliability.
    - Changes to your infrastructure should be made using automation. The changes that need to be managed include changes to the automation, which then can be tracked and reviewed.
- **Failure Management / automatically recover from failure:** In any system of reasonable complexity, it is expected that failures will occur. Reliability requires that your workload be aware of failures as they occur and take action to avoid impact on availability. Workloads must be able to both withstand failures and automatically repair issues:
    - With AWS, you can take advantage of automation to react to monitoring data. For example, when a particular metric crosses a threshold, you can trigger an automated action to remedy the problem. Also, rather than trying to diagnose and fix a failed resource that is part of your production environment, you can replace it with a new one and carry out the analysis on the failed resource out of band.
    - Since the cloud enables you to stand up temporary versions of a whole system at low cost, you can use automation to simulate different failures or to recreate scenarios that led to failures before (chaos engineering) and observe the full recovery processes
    - Regularly back up your data and test your backup files to ensure that you can recover from both logical and physical errors.
    - Tracking KPIs will help you identify and mitigate single points of failure.
    - These approaches expose failure pathways that you can test and fix before a real failure scenario occurs, thus reducing risk.

*Operational Excellence Design Principles and Best Practices*

- **Organization**: Your teams need to have a shared understanding of your entire workload, their role in it, and shared business goals to set the priorities that will enable business success. Well-defined priorities will maximize the benefits of your efforts.
    - Evaluate internal and external customer needs involving key stakeholders, including business, development, and operations teams, to determine where to focus efforts.
    - Ensure that you are aware of guidelines or obligations defined by your organizational governance and external factors, such as regulatory compliance requirements and industry standards that may mandate or emphasize specific focus. Validate that you have mechanisms to identify changes to internal governance and external compliance requirements.
    - Evaluate threats to the business (for example, business risk and liabilities, and information security threats) and maintain this information in a risk registry.

- **Understand your workloads and their expected behaviours**. You will then be able design them to provide insight to their status and build the procedures to support them. Design your workload so that it provides the information necessary for you to understand its internal state (for example, metrics, logs, events, and traces) across all components in support of observability and investigating issues. Iterate to develop the telemetry necessary to monitor the health of your workload, identify when outcomes are at risk, and enable effective responses.

- **Define expected outcomes**: Successful operation of a workload is measured by the achievement of business and customer outcomes. Determine how success will be measured, and identify metrics that will be used in those calculations to determine if your workload and operations are successful.

- **Evolve & experiment**: You must learn, share, and continuously improve to sustain operational excellence. Dedicate work cycles to making continuous incremental improvements. Perform post incident analysis of all customer impacting events. Identify the contributing factors and preventative action to limit or prevent recurrence. Try to accelerate employees learning and keeps team members interested and engaged. Teams must grow their skill sets to adopt new technologies, and to support changes in demand and responsibilities. Set up regular game days to test, review and validate that all procedures are effective and that teams are familiar with them.

- **Make frequent, small, reversible changes:** Design workloads to allow components to be updated regularly. Make changes in small increments that can be reversed if they fail (without affecting customers when possible).

- **Perform operations as code**: In the cloud, you can apply the same engineering discipline that you use for application code to your entire environment. You can define your entire workload (applications, infrastructure) as code and update it with code. You can implement your operations procedures as code and automate their execution by triggering them in response to events. By performing operations as code, you limit human error and enable consistent responses to events.

- **Anticipate failure and learn from it**: Perform “pre-mortem” exercises to identify potential sources of failure so that they can be removed or mitigated. Test your failure scenarios and validate your understanding of their impact. Test your response procedures to ensure that they are effective, and that teams are familiar with their execution. Set up regular game days to test workloads and team responses to simulated events. Drive improvement through lessons learned from all operational events and failures. Share what is learned across teams and through the entire organization.

*Performance Efficiency Design Principles and Best Practices*

- **Democratize advanced technologies and use serverless architectures:** Make advanced technology implementation easier for your team by delegating complex tasks to your cloud vendor via serverless architectures. Rather than asking your IT team to learn about hosting and running a new technology, consider consuming the technology as a service. For example, NoSQL databases, media transcoding, and machine learning are all technologies that require specialized expertise. In the cloud, these technologies become services that your team can consume, allowing your team to focus on product development rather than resource provisioning and management.

- **Go global in minutes:** Deploying your workload in multiple AWS Regions around the world allows you to provide lower latency and a better experience for your customers at minimal cost.

- **Selection:** The optimal solution for a particular workload varies, and solutions often combine multiple approaches. Well-architected workloads use multiple solutions and enable different features to improve performance. AWS resources are available in many types and configurations, which makes it easier to find an approach that closely matches your workload needs. Also understand how cloud services are consumed and always use the technology approach that aligns best with your workload goals. For example, consider data access patterns when you select database or storage approaches.

- **Trade-offs:** When you architect solutions, think about trade-offs to ensure an optimal approach. Depending on your situation, you could trade consistency, durability, and space for time or latency, to deliver higher performance. As you make changes to the workload, collect and evaluate metrics to determine the impact of those changes. Measure the impacts to the system and to the end-user to understand how your trade-offs impact your workload. Use a systematic approach, such as load testing, to explore whether the trade-off improves performance.

- **Monitoring:** After you implement your workload, you must monitor its performance so that you can remediate any issues before they impact your customers. Monitoring metrics should be used to raise alarms when thresholds are breached. Ensuring that you do not see false positives is key to an effective monitoring solution. Automated triggers avoid human error and can reduce the time it takes to fix problems. Plan for game days, where simulations are conducted in the production environment, to test your alarm solution and ensure that it correctly recognizes issues

- **Experiment more often**: With virtual and automatable resources, you can quickly carry out comparative testing using different types of instances, storage, or configurations.

- **Review:** Cloud technologies are rapidly evolving and you must ensure that workload components are using the latest technologies and approaches to continually improve performance. You must continually evaluate and consider changes to your workload components to ensure you are meeting its performance and cost objectives. New technologies, such as machine learning and artificial intelligence (AI), can allow you to reimagine customer experiences and innovate across all of your business workloads

*Security Design Principles and Best Practices*

- **Implement a strong identity foundation:** Implement the principle of least privilege and enforce separation of duties with appropriate authorization for each interaction with your AWS resources. Centralize identity management, and aim to eliminate reliance on long-term static credentials.

- **Keep people away from data**: Use mechanisms and tools to reduce or eliminate the need for direct access or manual processing of data. This reduces the risk of mishandling or modification and human error when handling sensitive data.

- **Data Protection:** Data classification provides a way to categorize organizational data based on levels of sensitivity, and encryption protects data by way of rendering it unintelligible to unauthorized access. AWS provides multiple means for encrypting data at rest and in transit. We build features into our services that make it easier to encrypt your data. Additionally, AWS has designed storage systems for exceptional resiliency. For example, Amazon S3 is designed to provide 99.999999999% durability of objects over a given year.

- **Enable traceability:** Monitor, alert, and audit actions and changes to your environment in real time. Integrate log and metric collection with systems to automatically investigate and take action.

- **Apply security at all layers**: Infrastructure protection encompasses control methodologies (such as defence in depth with multiple security controls), this could include controls, such as, enforcing boundary protection; monitoring points of ingress and egress; implementing stateful and stateless packet inspection; comprehensive logging monitoring, and alerting.

- **Detection:** You can use detective controls to identify a potential security threat or incident. There are different types of detective controls. For example, processing logs, monitoring events and conducting an inventory of assets and their detailed attributes promotes more effective decision making to help establish operational baselines. You can also use internal auditing, an examination of controls related to information systems, to ensure that practices meet policies and requirements and that you have set the correct automated alerting notifications based on defined conditions. These controls are important reactive factors that can help your organization identify and understand the scope of anomalous activity.

- **Automate security best practices:** Automated software-based security mechanisms improve your ability to securely scale more rapidly and cost-effectively. Create secure architectures, including the implementation of controls that are defined and managed as code in version-controlled templates.

- **Incident Preparation & Response**: Prepare for an incident by having incident management and investigation policy and processes that align to your organizational requirements. Even with extremely mature preventive and detective controls, your organization should still put processes in place to respond to and mitigate the potential impact of security incidents. The architecture of your workload affects the ability of your teams to operate effectively during an incident, to isolate or contain systems and to restore operations to a known good state. Putting in place the tools and access ahead of a security incident, then routinely practicing incident response through game days, will help you ensure that your architecture can accommodate timely investigation and recovery.

- **Stay up to date**: Staying up to date with AWS and industry recommendations and threat intelligence helps you evolve your threat model and control objectives. Automating security processes, testing, and validation allow you to scale your security operations.

- Finances

### *(AWS) Billing and Cost Management*

This is the service that you use to pay your AWS bill, monitor your usage, and analyse and control your costs.

AWS automatically charges the credit card that you provided when you signed up fora new account with AWS. Charges appear on your monthly credit card bill. You can view or update your credit card information, including designating a different credit card for AWS to charge, on the Payment Methods page in the Billing and Cost Management console. You can set a specific payment currency here also. AWS Billing and Cost Management provides useful tools to help you gather information related to your cost and usage, analyse your cost drivers and usage trends, and take action to budget your spending.

- **Analyzing Costs with Cost Explorer**
    - The AWS Billing and Cost Management console includes the no-cost Cost Explorer tool for viewing your AWS cost data as a graph. With Cost Explorer, you can filter graphs by values such as API operation, Availability Zone, AWS service, custom cost allocation tag, Amazon EC2 instance type, purchase option, AWS Region, usage type, usage type group, and more. If you use consolidated billing, you can also filter by member account. In addition, you can see a forecast of future costs based on your historical cost data.
    - Cost allocation tags – are key-value pairs that allow you to organize your AWS resources into groups. For each resource, each tag key must be unique, and each tag key can have only one value. AWS provides two types of cost allocation tags, an AWS generated tags and user-defined tags. You can use tags to:
        - organize your resources, and cost allocation tags to track your AWS costs on a detailed level
        - Visualize information about tagged resources in one place, in conjunction with Resource Groups.
        - View billing information using Cost Explorer and the AWS Cost and Usage report.
        - Send notifications about spending limits using AWS Budgets.
        - Use logical groupings of your resources that make sense for your infrastructure or business. For example, you could organize your resources by:
            - Project
            - Cost center
            - Development environment
            - Application
            - Department
- **AWS Budgets**
    - You can use AWS Budgets to track your AWS usage and costs. Budgets use the cost visualization provided by Cost Explorer to show you the status of your budgets. This provides forecasts of your estimated costs and tracks your AWS usage, including your free tier usage. You can also use budgets to create Amazon Simple Notification Service (Amazon SNS) notifications that tell you when you go over your budgeted amounts, or when your estimated costs exceed your budgets.

### *Total Cost of Ownership (TCO) Calculator*

The TCO tool makes a comparison between On Premise IT infrastructure expense the equivalent expense that would exist in the AWS cloud. It then lets the customer know what their cost savings would be if they decided to move their existing IT infrastructure to the AWS cloud.

### *(AWS) Pricing Calculator*

Configure a cost estimate that fits your unique business or personal needs with AWS products and services. Previously known as Simply Monthly Calculator. Transparent pricing lets you see the math behind the price for your service configurations. View prices per service or per group of services to analyse your architecture costs.

Configure services, or groups of services, in multiple AWS Regions. Prices and availability of AWS services vary per Region.

See and analyse service costs grouped by different parts of your architecture.

Export your estimate to a .csv file to quickly share and analyse your proposed architecture spend.

### *Amazon CloudWatch Billing Monitoring and Alerts*

You can monitor your estimated AWS charges by using Amazon CloudWatch. When you enable the monitoring of estimated charges for your AWS account, the estimated charges are calculated and sent several times daily to CloudWatch as metric data.

Billing metric data is stored in the US East (N. Virginia) Region and represents worldwide charges. This data includes the estimated charges for every service in AWS that you use, in addition to the estimated overall total of your AWS charges.

Alerts and alarms can be set up to notify you when you have reached a specific usage cost in your AWS account. It's a notification that you will receive automatically when a certain level of AWS spend has been reached. This can be set up globally in your AWS account in the Billing & Cost Management Dashboard and region specific in the CloudWatch service.

(http://kayleigholiver.com/aws-cloud-practitioner-aws-cost-management/)

### *(AWS) Cost & Usage Report (CUR)*

The AWS Cost and Usage Reports contains the most comprehensive set of cost and usage data available. You can use Cost and Usage Reports to publish your AWS billing reports to an Amazon Simple Storage Service (Amazon S3) bucket that you own.

You can receive reports that break down your costs by the hour or day, by product or product resource, or by tags that you define yourself. AWS updates the report in your bucket (once a day by default, but customisable to up to three times a day) in comma-separated value (CSV) format. Each update is cumulative, so each version of the Cost and Usage Reports includes all of the line items and information from the previous version.

The reports generated throughout the month are estimated, and subject to change during the rest of the month as you continue to use your AWS services. AWS finalizes the report at the end of each month. Finalized reports have the calculations for your blended and unblended costs, and cover all of your usage for the month. AWS might update reports after they have been finalized if AWS applies refunds, credits, or support fees to your usage for the month.

The report is available within 24 hours of the date that you create a report on the Cost & Usage Reports page of the Billing and Cost Management console.

You can view the reports using spreadsheet software such as Microsoft Excel or Apache OpenOffice Calc, or access them from an application using the Amazon S3 API. You can also load your cost and usage information into Amazon Athena, Amazon Redshift, AWS QuickSight, or a tool of your choice.

You can create, retrieve, and delete your reports using the AWS CUR API Reference

AWS Cost and Usage Reports tracks your AWS usage and provides estimated charges associated with your account. Each report contains line items for each unique combination of AWS products, usage type, and operation that you use in your AWS account.

### *(AWS) Organizations*

- AWS Organizations helps you centrally manage and govern your environment as you grow and scale your AWS resources. As an administrator of an organization, you can create accounts in your organization and invite existing accounts to join the organization.
- Allows you to:
    - programmatically create new AWS accounts and allocate resources
    - group accounts to organize your workflows
    - apply policies to accounts or groups for governance
    - define central configurations and audit requirements
    - simplify billing by centralising it and using a single payment method for all of your account. These account management and consolidated billing capabilities enable you to better meet the budgetary, security, and compliance needs of your business.
    - control access, manage compliance, coordinate security mechanisms (including restricting the AWS services, resources, and individual API actions accessible by specific users, groups and roles)
    - share resources across your AWS accounts.
    - combine usage from all accounts in the organization to qualify you for volume pricing discounts. If you have multiple standalone accounts, your charges might decrease if you add the accounts to an organization.
- AWS Organizations is tightly integrated with other AWS services
- AWS Organizations is offered at no additional charge. You are charged only for AWS resources that users and roles in your member accounts use.
- Important to not place resources in the master account
- AWS Support plans on the master account of an organization do not automatically apply to member accounts in the organization

### *Sharing Resources in an Organizations Master Account*

- If you would like to share resources with your organization or organizational units, then you must use the AWS RAM console or CLI command to enable sharing with AWS Organizations.
    - The account that originally purchased the Reserved Instance receives the discount first. If the purchasing account doesn't have any instances that match the terms of the Reserved Instance, the discount for the Reserved Instance is assigned to any matching usage on another account in the organization. For billing purposes, the consolidated billing feature of AWS Organizations treats all the accounts in the organization as one account. This means that all accounts in the organization can receive the hourly cost benefit of Reserved Instances that are purchased by any other account.
    - If Reserved Instance sharing is turned off for an account in an organization. Reserved Instance discounts apply only to the account that purchased the Reserved Instance.

*Removing Accounts from an Organizations Master Account*

Requirements for removing an account from an organization:

- The account that you want to remove must have the information that is required for it to operate as a standalone account
- The account that you want to remove must not be a delegated administrator account for any AWS service enabled for your organization
- Customers' agreements with us, and the rights and obligations under those agreements, cannot be assigned or transferred without our prior consent. To obtain our consent, contact us at https://aws.amazon.com/contact-us/

How to remove an account from an organization:

- Sign in as an IAM user or role in the management account with the required permissions
    - Go to the 'Accounts' tab and select 'Remove account' for the account you wish to remove
    - AWS will redirect you the AWS Organizations console for the chosen member account, here select 'Leave organization'
    - Remove the IAM roles that grant access to your member account from the organization.
- Sign in as an IAM user or role in the member account with the required permissions
    - Go to the 'Organization' page and choose 'Leave Organization'
    - Remove the IAM roles that grant access to your account from the organization.

Effects of removing an account from an organization

- When a member account leaves an organization, that account no longer has access to cost and usage data from the time range when the account was a member of the organization. If an account rejoins an organization that it previously belonged to, the account regains access to its historical cost and usage data.
- When a member account leaves an organization, all tags attached to the account are deleted.
- The account is now responsible for paying its own charges and must have a valid payment method attached to the account.
- The principals in the account are no longer affected by any policies that applied in the organization. This means that restrictions imposed by SCPs are gone, and the users and roles in the account might have more permissions than they had before. Other organization policy types can no longer enforced or processed
- Integration with other services might be disabled. For example, AWS Single Sign-On (SSO) requires an organization to operate, so if you remove an account from an organization that supports AWS SSO, the users in that account can no longer use that service

# Support

## *Customer Support*

**Basic Support** is included for all AWS customers and includes:

- Customer Service and Communities - 24x7 access to customer service, documentation, whitepapers, and support forums.
- AWS Trusted Advisor –
    - Guidance to provision your resources following best practices to increase performance and improve security. AWS Trusted Advisor is an online tool that provides you real time guidance to help you provision your resources following AWS best practices. Trusted Advisor checks help optimize your AWS infrastructure, increase security and performance, reduce your overall costs, and monitor service limits. Whether establishing new workflows, developing applications, or as part of ongoing improvement, take advantage of the recommendations provided by Trusted Advisor on a regular basis to help keep your solutions provisioned optimally.
    - AWS Trusted Advisor analyzes your AWS environment and provides best practice recommendations in five categories:
        - **P**erformance: AWS Trusted Advisor can improve the performance of your service by checking your service limits, ensuring you take advantage of provisioned throughput, and monitoring for overutilized instances.
        - Service **Q**uotas: AWS Trusted Advisor checks for service usage that is more than 80% of the service quota.
            - AWS maintains service quotas (formerly called service limits) for each account to help guarantee the availability of AWS resources and prevent accidental provisioning of more resources than needed.
            - Some service quotas are raised automatically over time as you use AWS. However, most AWS services require that you request quota increases manually. You can use AWS Service Quotas console to view and request increases for most AWS quotas.
            - Values are based on a snapshot, so your current usage might differ. Limit and usage data can take up to 24 hours to reflect any changes.
        - Cost optimization/**R**eduction: AWS Trusted Advisor can save you money on AWS by eliminating unused and idle resources or by making commitments to reserved capacity.
        - **S**ecurity: AWS Trusted Advisor can improve the security of your application by closing gaps, enabling various AWS security features, and examining your permissions.
        - Fault **T**olerance: AWS Trusted Advisor can increase the availability and redundancy of your AWS application by take advantage of auto scaling, health checks, multi AZ, and backup capabilities.

        - Easily remembered as **...PQRST...** (like the alphabet)

- AWS Basic Support and AWS Developer Support customers get access to 6 security checks (listed below) and 50 service limit checks (to see how close you are to exceeding use quotas):
  - S3 Bucket Permissions
  - Security Groups – Specific Ports Unrestricted
  - IAM Use
  - MFA on Root Account
  - EBS Public Snapshots
  - RDS Public Snapshots
- AWS Business Support and AWS Enterprise Support customers get access to all 115 Trusted Advisor checks (14 cost optimization, 17 security, 24 fault tolerance, 10 performance, and 50 service limits) and recommendations.

### *(AWS) Health*

Provides ongoing visibility into your resource performance and the availability of your AWS services and accounts. You can use AWS Health events to learn how service and resource changes might affect your applications running on AWS. AWS Health provides relevant and timely information to help you manage events in progress. AWS Health also helps you be aware of and to prepare for planned activities. The service delivers alerts and notifications triggered by changes in the health of AWS resources, so that you get near-instant event visibility and guidance to help accelerate troubleshooting. Additionally, AWS Support customers who have a Business or Enterprise support plan can use the AWS Health API to integrate with in-house and third-party systems.

- (AWS) Personal Health Dashboard (PHD) - All customers can use this, it is powered by the AWS Health API. A personalized view of the health of AWS services, and alerts when your resources are impacted. It provides alerts and remediation guidance when AWS is experiencing events that may impact you. Personal Health Dashboard gives you a personalized view into the performance and availability of the AWS services underlying your AWS resources. The dashboard requires no setup, and it's ready to use for authenticated AWS users.

### *(AWS) Service Health Dashboard*

Displays the general status of AWS services. The dashboard provides access to current status and historical data about each and every Amazon Web Service. If there's a problem with a service, you'll be able to expand the appropriate line in the Details section. You can even subscribe to the RSS feed for any service. You can use the "Report an Issue" link to make sure that we are aware of any system-wide service issues. You will be able to see a record of service status, on a per-service basis, for the previous 35 days.

## *Support Levels*

| | Developer | Business | Enterprise |
|---|---|---|---|
| | **Developer** | **Business** | **Enterprise** |
| | *Recommended if you are experimenting or testing in AWS.* | *Recommended if you have production workloads in AWS.* | *Recommended if you have business and/or mission critical workloads in AWS.* |
| **AWS Trusted Advisor Best Practice Checks** | 7 Core checks | Full set of checks | Full set of checks |
| **Enhanced Technical Support** | Business hours** email access to Cloud Support Associates<br>Unlimited cases / 1 primary contact | 24x7 phone, email, and chat access to Cloud Support Engineers<br>Unlimited cases / unlimited contacts (IAM supported) | 24x7 phone, email, and chat access to Cloud Support Engineers<br>Unlimited cases / unlimited contacts (IAM supported) |
| **Case Severity / Response Times*** | General guidance: < 24 hours**<br>System impaired: < 12 hours** | General guidance: < 24 hours<br>System impaired: < 12 hours<br>Production system impaired: < 4 hours<br>Production system down: < 1 hour | General guidance: < 24 hours<br>System impaired: < 12 hours<br>Production system impaired: < 4 hours<br>Production system down: < 1 hour<br>Business-critical system down: < 15 minutes |
| **Architectural Guidance** | General | Contextual to your use-cases | Consultative review and guidance based on your applications |
| **Programmatic Case Management** | | AWS Support API | AWS Support API |
| **Third-Party Software Support** | | Interoperability and configuration guidance and troubleshooting | Interoperability and configuration guidance and troubleshooting |
| **Proactive Programs and Services** | | Access to Infrastructure Event Management for additional fee | Infrastructure Event Management<br>Well-Architected Reviews<br>Access to proactive reviews, workshops, and deep dives |
| **Technical Account Management** | | | Designated Technical Account Manager (TAM) to proactively monitor your environment and assist with optimization and coordinate access to programs and AWS experts |
| **Training** | | | Access to online self-paced labs |
| **Account Assistance** | | | Concierge Support Team |
| **Pricing** | Greater of $29 / month***<br>- or -<br>3% of monthly AWS usage<br>See pricing detail and example. | Greater of $100 / month***<br>- or -<br>10% of monthly AWS usage for the first $0–$10K<br>7% of monthly AWS usage from $10K–$80K<br>5% of monthly AWS usage from $80K–$250K<br>3% of monthly AWS usage over $250K<br>See pricing detail and example. | Greater of $15,000<br>- or -<br>10% of monthly AWS usage for the first $0–$150K<br>7% of monthly AWS usage from $150K–$500K<br>5% of monthly AWS usage from $500K–$1M<br>3% of monthly AWS usage over $1M<br>See pricing detail and example. |

*We will make every reasonable effort to respond to your initial request within the corresponding timeframes.

**Business hours are generally defined as 8:00 AM to 6:00 PM in the customer country as set in My Account console, excluding holidays and weekends. These times may vary in countries with multiple time zones.

*** Plans are subject to a 30 day minimum term.

Note: if you work with an AWS partner and would like to learn more about Partner-led Support, click here.

(https://aws.amazon.com/premiumsupport/plans/)

### *Infrastructure Event Management (IEM)*

Offers architecture and scaling guidance and operational support during the preparation and execution of planned events, such as shopping holidays, product launches, and migrations. For these events, AWS Infrastructure Event Management will help you assess operational readiness, identify and mitigate risks, and execute your event confidently with AWS experts by your side.

### *AWS Concierge*

Your AWS Concierge is a senior customer service agent who is assigned to your account when you subscribe to an Enterprise or qualified Reseller Support plan.

This Concierge agent is your primary point of contact for billing or account inquiries; when you don't know whom to call, they will find the right people to help.

In most cases, the AWS Concierge is available during regular business hours in your headquarters' geography. Outside of business hours, the global customer service team can assist you 24x7x365. The best way to contact the AWS Concierge is through the AWS Support Center.

### *Technical Account Manager (TAM)*

Only available for Enterprise support level customers. A Technical Account Manager (TAM) is your designated technical point of contact who helps you onboard, provides advocacy and guidance to help plan and build solutions using best practices, coordinates access to subject matter experts, assists with case management, presents insights and recommendations on your AWS spend, workload optimization, and event management, and proactively keeps your AWS environment healthy.

### *AWS Support API*

The AWS Support API provides access to some of the features of the AWS Service Catalog. AWS provides this access for AWS Support customers who have a Business or Enterprise support plan. The service currently provides two different groups of operations:

- Support case management operations to manage the entire life cycle of your AWS support cases, from creating a case to resolving it. You can perform these tasks:
    - Open a support case.
    - Get a list and detailed information about recent support cases.
    - Narrow your search for support cases by dates and case identifiers, including cases that are resolved.
    - Add communications and file attachments to your cases, and add the email recipients for case correspondence.
    - Resolve your cases.
- Trusted Advisor operations to access the checks provided by AWS Trusted Advisor. You can perform these tasks:
    - Get names and identifiers for the checks that Trusted Advisor offers.
    - Request that a Trusted Advisor check be run against your account and resources.
    - Obtain summaries and detailed information for your Trusted Advisor checks.
    - Request that Trusted Advisor checks be refreshed.
    - Obtain the status of each Trusted Advisor check you have requested.
    - Also, AWS Support API supports CloudWatch Events for Trusted Advisor operations

*APN*

The **AWS Partner Network** (APN) is the global partner program for technology and consulting businesses who leverage Amazon Web Services to build solutions and services for customers. The APN helps companies build, market, and sell their AWS offerings by providing valuable business, technical, and marketing support. There are two main APN partner types:

- **APN Technology Partners** provide hardware, connectivity services, or software solutions that are hosted on, or integrated with, the AWS Cloud.
    - Hardware providers include original equipment manufacturers (OEMs) and semiconductor manufacturers.
    - Connectivity services providers include network carriers.
    - Software solution providers include SaaS providers and independent software vendors (ISVs).
- **AWS Consulting Partners** are professional services firms that help customers of all types and sizes design, architect, build, migrate, and manage their workloads and applications on AWS, accelerating their journey to the cloud. These professional services firms include system integrators, strategic consultancies, agencies, managed service providers (MSPs), and value-added resellers.
    - **AWS Managed Service Provider (MSP) Partners** provide customers full lifecycle solutions in cloud infrastructure and application migration. They offer support in four key areas: plan and design; build and migrate; run and operate; and optimize. AWS MSP Partners receive their designation by undergoing an extensive third-party validation audit that demonstrates next-generation managed service practices.

**APN Delivery Partners** accelerate customers' cloud migration by providing technical support, personnel, and professional services. Delivery Partners take additional responsibilities for customers' migration implementation and project ownership.

The **AWS Professional Services** organization is a global team of experts that can help you realize your desired business outcomes when using the AWS Cloud. We work together with your team and your chosen member of the AWS Partner Network (APN) to execute your enterprise cloud computing initiatives. Our team provides assistance through a collection of offerings which help you achieve specific outcomes related to enterprise cloud adoption. We also deliver focused guidance through our global specialty practices, which cover a variety of solutions, technologies, and industries. In addition to working alongside our customers, we share our experience through tech talk webinars, White Papers, and blog posts that are available to anyone.

- **Supplementing your team with specialized skills and experience** AWS Professional Services provides global specialty practices to support your efforts in focused areas of enterprise cloud computing e.g. Blockchain, quantum computing, robotics, space, Internet of Things & AI/Machine Learning. Specialty practices deliver targeted guidance through best practices, frameworks, tools, and services across solution, technology, and industry subject areas. Their deep expertise helps you take advantage of business benefits available with the AWS Cloud.

## *AWS Marketplace*

The AWS Marketplace enables qualified partners to market and sell their software to AWS Customers. AWS Marketplace is an online software store that helps customers find, buy, and immediately start using the software and services that run on AWS.

AWS Marketplace is designed for Independent Software Vendors (ISVs), Value-Added Resellers (VARs), and Systems Integrators (SIs) who have software products they want to offer to customers in the cloud. Partners use AWS Marketplace to be up and running in days and offer their software products to customers around the world.

Customers can quickly launch pre-configured software with just a few clicks, and choose software solutions in Amazon Machine Images (AMIs) and software as a service (SaaS) formats, as well as other formats. Additionally, you can browse and subscribe to data products. Flexible pricing options include free trial, hourly, monthly, annual, multi-year, and BYOL (Bring Your Own License), and get billed from one source. AWS handles billing and payments, and charges appear on customers' AWS bill.

(https://aws.amazon.com/partners/aws-marketplace)

(https://aws.amazon.com/about-aws/whats-new/2019/09/aws-marketplace-easier-to-find-solutions-from-aws-console/)

## *(AWS) Service Catalog*

- AWS Service Catalog allows organizations to create and manage catalogs of IT services that are approved for use on AWS. These IT services can include everything from virtual machine images, servers, software, and databases to complete multi-tier application architectures. AWS Service Catalog allows you to centrally manage deployed IT services and your applications, resources, and metadata. This helps you achieve consistent governance and meet your compliance requirements, while enabling users to quickly deploy only the approved IT services they need.
- With AWS Service Catalog AppRegistry, organizations can understand the application context of their AWS resources. You can define and manage your applications and their metadata, to keep track of cost, performance, security, compliance and operational status at the application level.
- AWS Service Catalog Delivery Partners are APN Consulting Partners who help create catalogs of IT services that are approved by the customer's organization for use on AWS. With AWS Service Catalog, customers and partners can centrally manage commonly deployed IT services to help achieve consistent governance and meet compliance requirements while enabling users to self-provision approved services.
- The AWS Service Delivery Program enables AWS customers to identify APN Consulting Partners with experience and a deep understanding of specific AWS services. These APN Partners follow best practices for AWS services and have proven success delivering AWS services to customers.

### *(AWS) Personal Health Dashboard*

AWS Personal Health Dashboard provides alerts and remediation guidance when AWS is experiencing events that may impact you. While the Service Health Dashboard displays the general status of AWS services, Personal Health Dashboard gives you a personalized view into the performance and availability of the AWS services underlying your AWS resources.

The dashboard displays relevant and timely information to help you manage events in progress, and provides proactive notification to help you plan for scheduled activities. With Personal Health Dashboard, alerts are triggered by changes in the health of AWS resources, giving you event visibility, and guidance to help quickly diagnose and resolve issues.

Detailed troubleshooting guidance - When you get an alert, it includes remediation details and specific guidance to enable you to take immediate action to address AWS events impacting your resources. For example, in the event of an AWS hardware failure impacting one of your Amazon Elastic Block Store (EBS) volumes, your alert would include a list of your affected resources, a recommendation to restore your volume, and links to the steps to help you restore it from a snapshot. This targeted and actionable information reduces the time needed to resolve issues.

Aggregate health events across AWS Organizations - If you use AWS Organizations, AWS Health allows you to aggregate notifications from all accounts in your organization. This provides centralized and real-time access to all AWS Health events posted to individual accounts in your organization, including operational issues, scheduled maintenance, and account notifications.

Integration and automation - AWS Personal Health Dashboard can integrate with Amazon CloudWatch Events, enabling you to build custom rules and select targets such as AWS Lambda functions to define automated remediation actions. The AWS Health API, the underlying service powering Personal Health Dashboard, allows you to integrate health data and notifications with your existing in-house or third party IT Management tools.

AWS Personal Health Dashboard is available to all AWS customers, and provides status and notifications for all services across all Regions and Availability Zones. Access to the AWS Health API is included as part of the AWS Business Support and AWS Enterprise Support plans.

Fine-grained access control via IAM policy conditions - AWS Personal Health Dashboard gives you fine-grained access control so that you can setup permissions based on event metadata. This allows you to grant or deny access to an AWS Identity and Access Management (IAM) user based on such attributes as event types, event types of a particular service, or other role-based attributes. With fine-grained access control, you can limit access of sensitive alerts, such as those related to security, to only those users that need to see them.

### *Difference between (AWS) Service Health Dashboard and (AWS) Personal Health Dashboard*

While the Service Health Dashboard displays the GENERAL status of AWS services, Personal Health Dashboard gives you a PERSONALIZED VIEW into the performance and availability of the AWS services underlying your AWS resources.

The difference between Personal and Health dashboards is "Health" provides the "generic status of overall AWS services and on in particular. But "Personal" provides status of services pertaining to "subscribed" AWS services. This is why they would call it "Personal"

### *Amazon Customer Engagement (ACE) Program*

The APN Customer Engagements (ACE) Program enables AWS Partners to build, grow, and drive successful customer engagements with AWS Sales. It provides Partners with a platform to collaborate with AWS Sales and Marketing teams, request funding, and technical support to help you co-sell with AWS.

### *(AWS) Partner Transformation Program*

The AWS Partner Transformation Program (PTP) is a comprehensive assessment, training, and enablement program focused on helping you build a successful and profitable AWS Cloud business. Whether you are new to the cloud or in the advanced stages of building your cloud business, this program provides partners with the guidance to accelerate the development of your AWS skills and expertise to better serve your customers' journey to the cloud.

Through the PTP, AWS helps partners expedite cloud readiness in key business areas to help customers migrate to the cloud. The result is partner transformation, building an innovative cloud business for partners to better serve the ultimate customer.

The PTP is open to partners that are either new to cloud and need help with planning for cloud migration, or have started the process but need help in accelerating their journey. Every PTP partner receives a customized Transformation Plan to accelerate their journey to AWS and support in the execution of identified activities.

### *AWS Account Abuse Team*

The AWS Abuse team can assist you when AWS resources are used to engage in the following types of abusive behavior:

- Spam: You are receiving unwanted emails from an AWS-owned IP address, or AWS resources are used to spam websites or forums.
- Port scanning: Your logs show that one or more AWS-owned IP addresses are sending packets to multiple ports on your server, and you believe this is an attempt to discover unsecured ports.
- Denial-of-service (DoS) attacks: Your logs show that one or more AWS-owned IP addresses are used to flood ports on your resources with packets, and you believe that this is an attempt to overwhelm or crash your server or the software running on your server.
- Intrusion attempts: Your logs show that one or more AWS-owned IP addresses are used to attempt to log in to your resources.
- Hosting objectionable or copyrighted content: You have evidence that AWS resources are used to host or distribute illegal content or distribute copyrighted content without the consent of the copyright holder.
- Distributing malware: You have evidence that AWS resources are used to distribute software that was knowingly created to compromise or cause harm to computers or machines on which it is installed.

## Services

### *Compute Services*

- Amazon EC2
- Amazon EC2 Auto Scaling
- Amazon Elastic Container Registry
- Amazon Elastic Container Service
- Amazon Elastic Kubernetes Service
- Amazon Lightsail
- AWS Batch
- AWS Elastic Beanstalk
- AWS Fargate
- AWS Lambda
- AWS Serverless Application Repository
- AWS Outposts
- VMware Cloud on AWS

### *Storage Services*

- Amazon Simple Storage Service (S3) / Amazon S3 Glacier
- Amazon Elastic Block Store (EBS)
- Amazon Elastic File System (EFS)
- AWS Storage Gateway
- AWS Snow Family (Snowcone, Snowball & Snowmobile)
- Amazon FSx for Lustre
- Amazon FSx for Windows File Server

### *Database Services*

- Amazon RDS
- Amazon Aurora
- Amazon DynamoDB
- Amazon Redshift
- Amazon ElastiCache for Memcached
- Amazon ElastiCache for Redis
- Amazon DocumentDB (with MongoDB compatibility)
- Amazon Keyspaces (for Apache Cassandra)
- Amazon Neptune
- Amazon Timestream
- Amazon QLDB (Quantum Ledger Database)

### *Elastic Cloud Compute (EC2)*

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

Amazon EC2 offers the broadest and deepest compute platform with choice of processor, storage, networking, operating system, and purchase model. We offer the fastest processors in the cloud and we are the only cloud with 400 Gbps ethernet networking. We have the most powerful GPU instances for machine learning training and graphics workloads, as well as the lowest cost-per-inference instances in the cloud. More SAP, HPC, Machine Learning, and Windows workloads run on AWS than any other cloud.

The customer is responsible for managing, support, patching and control of the guest operating system of EC2 instances.

### *Amazon Machine Image (AMI)*

- AMIs common software configurations for public use. In addition, members of the AWS developer community have published their own custom AMIs. AMIs enable you to quickly and easily start new instances that have everything you need.
    - For example, if your application is a website or a web service, your AMI could include a web server, the associated static content, and the code for the dynamic pages. As a result, after you launch an instance from this AMI, your web server starts, and your application is ready to accept requests.
- An AMI provides the information required to launch an instance. You must specify an AMI when you launch an instance. You can launch multiple instances from a single AMI when you need multiple instances with the same configuration.
- An AMI includes the following:
    - One or more EBS snapshots, or, for instance-store-backed AMIs, a template for the root volume of the instance (for example, an operating system, an application server, and applications).
    - Launch permissions that control which AWS accounts can use the AMI to launch instances.
    - A block device mapping that specifies the volumes to attach to the instance when it's launched.
- You can use a bootstrap action to install additional software, dependencies or customize the configuration of AMI instances.

### *AWS Auto Scaling*

AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost. Using AWS Auto Scaling, it's easy to setup application scaling for multiple resources across multiple services in minutes.

The service provides a simple, powerful user interface that lets you build scaling plans for resources including Amazon EC2 instances and Spot Fleets, Amazon ECS tasks, Amazon DynamoDB tables and indexes, and Amazon Aurora Replicas. AWS Auto Scaling makes scaling simple with recommendations that allow you to optimize performance, costs, or balance between them.

AWS Auto Scaling is available at no additional charge. You pay only for the AWS resources needed to run your applications and Amazon CloudWatch monitoring fees.

Amazon EC2 Auto Scaling can detect when an instance is unhealthy, terminate it, and launch an instance to replace it. You can also configure Amazon EC2 Auto Scaling to use multiple Availability Zones. If one Availability Zone becomes unavailable, Amazon EC2 Auto Scaling can launch instances in another one to compensate.

### *On Demand Instances*

On-Demand Instances let you pay for compute capacity by the hour or second (minimum of 60 seconds) with no long-term commitments. You have full control over its lifecycle—you decide when to launch, stop, hibernate, start, reboot, or terminate it. This frees you from the costs and complexities of planning, purchasing, and maintaining hardware and transforms what are commonly large fixed costs into much smaller variable costs.

Pricing is per instance-hour consumed for each instance, from the time an instance is launched until it is terminated or stopped. Each partial instance-hour consumed will be billed per-second for Linux Instances and as a full hour for all other instance types.

There is no long-term commitment required when you purchase On-Demand Instances. You pay only for the seconds that your On-Demand Instances are in the running state. The price per second for a running On-Demand Instance is fixed.

We recommend that you use On-Demand Instances for applications with short-term, irregular workloads that cannot be interrupted.

For significant savings over On-Demand Instances, use AWS Savings Plans, Spot Instances, or Reserved Instances.

*Reserved Instances*

A Reserved Instance is a reservation of resources and capacity, for either one or three years, for a particular Availability Zone within a region. When you purchase a reservation, you commit to paying for all of the hours of the 1- or 3-year term; in exchange, the hourly rate is lowered significantly.

Amazon EC2 Reserved Instances (RI) provide a significant discount (up to 72%) compared to On-Demand pricing and provide a capacity reservation when used in a specific Availability Zone. AWS Billing automatically applies your RI's discounted rate when attributes of EC2 instance usage match attributes of an active RI.

If an Availability Zone is specified, EC2 reserves capacity matching the attributes of the RI. The capacity reservation of an RI is automatically utilized by running instances matching these attributes.

You can also choose to forego the capacity reservation and purchase an RI that is scoped to a region. RIs that are scoped to a region automatically apply the RI's discount to instance usage across AZs and instance sizes in a region, making it easier for you to take advantage of the RI's discounted rate.

With RIs, you can choose the type that best fits your applications needs:

- **Standard RIs:** These provide the most significant discount (up to 72% off On-Demand) and are best suited for steady-state usage.
- **Convertible RIs**: These provide a discount (up to 54% off On-Demand) and the capability to change the attributes of the RI (instance family, operating system, and tenancy) as long as the exchange results in the creation of Reserved Instances of equal or greater value (even if this means switching RIs to a different instance family). There are no limits to how many times you perform an exchange. Like Standard RIs, Convertible RIs are best suited for steady-state usage.
- **Scheduled RIs**: These are available to launch within the time windows you reserve. This option allows you to match your capacity reservation to a predictable recurring schedule that only requires a fraction of a day, a week, or a month.

(https://support.cloudability.com/hc/en-us/articles/204307758-AWS-101-Reserved-Instances)

**Reserved Instance Savings Guide**

1. Standard one-year - all upfront = up to 72%
2. Standard three-years - all upfront = up to 72%
3. Standard one-year - all no upfront = 40%
4. Standard three-years - all no upfront = 60%
5. Convertible one-year - all upfront = up to 54%
6. Convertible three-years - all upfront = up to 54%
7. Convertible one-year- all no upfront = 31%
8. Convertible three-years - all no upfront = 54%

When you purchase a Reserved Instance, you determine the scope of the Reserved Instance. The scope is either regional or zonal.

- Regional: When you purchase a Reserved Instance for a Region, it's referred to as a regional Reserved Instance.
- Zonal: When you purchase a Reserved Instance for a specific Availability Zone, it's referred to as a zonal Reserved Instance.

| | Regional Reserved Instances | Zonal Reserved Instances |
|---|---|---|
| Availability Zone flexibility | The Reserved Instance discount applies to instance usage in any Availability Zone in the specified Region. | No Availability Zone flexibility—the Reserved Instance discount applies to instance usage in the specified Availability Zone only. |
| Ability to reserve capacity | A regional Reserved Instance does *not* reserve capacity. | A zonal Reserved Instance reserves capacity in the specified Availability Zone. |
| Instance size flexibility | The Reserved Instance discount applies to instance usage within the instance family, regardless of size. Only supported on Amazon Linux/Unix Reserved Instances with default tenancy. For more information, see Instance size flexibility determined by normalization factor. | No instance size flexibility—the Reserved Instance discount applies to instance usage for the specified instance type and size only. |

### *On-Demand Capacity Reservations*

- On-Demand Capacity Reservations enable you to reserve capacity for your Amazon EC2 instances in a specific Availability Zone for any duration. This gives you the ability to create and manage Capacity Reservations independently from the billing discounts offered by Savings Plans or regional Reserved Instances.
- By creating Capacity Reservations, you ensure that you always have access to EC2 capacity when you need it, for as long as you need it. You can create Capacity Reservations at any time, without entering into a one-year or three-year term commitment, and the capacity is available immediately. When you no longer need it, cancel the Capacity Reservation to stop incurring charges.
- On-Demand Capacity Reservations are priced exactly the same as their equivalent (On-Demand) instance usage. If a Capacity Reservation is fully utilized, you only pay for instance usage and nothing towards the Capacity Reservation. If a Capacity Reservation is partially utilized, you pay for the instance usage and for the unused portion of the Capacity Reservation.
- When you create a Capacity Reservation, you specify:
    - The Availability Zone in which to reserve the capacity
    - The number of instances for which to reserve capacity
    - The instance attributes, including the instance type, tenancy, and platform/OS
- Capacity Reservations can only be used by instances that match their attributes. By default, they are automatically used by running instances that match the attributes. If you don't have any running instances that match theattributes of the Capacity Reservation, it remains unused until you launch an instance with matching attributes.
- In addition, you can use Savings Plans and regional Reserved Instances with your Capacity Reservations to benefit from billing discounts. AWS automatically applies your discount when the attributes of a Capacity Reservation match the attributes of a Savings Plan or regional Reserved Instance.

### *Differences between Capacity Reservations, Reserved Instances, and Savings Plans*

The following table highlights key differences between Capacity Reservations, Reserved Instances, and Savings Plans:

| | Capacity Reservations | Zonal Reserved Instances | Regional Reserved Instances | Savings Plans |
|---|---|---|---|---|
| **Term** | No commitment required. Can be created and cancelled as needed. | Require fixed one-year or three-year commitment | | |
| **Capacity benefit** | Capacity reserved in a specific Availability Zone. | | Do not reserve capacity in an Availability Zone. | |
| **Billing discount** | No billing discount. Instances launched into a Capacity Reservation are charged at their standard On-Demand rates. However, you can use Savings Plans or regional Reserved Instances with Capacity Reservations to get a billing discount. Zonal Reserved Instances do not apply to Capacity Reservations. | Provide billing discounts | | |
| **Instance Limits** | Limited to your On-Demand Instance limits per Region. | Limited to 20 per Availability Zone. A limit increase can be requested. | Limited to 20 per Region. A limit increase can be requested. | No limits. |

### *Spot Instances*

Amazon EC2 Spot Instances let you take advantage of unused EC2 capacity in the AWS cloud. Spot Instances are available at up to a 90% discount compared to On-Demand prices. You can use Spot Instances for various stateless, fault-tolerant, or flexible applications such as big data, containerized workloads, CI/CD, web servers, high-performance computing (HPC), and test & development workloads. Because Spot Instances are tightly integrated with AWS services such as Auto Scaling, EMR, ECS, CloudFormation, Data Pipeline and AWS Batch, you can choose how to launch and maintain your applications running on Spot Instances.

Moreover, you can easily combine Spot Instances with On-Demand, RIs and Savings Plans Instances to further optimize workload cost with performance. Due to the operating scale of AWS, Spot Instances can offer the scale and cost savings to run hyper-scale workloads. You also have the option to hibernate, stop or terminate your Spot Instances when EC2 reclaims the capacity back with two-minutes of notice. Only on AWS, you have easy access to unused compute capacity at such massive scale - all at up to a 90% discount.

The Spot prices are determined by 'supply and demand' for Amazon EC2 spare capacity. The price per second for a running On-Demand Instance is fixed

### *Dedicated Instances*

Dedicated Instances are Amazon EC2 instances that run in a virtual private cloud (VPC) on hardware that's dedicated to a single customer. Dedicated Instances that belong to different AWS accounts are physically isolated at a hardware level, even if those accounts are linked to a single payer account. However, Dedicated Instances may share hardware with other instances from the same AWS account that are not Dedicated Instances.

### *Dedicated Hosts*

An Amazon EC2 Dedicated Host is a physical server with EC2 instance capacity fully dedicated to your use. Dedicated Hosts give you additional visibility and control over how instances are placed on a physical server, and you can reliably use the same physical server over time. As a result, Dedicated Hosts allow you to use your existing per-socket, per-core, or per-VM software licenses, including Windows Server, Microsoft SQL Server, SUSE, and Linux Enterprise Server; and address corporate compliance and regulatory requirements.

*Difference between Dedicated Instances and Dedicated Hosts*

- Dedicated instances and dedicated hosts are separate offerings.
    - Dedicated Instances are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer.
        - Your Dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts. This means that no other AWS Account will run an instance on the same Host, but other instances (both dedicated and non-dedicated) from the same AWS Account might run on the same Host.
        - A dedicated instance is partitioned under a hypervisor on a shared server
    - A dedicated host is a complete physical machine with a single partition that is dedicated to a single customer.
        - Other important differences between a Dedicated Host and a Dedicated instance is that a Dedicated Host gives you additional visibility and control over how instances are placed on a physical server, you have visibility over physical cores and visibility over socket usage. Also, you can consistently deploy your instances to the same physical server over time.
        - As a result, Dedicated Hosts enable you to use your existing server-bound software licenses (from vendors such as Microsoft and Oracle) and address corporate compliance and regulatory requirements.
        - Amazon EC2 Dedicated Hosts allow you to get the flexibility and cost effectiveness of using your own licenses, but with the resiliency, simplicity and elasticity of AWS.
        - Amazon EC2 Dedicated Host is also integrated with AWS License Manager (see below)
- In some cases due to licensing restrictions some software isn't allowed to be run on a shared tenancy model. For instance if you're trying to use Bring Your Own License (BYOL) to AWS, some licenses are based on the Socket model where the number of hosts sockets are used for licensing. In other circumstances, regulatory compliance may dictate that you can't use the shared model.
- Dedicated Hosts and Dedicated Instances can both be used to launch Amazon EC2 instances onto physical servers that are dedicated for your use. There are no performance, security, or physical differences between Dedicated Instances and instances on Dedicated Hosts. However, there are some differences between the two. The following table highlights some of the key differences between Dedicated Hosts and Dedicated Instances:

| | Dedicated Host | Dedicated Instance |
|---|---|---|
| Billing | Per-host billing | Per-instance billing |
| Visibility of sockets, cores, and host ID | Provides visibility of the number of sockets and physical cores | No visibility |
| Host and instance affinity | Allows you to consistently deploy your instances to the same physical server over time | Not supported |
| Targeted instance placement | Provides additional visibility and control over how instances are placed on a physical server | Not supported |

| | | |
|---|---|---|
| Automatic instance recovery | Supported. For more information, see Host recovery below. | Supported |
| Bring Your Own License (BYOL) | Supported | Not supported |

### *AWS License Manager*

A service which helps you manage your software licenses, including Microsoft Windows Server and Microsoft SQL Server licenses. In License Manager, you can specify your licensing terms for governing license usage, as well as your Dedicated Host management preferences for host allocation and host capacity utilization. Once setup, AWS takes care of these administrative tasks on your behalf, so that you can seamlessly launch virtual machines (instances) on Dedicated Hosts just like you would launch an EC2 instance with AWS provided licenses.

### *Host recovery*

Host recovery automatically restarts your instances on to a new replacement host if failures are detected on your Dedicated Host. Host recovery reduces the need for manual intervention and lowers the operational burden if there is an unexpected Dedicated Host failure.

Additionally, built-in integration with AWS License Manager automates the tracking and management of your licenses if a host recovery occurs.

### *EC2 Instance Store*

Some Amazon Elastic Compute Cloud (Amazon EC2) instance types come with a form of directly attached, block-device storage known as the instance store. The instance store is ideal for temporary storage, because the data stored in instance store volumes is not persistent through instance stops, terminations, or hardware failures.

For data you want to retain longer, or if you want to encrypt the data, use Amazon Elastic Block Store (Amazon EBS) volumes instead.

### *(Amazon) Elastic Block Store (EBS)*

Amazon Elastic Block Store (EBS) is an easy to use, high-performance, block-storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale. A broad range of workloads, such as relational and non-relational databases, enterprise applications, containerized applications, big data analytics engines, file systems, and media workflows are widely deployed on Amazon EBS.

You can choose from different volume types to balance optimal price and performance. You can achieve single-digit-millisecond latency for high-performance database workloads or gigabyte per second throughput for large, sequential workloads. You can change volume types, tune performance, or increase volume size without disrupting your critical applications, so you have cost-effective storage when you need it.

Designed for mission-critical systems, EBS volumes are replicated within an Availability Zone (AZ) and can easily scale to petabytes of data. Also, you can use EBS Snapshots with automated lifecycle policies to back up your volumes in Amazon S3, while ensuring geographic protection of your data and business continuity.

EBS volumes preserve their data through instance stops and terminations, can be removed from one instance and reattached to another, and support full-volume encryption. Best practice for performance on DBs is EBS, as instance store is ephemeral.

### *Load Balancers*

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, Lambda functions, and virtual appliances. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones. Elastic Load Balancing offers four types of load balancers that all feature the high availability, automatic scaling, and robust security necessary to make your applications fault tolerant. Elastic Load Balancing scales with web traffic.

- Application Load Balancers - best suited for load balancing of HTTP and HTTPS traffic and provides advanced request routing targeted at the delivery of modern application architectures, including microservices and containers. Application Load Balancer routes traffic to targets within Amazon VPC based on the content of the request. You can load balance HTTP/HTTPS applications for layer 7-specific features
- Network Load Balancers – best suited for load balancing of Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Transport Layer Security (TLS) traffic where extreme performance is required. Network Load Balancer routes traffic to targets within Amazon VPC and is capable of handling millions of requests per second while maintaining ultra-low latencies. You can use strict layer 4 load balancing for applications that rely on the TCP and UDP protocols.
- Gateway Load Balancers - makes it easy to deploy, scale, and run third-party virtual networking appliances. Providing load balancing and auto scaling for fleets of third-party appliances, Gateway Load Balancer is transparent to the source and destination of traffic. This capability makes it well suited for working with third-party appliances for security, network analytics, and other use cases.
- Classic Load Balancers - provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and the connection level. Classic Load Balancer is intended for applications that were built within the EC2-Classic network.

### *Load Balancers & Encryption*

Elastic Load Balancing simplifies the process of building secure web applications by terminating HTTPS and TLS traffic from clients at the load balancer. The load balancer performs the work of encrypting and decrypting the traffic, instead of requiring each EC2 instance to handle the work for TLS termination. When you configure a secure listener, you specify the cipher suites and protocol versions that are supported by your application, and a server certificate to install on your load balancer. You can use AWS Certificate Manager (ACM) or AWS Identity and Access Management (IAM) to manage your server certificates. Application Load Balancers support HTTPS listeners. Network Load Balancers support TLS listeners. Classic Load Balancers support both HTTPS and TLS listeners.

*What are Elastic Load Balancer Listeners and Targets/Target Groups?*

- **ELB Listener**
  - a process that checks for connection requests using the protocol and port number that you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets
- **ELB Target**
  - A destination for traffic based on established listener rules
- **ELB Target Group**
  - is a group of targets, for example, a group of EC2 instances
  - routes requests to one or more registered targets using the protocol and port number specified.
  - A target can be registered with multiple target groups.
  - Health checks can be configured on a per target group basis. The load balancer continually monitors the health of all targets registered with the target group that are in an Availability Zone enabled for the load balancer. The load balancer routes requests to the registered targets that are healthy

*Block Storage vs. Object Storage*

**What is Block Storage?**

Block storage is the oldest and simplest form of data storage. Block storage stores data in fixed-sized chunks called — you guessed it — 'blocks'. By itself, a block typically only houses a portion of the data. The application makes SCSI (Small Computer System Interface - set of standards for physically connecting and transferring data between computers and peripheral devices) calls to find the correct address of the blocks, then organizes them to form the complete file. Because the data is piecemeal, the address is the only identifying part of a block — there is no metadata associated with blocks. This structure leads to faster performance when the application and storage are local, but can lead to more latency when they are farther apart. The granular control that block storage offers makes it an ideal fit for applications that require high performance, such as transactional or database applications.

**What is Object Storage?**

Compared to block storage, object storage is much newer. With object storage, data is bundled with customizable metadata tags and a unique identifier to form objects. Objects are stored in a flat address space and there is no limit to the number of objects stored, making it much easier to scale out.

Each object has data, a key, and metadata.

- The object key uniquely identifies the object in the storage area.
- Object metadata is a set of name-value pairs. The metadata tags are a key advantage with object storage — they allow for much better identification and classification of data. You can think of objects as self-describing: They have descriptive labels assigned by the user or application that writes the object. Using a search application you can easily search for a specific object, even if the data itself is not easily searched (such as an image, or media clip, or data set).

For storing unstructured data, block storage vs object storage is no contest. Search capabilities and unlimited scale make object storage ideal for unstructured data, a classification that is currently expected to hit 44 zettabytes by 2020. Object storage is the only option that can effectively store this data at scale. Block storage has many uses within enterprises, but object storage is best equipped to handle the explosive growth of unstructured data. For a clearer side-by-side comparison of block storage vs object storage, take a look at the table below:

| | OBJECT STORAGE | BLOCK STORAGE |
|---|---|---|
| **PERFORMANCE** | Performs best for big content and high stream throughput | Strong performance with database and transactional data |
| **GEOGRAPHY** | Data can be stored across multiple regions | The greater the distance between storage and application, the higher the latency |
| **SCALABILITY** | Can scale infinitely to petabytes and beyond | Addressing requirements limit scalability |
| **ANALYTICS** | Customizable metadata allows data to be easily organized and retrieved | No metadata |

(https://cloudian.com/blog/object-storage-vs-block-storage/)

*Which Services are Global, Regional and Availability Zone Based?*

- IAM
  - Users, Groups, Roles, Accounts – **Global**
    - Same AWS accounts, users, groups and roles can be used in all regions
  - Key Pairs – **Global** or **Regional**
    - Amazon EC2 created key pairs are specific to the region
    - RSA key pair can be created and uploaded that can be used in all regions
- Virtual Private Cloud
  - VPC – **Regional**
    - VPC are created within a region
  - Subnet – **Availability Zone**
    - Subnet can span only a single Availability Zone
  - Security groups – **Regional**
    - A security group is tied to a region and can be assigned only to instances in the same region.
  - VPC Endpoints – **Regional**
    - You cannot create an endpoint between a VPC and an AWS service in a different region.
  - VPC Peering – **Regional**
    - VPC Peering can be performed across VPC in the same account of different AWS accounts. VPC Peering can span inter-region.
  - Elastic IP Address – Regional
    - Elastic IP address created within the region can be assigned to instances within the region only
- S3 – Global but Data is **Regional**
  - S3 buckets are created within the selected region
  - Objects stored are replicated across Availability Zones to provide high durability but are not cross region replicated unless done explicitly
- Route53 – **Global**
  - Route53 services are offered at AWS edge locations and are global
- DynamoDb – **Regional**
  - All data objects are stored within the same region and replicated across multiple Availability Zones in the same region
  - Data objects can be explicitly replicated across regions using cross-region replication
- WAF – **Global**
  - Web Application Firewall (WAF) services protects web applications from common web exploits are offered at AWS edge locations and are global
- CloudFront – **Global**
  - CloudFront is the global content delivery network (CDN) services are offered at AWS edge locations
- Storage Gateway – **Regional**
  - AWS Storage Gateway stores volume, snapshot, and tape data in the AWS region in which the gateway is activated

- EC2
  - Resource Identifiers – **Regional**
    - Each resource identifier, such as an AMI ID, instance ID, EBS volume ID, or EBS snapshot ID, is tied to its region and can be used only in the region where you created the resource.
  - Instances – **Availability Zone**
    - An instance is tied to the Availability Zones in which you launched it. However, note that its instance ID is tied to the region.
  - EBS Volumes – **Availability Zone**
    - Amazon EBS volume is tied to its Availability Zone and can be attached only to instances in the same Availability Zone.
  - EBS Snapshot – **Regional**
    - An EBS snapshot is tied to its region and can only be used to create volumes in the same region and has to be copied from One region to other if needed
  - AMIs – **Regional**
    - AMI provides templates to launch EC2 instances
    - AMI is tied to the Region where its files are located with Amazon S3. For using AMI in different regions, the AMI can be copied to other regions
  - Auto Scaling – **Regional**
    - Auto Scaling spans across multiple Availability Zones within the same region but cannot span across regions
  - Elastic Load Balancer – **Regional**
    - Elastic Load Balancer distributes traffic across instances in multiple Availability Zones in the same region
  - Cluster Placement Groups – **Availability Zone**
    - Cluster Placement groups can be span across Instances within the same Availability Zones

(https://jayendrapatil.com/aws-global-vs-regional-vs-az-resources)

*What is a Managed Service?*

- Managed Services –
    - is a cloud feature that you can use without having to take care of the underlying hardware's administration. For instance, in the Amazon ecosystem, you will find AWS Fargate, AWS Lambda, AWS Aurora, Amazon DynamoDB, and Elastic Beanstalk, among others. What do all those services have in common? The service provider, and not your organization, is responsible for getting deployments up and running on these platforms.
    - In managed services common activities are automated and implemented according to best practices, such as change requests, monitoring, patch management, security, and backup services. AWS Managed Services provide full-lifecycle services to provision, run, and support your infrastructure; and thus unburdens you from infrastructure operations so you can direct resources toward differentiating your business.
    - AWS Managed Services takes care of all of your patching and backup activities to help keep your resources current and secure. When updates or patches are released by OS vendors, AWS Managed Services applies them in a timely and consistent manner to minimize the impact on your business. Critical security patches are applied immediately, while others are applied based on the patch schedule you request.

### *What is Serverless?*

- It is a way to describe the services, practices, and strategies that enable you to build more agile applications so you can innovate and respond to change faster. With serverless computing, infrastructure management tasks like capacity provisioning and patching are handled by AWS, so you can focus on only writing code that serves your customers. Serverless services like AWS Lambda come with automatic scaling, built-in high availability, and a pay-for-value billing model. Lambda is an event-driven compute service that enables you to run code in response to events from over 150 natively-integrated AWS and SaaS sources - all without managing any servers.
- Benefits:
    - Move from idea to market, faster - By eliminating operational overhead, your teams can release quickly, get feedback, and iterate to get to market faster.
    - Lower your costs - With a pay-for-value billing model, you never pay for over-provisioning and your resource utilization is optimized on your behalf.
    - Adapt at scale - With technologies that automatically scale from zero to peak demands, you can adapt to customer needs faster than ever.
    - Build better applications, easier - Serverless applications have built-in service integrations, so you can focus on building your application instead of configuring it.
    - Is a way to describe the services, practices, and strategies that enable you to build more agile applications so you can innovate and respond to change faster. With serverless computing, infrastructure management tasks like capacity provisioning and patching are handled by AWS, so you can focus on only writing code that serves your customers. Serverless services (like AWS Lambda) come with automatic scaling, built-in high availability, and a pay-for-value billing model

### *Serverless Services List*

- AWS Lambda
- Amazon Fargate
- Amazon EventBridge
- AWS Step Functions
- Amazon SQS
- Amazon SNS
- Amazon API Gateway
- AWS AppSync
- Amazon S3
- Amazon DynamoDB
- Amazon RDS Proxy
- Amazon Aurora Serverless

### *What is the Difference between Managed Service and Serverless?*

- If a service or product is "Serverless", that means that it is also "Managed". But not all managed services are serverless; serverless is a special kind of managed service.
- What is it about serverless that makes it special? You completely stop thinking about the different kinds of "servers" in your architectures.
    - You stop thinking about asking the file "server" for something; you instead ask the data storage service to get it for you
    - You stop thinking about talking with the database "server"; you instead ask the query service to process your query.
    - You stop thinking about running your application "server" on a "server" instance; you instead have the service run your processing code whenever it's needed
- You can still have lots of great managed services that are not serverless, where you still have to choose the right size for your servers, however the cloud provider runs and manages those servers for you.

## *Amazon S3 Basics (Simple Storage Service)*

- S3 is a "global" service (available on every region however it is not truly global because while you can replicate your buckets/objects across regions for reliability & disaster recovery purposes, by default S3 objects sit only in one region though they are stored on multiple devices across multiple Availability Zones.
- S3 provides developers and IT teams with secure, durable, highly-scalable binary object storage.
- It has a simple, easy to use, web services interface to store and retrieve any amount of data from anywhere on the web.
- S3 is a safe Object-based storage for e.g. picture, text files, videos NOT databases, application or OS.
- Size of files can be from 0 – 5TB.
- Unlimited storage paid by the GB.
- Stored in buckets (folders in the cloud).
- When you upload a file to s3 you'll get a HTTP 200 code to show successful upload
- Is A Simple Key-value object store
    - Key – name of object
    - Value – data of file (sequence of bytes)
    - Version ID (important for versioning)
    - Metadata – data about what you're storing
- It's essentially a type of NoSQL database. Each bucket is a new "database", with keys being your "folder path" and values being the binary objects (files). It's presented like a file system and people tend to use it like one. Underneath, however, its not a file system at all and lacks many of the common traits of a file system.
- Security
    - Access Control Lists (ACLs) – (file level)
        - are used to define which AWS accounts or groups are granted access and the type of access. When a request is received against an S3 resource, the corresponding resource ACL is checked to verify that the requester has the necessary access permissions.
    - Bucket Policies - (bucket level)
        - is a resource-based AWS Identity andAccess Management (IAM) policy. You add a bucket policy to a bucket to grant other AWS accounts or IAM users access permissions for the bucket and the objects in it.
- have universal namespace e.g. url that can be accessed. Buckets need to be unique globally. For example, bucket url: s3-[region].amazonaws.com/[bucketName]
- Built for 99.99% availability
- Amazon guarantee 11 x 9s durability for S3 information
- Tiered Storage Available
- Encryption (encrypt your files at rest)
- S3 charged:
    - All according to Storage Tier Pricing
    - For Storage per GB
    - Per # of Requests
    - You pay for requests made against your S3 buckets and objects. You pay for all bandwidth into and out of Amazon S3, except for the following:
        - Data transferred in from the internet.
        - Data transferred out to an Amazon Elastic Compute Cloud (Amazon EC2) instance, when the instance is in the same AWS Region as the S3 bucket (including to a different account in the same AWS region).
        - Data transferred out to Amazon CloudFront (CloudFront).

    - For Data Transfer (transferring from one region to another)
    - For Transfer Acceleration - Enables fast, easy and secure transfers of files over long distances between your end users and an S3 bucket.
    - For taking advantage of Cloudfront's globally distributed edge locations.
    - For Cross Region Replication (CRR)
- Bucket names share a common name space. Their names must be unique.
- You view the buckets globally but you can have buckets in individual regions
- Contents uploaded to buckets are private by default

(http://kayleigholiver.com/aws-cloud-practitioner-s3/)

*What is S3 Replication?*

Replication enables automatic, asynchronous copying of objects across Amazon S3 buckets. Buckets that are configured for object replication can be owned by the same AWS account or by different accounts. There are two kinds of S3 replication:

- Cross Region Replication (CRR). When an item has been uploaded to a primary bucket is replicated to a secondary bucket in a different AWS Region.
- Same-Region replication (SRR) is used to copy objects across Amazon S3 buckets in the same AWS Region.

Requirements:

- Both source and destination buckets must have versioning enabled.
- The source bucket owner must have the source and destination AWS Regions enabled for their account. The destination bucket owner must have the destination Region-enabled for their account. For more information about enabling or disabling an AWS Region, see AWS Service Endpoints in the AWS General Reference.
- If the source bucket has S3 Object Lock enabled, the destination bucket must also have S3 Object Lock enabled
- Amazon S3 must have permissions to replicate objects from the source bucket to the destination bucket on your behalf.
- If the owner of the source bucket doesn't own the object in the bucket, the object owner must grant the bucket owner READ and READ_ACP permissions with the object access control list (ACL)

*S3 Tiers*

**S3 Standard**

- Built for 99.99% availability
- Amazon guarantee 11 x 9s durability for S3 information
- Stored redundantly across multiple devices in multiple facilities
- Designed to sustain the loss of 2 facilities concurrently

**S3 – IA (Infrequently Accessed)**

- Amazon guarantee 11 x 9s durability for S3 information
- For data accessed less frequently, but requires rapid access.
- Lower fee than S3, but you are charged a retrieval fee

**S3 One Zone – IA**

- Amazon guarantee 11 x 9s durability for S3 information
- Lower cost option for infrequently accessed data
- Only available in one availability zone

**S3 – Intelligent Tiering**

- Amazon guarantee 11 x 9s durability for S3 information
- Uses ML looking at your usage patterns
- Moves data to the most cost-effective access tier without performance impact or operational overhead (can move your data across the other 3 tiers)
- Available from only one AZ

**S3 Glacier**

- Amazon guarantee 11 x 9s durability for S3 information
- Low cost storage
- Used for archival only
- Comes in the models: Expedited, Standard or Bulk.
- Standard retrieval configurable from minutes to hours

**S3 Glacier Deep Archive**

- Amazon guarantee 11 x 9s durability for S3 information
- Lowest storage class
- Retrieval time of 12 hours

(http://kayleigholiver.com/aws-cloud-practitioner-s3/)

### *Creating a Website on S3*

- You can use bucket policies to make entire S3 buckets public (instead of individually updating the permissions on each object within the bucket) by enabling **"Edit public access settings"** to make everything in a bucket public by default
- You can use S3 to host only a **static** website e.g. .html. websites that require a database connection e.g. a WordPress site cannot be hosted on S3.
- S3 scales automatically to meet your demand. Useful when there will be a large number of requests e.g. movie previews.
- Recognise the url that a statically hosted website will use i.e. **http://sitename-website2019.s3-website-us-east-1.amazonaws.com**

(http://kayleigholiver.com/aws-cloud-practitioner-s3/)

### *How can you create an Amazon S3 bucket that cannot have any public objects due to compliance requirements?*

Enable the 'Block public access' option. This can be done in S3 Console, the CLI, the S3 APIs or from within CloudFormation templates. Additionally you have the option to enable/disable the following options:

- Block public access to buckets and objects granted through new access control lists (ACLs)
- Block public access to buckets and objects granted through any access control lists (ACLs)
- Block public access to buckets and objects granted through new public bucket policies
- Block public and cross-account access to buckets and objects through any public bucket policies

### *S3 Bucket Policies*

S3 Bucket Policies contain five key elements. Effect, Action, Resource, Condition and Prinipal:

- Effect – Use Allow or Deny to indicate whether the policy allows or denies access.
- Action – Include a list of actions that the policy allows or denies.
- Resource (Required in only some circumstances) – If you create an IAM permissions policy, you must specify a list of resources to which the actions apply. If you create a resource-based policy, this element is optional. If you do not include this element, then the resource to which the action applies is the resource to which the policy is attached.
- Condition (Optional) – Specify the circumstances under which the policy grants permission.
- Principal is used by Resource Policies (SNS, S3 Buckets, SQS, etc) to define who the policy applies to. In most cases the Principal is the root user of a specific AWS account. That AWS account can then delegate permission (via IAM) to users or roles. That means when you trust the root of another AWS Account, you're trusting all the IAM or federated users in that account.

### *Route 53*

- Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost effective way to route end users to Internet applications by translating names like www.example.com into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other. Amazon Route 53 is fully compliant with IPv6 as well.
- 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. You can use Amazon Route 53 to configure DNS health checks to route traffic to healthy endpoints or to independently monitor the health of your application and its endpoints.
- Amazon Route 53 Traffic Flow makes it easy for you to manage traffic globally through a variety of routing types, including Latency Based Routing, Geo DNS, Geoproximity, and Weighted Round Robin—all of which can be combined with DNS Failover in order to enable a variety of low-latency, fault-tolerant architectures. Using Amazon Route 53 Traffic Flow's simple visual editor, you can easily manage how your end-users are routed to your application's endpoints—whether in a single AWS region or distributed around the globe.
- Amazon Route 53 also offers Domain Name Registration – you can purchase and manage domain names such as example.com and Amazon Route 53 will automatically configure DNS settings for your domains.
- Inbound query capability is provided by Route 53 Resolver Endpoints, allowing DNS queries that originate on-premises to resolve AWS hosted domains.

### *Amazon GuardDuty*

- Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behavior to protect your AWS accounts, workloads, and data stored in Amazon S3. With the cloud, the collection and aggregation of account and network activities is simplified, but it can be time consuming for security teams to continuously analyze event log data for potential threats. With GuardDuty, you now have an intelligent and cost-effective option for continuous threat detection in AWS.
- The service uses machine learning, anomaly detection, and integrated threat intelligence to identify and prioritize potential threats. GuardDuty analyzes tens of billions of events across multiple AWS data sources, such as AWS CloudTrail event logs, Amazon VPC Flow Logs, and DNS logs.
- With a few clicks in the AWS Management Console, GuardDuty can be enabled with no software or hardware to deploy or maintain. By integrating with Amazon CloudWatch Events, GuardDuty alerts are actionable, easy to aggregate across multiple accounts, and straightforward to push into existing event management and workflow systems.
- Threats can include issues like escalations of privileges, uses of exposed credentials, or communication with malicious IPs, URLs, or domains. For example, GuardDuty can detect compromised EC2 instances that serve malware, unauthorized infrastructure deployments such as EC2 instances deployed in a Region that has never been used, or unusual API calls like a password policy change to reduce password strength.
- GuardDuty informs you of the status of your AWS environment by producing security findings that you can view in the GuardDuty console or through Amazon CloudWatch events.

*(Amazon) CloudFront*

- When your web traffic is geo-dispersed, it's not always feasible and certainly not cost effective to replicate your entire infrastructure across the globe. A content delivery network (CDN) provides you the ability to utilize its global network of edge locations to deliver a cached copy of web content such as videos, webpages, images and so on to your customers. To reduce response time, the CDN utilizes the nearest edge location to the customer or originating request location in order to reduce the response time. Throughput is dramatically increased given that the web assets are delivered from cache. For dynamic data, many CDNs can be configured to retrieve data from the origin servers.
- What is CloudFront Regional Edge Cache? CloudFront delivers your content through a worldwide network of data centers called edge locations. The regional edge caches are located between your origin web server and the global edge locations that serve content directly to your viewers. This helps improve performance for your viewers while lowering the operational burden and cost of scaling your origin resources. It is essentially a fast 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. CloudFront is integrated with AWS – both physical locations that are directly connected to the AWS global infrastructure, as well as other AWS services.
- CloudFront works seamlessly with services including AWS Shield for DDoS mitigation, Amazon S3, Elastic Load Balancing or Amazon EC2 as origins for your applications, and Lambda@Edge to run custom code closer to customers' users and to customize the user experience.
- If you use AWS origins such as Amazon S3, Amazon EC2 or Elastic Load Balancing, you don't pay for any data transferred between these services and CloudFront. Amazon's CDN offers a simple, pay-as-you-go pricing model with no upfront fees or required long-term contracts, and support for the CDN is included in your existing AWS Support subscription.
- You also have to use S3 in order to make use of CloudFront. CloudFront doesn't work with EBS and EFS.

*(Amazon) RDS (Relational Database)*

RDS makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. It frees you to focus on your applications so you can give them the fast performance, high availability, security and compatibility they need.

Amazon RDS is available on several database instance types - optimized for memory, performance or I/O - and provides you with six familiar database engines to choose from, including Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle Database, and SQL Server. You can use the AWS Database Migration Service to easily migrate or replicate your existing databases to Amazon RDS.

Easy storage scaling - As your storage requirements grow, you can also provision additional storage. The Amazon Aurora engine will automatically grow the size of your database volume as your database storage needs grow, up to a maximum of 64 TB or a maximum you define. The MySQL, MariaDB, Oracle, and PostgreSQL engines allow you to scale up to 64 TB of storage and SQL Server supports up to 16 TB. Storage scaling is on-the-fly with zero downtime.

Database Snapshots - are user-initiated backups of your instance stored in Amazon S3 that are kept until you explicitly delete them. You can create a new instance from a database snapshots whenever you desire. Although database snapshots serve operationally as full backups, you are billed only for incremental storage use.

Automated backups - Amazon RDS creates and saves automated backups of your DB instance during the backup window of your DB instance. RDS creates a storage volume snapshot of your DB instance, backing up the entire DB instance and not just individual databases. RDS saves the automated backups of your DB instance according to the backup retention period that you specify. If necessary, you can recover your database to any point in time during the backup retention period.

Online transaction processing (OLTP) captures, stores, and processes data from transactions in real time. Online analytical processing (OLAP) uses complex queries to analyze aggregated historical data from OLTP systems

**Multi-AZ RDS Deployment**

When you provision a Multi-AZ DB Instance, Amazon RDS automatically creates a primary DB Instance and synchronously replicates the data to a standby instance in a different Availability Zone (AZ). Each AZ runs on its own physically distinct, independent infrastructure, and is engineered to be highly reliable. In case of an infrastructure failure, Amazon RDS performs an automatic failover to the standby (or to a read replica in the case of Amazon Aurora), so that you can resume database operations as soon as the failover is complete. Since the endpoint for your DB Instance remains the same after a failover, your application can resume database operation without the need for manual administrative intervention.

*RDS Vertical vs Horizontal Scaling*

- **Push-button vertical scaling** - You can scale vertically to address the growing demands of an application that uses a roughly equal number of reads and writes. You can scale the compute and memory resources powering your deployment up or down, up to a maximum of 32 vCPUs and 244 GiB of RAM. Compute scaling operations typically complete in a few minutes. To handle a higher load in your database, you can vertically scale up your master database with a simple push of a button. There are currently over 18 instance sizes that you can choose from when resizing your RDS MySQL, PostgreSQL, MariaDB, Oracle, or Microsoft SQL Server instance. In addition to scaling your master database vertically, you can also improve the performance of a read-heavy database by using read replicas to horizontally scale your database.
- **Read Replicas (horizontal scaling)** make it easy to elastically scale out beyond the capacity constraints of a single DB instance for read-heavy database workloads. You can create one or more replicas of a given source DB instance and serve high-volume application read traffic from multiple copies of your data, thereby increasing aggregate read throughput. RDS MySQL, PostgreSQL, and MariaDB can have up to 5 read replicas. Amazon Aurora can have up to 15 read replicas. Read replicas allow you to create read-only copies that are synchronized with your master database. You can also place your read replica in a different AWS Region closer to your users for better performance. Also, you can use read replicas to increase the availability of your database by promoting a read replica to a master for faster recovery in the event of a disaster. However, read replicas are not a replacement for the high availability and automatic failover capabilities that Multi-AZ provides. Currently, RDS read replicas support transparent load balancing of queries or connections.
    - Each replica has a unique Domain Name Service (DNS) endpoint so that an application can implement load balancing by connecting to the replica endpoint.

### *(Amazon) Aurora*

- is a MySQL and PostgreSQL-compatible relational database built for the cloud, that combines the performance and availability of traditional enterprise databases with the simplicity and cost-effectiveness of open source databases.
- Amazon Aurora is up to five times faster than standard MySQL databases and three times faster than standard PostgreSQL databases. It provides the security, availability, and reliability of commercial databases at 1/10th the cost. Amazon Aurora is fully managed by Amazon Relational Database Service (RDS), which automates time-consuming administration tasks like hardware provisioning, database setup, patching, and backups.
- Amazon Aurora features a distributed, fault-tolerant, self-healing storage system that auto-scales up to 128TB per database instance. It delivers high performance and availability with up to 15 low-latency read replicas, point-in-time recovery, continuous backup to Amazon S3, and replication across three Availability Zones (AZs).

### *(Amazon) DynamoDB*

Fast and flexible NoSQL database service for any scale. A key-value and document database that delivers single-digit millisecond performance at any scale. It's a fully managed, multiregion, multimaster, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications. DynamoDB can handle more than 10 trillion requests per day and can support peaks of more than 20 million requests per second.

### *(Amazon) Elastic File System (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 the throughput, IOPS, and low latency needed for **Linux workloads**. Throughput and IOPS scale as a file system grows and can burst to higher throughput levels for short periods of time to support the unpredictable performance needs of file workloads. For the most demanding workloads, Amazon EFS can support performance over 10 GB/sec and up to 500,000 IOPS.
- With Elastic File System (EFS), you can share data between multiple EC2 instances and your data is replicated between multiple
- EFS file system can be used by multiple EC2 instances from different data centers in parallel. Additionally, the data of the EFS file system is replicated among multiple data centers/availability zones (AZs)

### *(AWS) Lambda*

- AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

- With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.

- Continuous scaling - AWS Lambda automatically scales your application by running code in response to each trigger. Your code runs in parallel and processes each trigger individually, scaling precisely with the size of the workload.

- With AWS Lambda, you pay only for what you use. You are charged based on the number of requests for your functions and the duration, the time it takes for your code to execute.
    - Lambda counts a request each time it starts executing in response to an event notification or invoke call, including test invokes from the console.
    - Duration is calculated from the time your code begins executing until it returns or otherwise terminates, rounded up to the nearest 100ms*. The price depends on the amount of memory you allocate to your function. In the AWS Lambda resource model, you choose the amount of memory you want for your function, and are allocated proportional CPU power and other resources. An increase in memory size triggers an equivalent increase in CPU available to your function. To learn more, see the Function Configuration documentation.
    - The AWS Lambda free usage tier includes 1M free requests per month and 400,000 GB-seconds of compute time per month.
    - AWS Lambda participates in Compute Savings Plans, a flexible pricing model that offers low prices on EC2, Fargate, and Lambda usage, in exchange for a commitment to a consistent amount of usage (measured in $/hour) for a 1 or 3 year term. With Compute Savings Plans you can save up to 17% on AWS Lambda. Savings apply to Duration, Provisioned Concurrency, and Duration (Provisioned Concurrency).

### *(AWS) CloudFormation*

- Speed up cloud provisioning with infrastructure as code. Gives you an easy way to model a collection of related AWS and third-party resources, provision them quickly and consistently, and manage them throughout their lifecycles, by treating infrastructure as code (IaC). A CloudFormation template describes your desired resources and their dependencies so you can launch and configure them together as a stack. You can use a template to create, update, and delete an entire stack as a single unit, as often as you need to, instead of managing resources individually. You can manage and provision stacks across multiple AWS accounts and AWS Regions. The CloudFormation template acts as a *"single source of truth"* for an AWS cloud environment.

### *(AWS) Config*

- AWS Config is a service that enables you to assess, audit, and evaluate the configurations of your AWS resources.
- Config continuously monitors and records your AWS resource configurations and allows you to automate the evaluation of recorded configurations against desired configurations.
- With Config, you can review changes in configurations and relationships between AWS resources, dive into detailed resource configuration histories, and determine your overall compliance against the configurations specified in your internal guidelines. This enables you to simplify compliance auditing, security analysis, change management, and operational troubleshooting.
- (AWS) Config continuously monitors and records your AWS resource configurations. It can detect drift and trigger (AWS) Systems Manager Automation to fix it and raise alarms.

- *(Amazon) CloudWatch –*
    - Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), and IT managers.
    - 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.
    - You can use CloudWatch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, **take automated actions**, troubleshoot issues, and discover insights to keep your applications running smoothly.
    - 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.
        - With Basic monitoring you get data on your Cloudwatch metrics every 5 minutes. Enabling detailed monitoring, you will get the data every one minute.

- (Amazon) CloudWatch Events
    - Amazon CloudWatch Events delivers a near real-time stream of system events that describe changes in Amazon Web Services (AWS) resources. Using simple rules that you can quickly set up, you can match events and route them to one or more target functions or streams. CloudWatch Events becomes aware of operational changes as they occur. CloudWatch Events responds to these operational changes and takes corrective action as necessary, by sending messages to respond to the environment, activating functions, making changes, and capturing state information.
    - You can also use CloudWatch Events to schedule automated actions that self-trigger at certain times using cron (The origin of the name cron is from the Greek word for time, χρόνος (chronos)) or rate expressions.
    - The following services are used in conjunction with CloudWatch Events: AWS CloudTrail, AWS CloudFormation, AWS Config, AWS Identity and Access Management (IAM), Amazon Kinesis Data Streams and AWS Lambda

- (Amazon) CloudWatch Logs
    - You can use Amazon CloudWatch Logs to monitor, store, and access your log files from Amazon Elastic Compute Cloud (Amazon EC2) instances, AWS CloudTrail, Route 53, and other sources.
    - CloudWatch Logs enables you to centralize the logs from all of your systems, applications, and AWS services that you use, in a single, highly scalable service. Log data can be stored and accessed indefinitely in highly durable, low-cost storage so you don't have to worry about filling up hard drives
    - You can then easily view them, monitor your logs, in NEAR real-time for specific error codes or patterns, filter them based on specific fields, or archive them securely for future analysis. CloudWatch Logs enables you to see all of your logs, regardless of their source, as a single and consistent flow of events ordered by time, and you can query them and sort them based on other dimensions, group them by specific fields, create custom computations with a powerful query language, and visualize log data in dashboards. You can also view the original log data to see the source of the problem
    - CloudWatch Logs Additional Features:
        - Log Retention – By default, logs are kept indefinitely and never expire. You can adjust the retention policy for each log group, keeping the indefinite retention, or choosing a retention period between 10 years and one day.
        - Archive Log Data
        - Log Route 53 DNS Queries

- Amazon CloudWatch Alarms
    - allow you to set a threshold on metrics and trigger an action. You can create high-resolution alarms, set a percentile as the statistic, and either specify an action or ignore as appropriate. For example, you can create alarms on Amazon EC2 metrics, set notifications, and take one or more actions to detect and shut down unused or underutilized instances. Real-time alarming on metrics and events enables you to minimize downtime and potential business impact.
    - Alarms invoke actions for sustained state changes only. CloudWatch alarms do not invoke actions simply because they are in a particular state. The state must have changed and been maintained for a specified number of periods.
    - A composite alarm includes a rule expression that takes into account the alarm states of other alarms that you have created. The composite alarm goes into ALARM state only if all conditions of the rule are met. The alarms specified in a composite alarm's rule expression can include metric alarms and other composite alarms. Using composite alarms can reduce alarm noise. You can create multiple metric alarms, and also create a composite alarm and set up alerts only for the composite alarm. For example, a composite might go into ALARM state only when all of the underlying metric alarms are in ALARM state.

### *(AWS) CloudTrail*

- Track user activity and API usage. Helps you enable governance, compliance, and operational and risk auditing of your AWS account. Actions taken by a user, role, or an AWS service are recorded as events in CloudTrail. Events include actions taken in the AWS Management Console, AWS Command Line Interface, and AWS SDKs and APIs.

### *(AWS) CodeStar Services -*

- enables you to quickly develop, build, and deploy applications on AWS.
- provides a unified user interface, enabling you to easily manage your software development activities in one place. You can set up your entire continuous delivery toolchain in minutes, allowing you to start releasing code faster. Makes it easy for your whole team to work together securely, allowing you to easily manage access and add owners, contributors, and viewers to your projects.
- Each AWS CodeStar project comes with a project management dashboard, including an integrated issue tracking capability powered by Atlassian JIRA Software. With the AWS CodeStar project dashboard, you can easily track progress across your entire software development process, from your backlog of work items to teams' recent code deployments.
- Related AWS Code services are:
    - CodeCommit - A secure and scalable source/version control service supporting Git workflows
    - CodePipeline - A service for fast and reliable continuous integration (CI) and continuous delivery (CD)
    - CodeBuild - A scalable service to compile, test, and package source code
    - CodeDeploy - A service to automate code deployments anywhere

*(AWS) OpsWorks*

- AWS OpsWorks is a configuration management service that provides managed instances of Chef and Puppet.
- Chef and Puppet are automation platforms that allow you to use code to automate the configurations of your servers. OpsWorks lets you use Chef and Puppet to automate how servers are configured, deployed, and managed across your Amazon EC2 instances or on-premises compute environments.
- You model your application as a stack, consisting of various layers. These layers are like blueprints detailing how to setup and configure a set of EC2 instances and related resources. There are prebuilt layers for common components. “Chef recipes” detail your layout and configuration. Automatically and manually scalable. Essentially opsworks automates your infrastructure deployment.
- OpsWorks comes at no additional cost, you pay only for the resources and services you use to run your applications.
- OpsWorks has three offerings, AWS Opsworks for Chef Automate, AWS OpsWorks for Puppet Enterprise, and AWS OpsWorks Stacks.
    - **AWS OpsWorks for Chef Automate** provides a fully managed Chef Automate server and suite of automation tools that give you workflow automation for continuous deployment, automated testing for compliance and security, and a user interface that gives you visibility into your nodes and their status. The Chef Automate platform gives you full stack automation by handling operational tasks such as software and operating system configurations, continuous compliance, package installations, database setups, and more. The Chef server centrally stores your configuration tasks and provides them to each node in your compute environment at any scale, from a few nodes to thousands of nodes. OpsWorks for Chef Automate is completely compatible with tooling and cookbooks from the Chef community and automatically registers new nodes with your Chef server.
    - **AWS OpsWorks for Puppet Enterprise** is a fully managed configuration management service that hosts Puppet Enterprise, a set of automation tools from Puppet for infrastructure and application management. OpsWorks also maintains your Puppet master server by automatically patching, updating, and backing up your server. OpsWorks eliminates the need to operate your own configuration management systems or worry about maintaining its infrastructure. OpsWorks gives you access to all of the Puppet Enterprise features, which you manage through the Puppet console. It also works seamlessly with your existing Puppet code.
    - **AWS OpsWorks Stacks** lets you manage applications and servers on AWS and on-premises. With OpsWorks Stacks, you can model your application as a stack containing different layers, such as load balancing, database, and application server. You can deploy and configure Amazon EC2 instances in each layer or connect other resources such as Amazon RDS databases. OpsWorks Stacks lets you set automatic scaling for your servers based on preset schedules or in response to changing traffic levels, and it uses lifecycle hooks to orchestrate changes as your environment scales. You run Chef recipes using Chef Solo, allowing you to automate tasks such as installing packages and programming languages or frameworks, configuring software, and more.

### *AWS X-Ray*

- Helps developers analyse and debug production, distributed applications, such as those built using a microservices architecture. With X-Ray, you can understand how your application and its underlying services are performing to identify and troubleshoot the root cause of performance issues and errors.
- provides an end-to-end view of requests as they travel through your application, and shows a map of your application's underlying components. You can use X-Ray to analyse both applications in development and in production, from simple three-tier applications to complex microservices applications consisting of thousands of services.

### *(AWS) Fargate*

- AWS Fargate is a serverless compute engine for containers that works with both Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS).
- Fargate makes it easy for you to focus on building your applications. Fargate removes the need to provision and manage servers, lets you specify and pay for resources per application, and improves security through application isolation by design. Fargate allocates the right amount of compute power, eliminating the need to choose instances and scale cluster capacity. You only pay for the resources required to run your containers, so there is no over-provisioning and paying for additional servers.
- Fargate runs each task or pod in its own kernel providing the tasks and pods their own isolated compute environment. This enables your application to have workload isolation and improved security by design.
- Containers are used in PaaS where customer is responsible for app and data and the rest is taken care of by the cloud provider.

*(Amazon) ElastiCache*

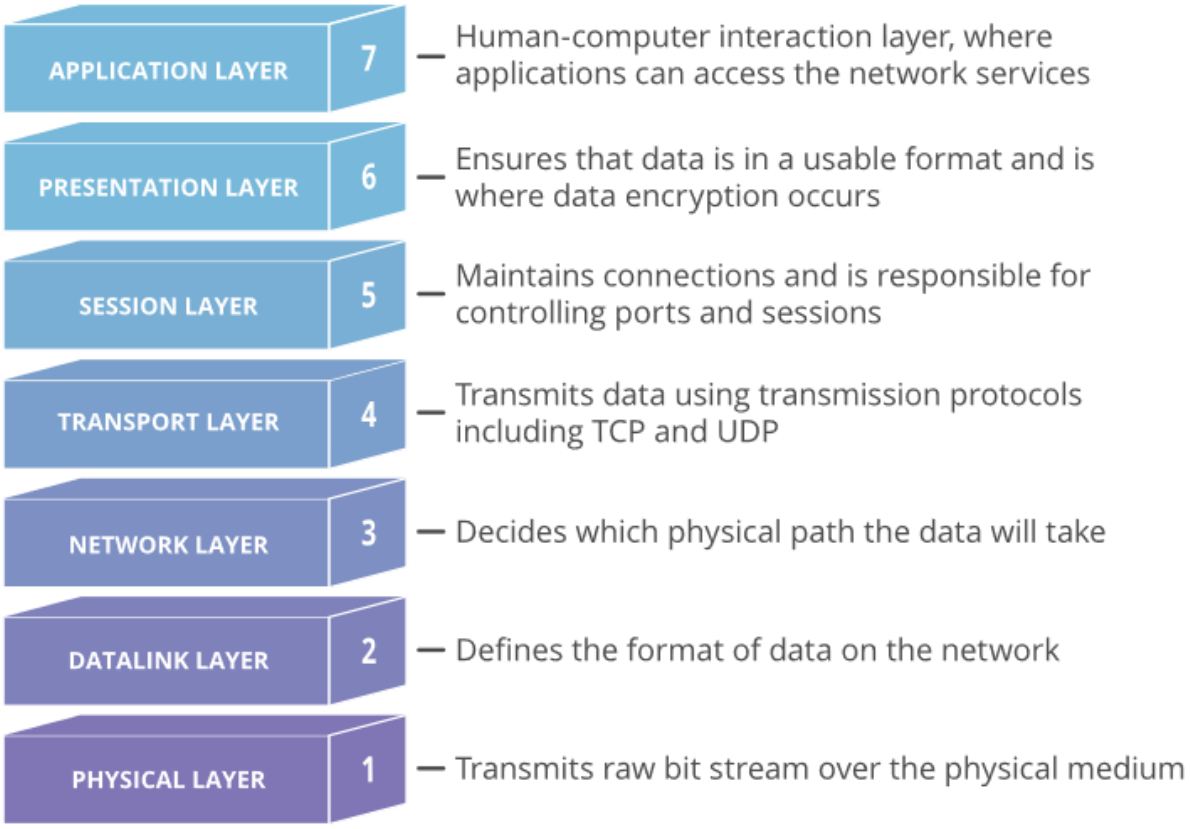
- Fully managed in-memory data store, compatible with Redis or Memcached. Power real-time applications with sub-millisecond latency. Amazon ElastiCache allows you to seamlessly set up, run, and scale popular open-source compatible in-memory data stores in the cloud.
- Elasticache does not run at edge locations, instead it simply improves the performance of web applications by allowing you to retrieve information from fast, managed, in-memory data stores, instead of relying entirely on slower disk-based databases.
- Database query results caching, persistent session caching, Gaming, Geospatial Services, Real-Time Analytics, Queuing and full-page caching are all popular examples of caching. 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.
- Memcached
    - is an easy-to-use, high-performance, in-memory data store. It offers a mature, scalable, open-source solution for delivering sub-millisecond response times making it useful as a cache or session store. Memcached is a popular choice for powering real-time applications in Web, Mobile Apps, Gaming, Ad-Tech, and E-Commerce. Unlike databases that store data on disk or SSDs, Memcached keeps its data in memory. By eliminating the need to access disks, in-memory key-value stores such as Memcached avoid seek time delays and can access data in microseconds. Memcached is also distributed, meaning that it is easy to scale out by adding new nodes. And since Memcached is multithreaded, you can easily scale up compute capacity. As a result of its speed and scalability as well as its simple design, efficient memory management, and API support for most popular languages Memcached is a popular choice for high-performance, large-scale caching use cases.
- Redis
    - Redis, which stands for Remote Dictionary Server, is a fast, open-source, in-memory key-value data store for use as a database, cache, message broker, and queue. Redis now delivers sub-millisecond response times enabling millions of requests per second for real-time applications. All Redis data resides in-memory, in contrast to databases that store data on disk or SSDs. By eliminating the need to access disks, in-memory data stores such as Redis avoid seek time delays and can access data in microseconds. Redis features versatile data structures, high availability, geospatial, Lua scripting, transactions, on-disk persistence, and cluster support making it simpler to build real-time internet scale apps.
- Redis vs Memcached
    - Both are in-memory, open-source data stores. Memcached, a high-performance distributed memory cache service, is designed for simplicity while Redis offers a richer set of features that make it more effective for a wide range of use cases. They work with relational or key-value databases to improve performance, such as MySQL, Postgres, Aurora, Oracle, SQL Server, DynamoDB, and more…

*(Amazon) Rekognition*

- image and video analysis for your applications using highly scalable, deep learning technology that requires no machine learning expertise to use. With Amazon Rekognition, you can identify objects, people, text, scenes, and activities in images and videos, as well as detect any inappropriate content.
- Amazon Rekognition also provides highly accurate facial analysis and facial search capabilities that you can use to detect, analyze, and compare faces for a wide variety of user verification, people counting, and public safety use cases.

## Networks

*The 7-layer OSI (Open Systems Interconnection) Model*




(https://www.cloudflare.com/en-gb/learning/ddos/glossary/open-systems-interconnection-model-osi/)

*Kinds of Cyber Attacks Terminology*

- DoS (Denial-of-Service)
    - A DoS attack is an attack meant to shut down a machine or network, making it inaccessible to its intended users. DoS attacks accomplish this by flooding the target with traffic, or sending it information that triggers a crash
- DDoS (Distributed Denial-of-Service)
    - occurs when multiple systems flood the bandwidth or resources of a targeted system, usually one or more web servers.[14] A DDoS attack uses more than one unique IP address or machines, often from thousands of hosts infected with malware
- HTTP Flood
    - a type of volumetric DDoS attack designed to overwhelm a targeted server with HTTP requests. The request can be either “GET” or “POST”. The aim of the attack is when to compel the server to allocate as many resources as possible to serving the attack, thus denying legitimate users access to the server's resources and thus impact the operation of web servers and any applications they are running.
- DNS Query Flood
    - DNS flood is a type of Distributed Denial of Service (DDoS) attack in which the attacker targets one or more Domain Name System (DNS) servers belonging to a given zone, attempting to hamper legitimate resolution of resource records of that zone and its sub-zones.
    - DNS servers are the “roadmap” of the Internet, helping requestors find the servers they seek. A DNS zone is a distinct portion of the domain name space in the Domain Name System (DNS). For each zone, administrative responsibility is delegated to a single server cluster.
- Reflection attacks
    - The attacker spoofs the victim’s IP address and sends a request for information via UDP to servers known to respond to that type of request. The server answers the request and sends the response to the victim’s IP address. From the servers’ perspective, it was the victim who sent the original request. All the data from those servers pile up, congesting the target’s Internet connectivity. With the maximized bandwidth, normal traffic cannot be serviced and clients cannot connect.
- Amplification attacks
    - is a reflection attack where the reply is larger than the the request. It is any attack where an attacker is able to use an amplification factor to multiply its power, amplification attacks are "asymmetric", meaning that a relatively small number or low level of resources is required by an attacker to cause a significantly greater number or higher level of target resources to malfunction or fail.
    - DNS amplification attacks, for example, use DNS requests with a spoofed source address as the target, the DNS request is not sent back to the computer that issued the request, but instead to the victim. Through this method the attacker uses a modest number of machines with little bandwidth to send fairly substantial attacks.
- IP address spoofing
    - is the creation of Internet Protocol (IP) packets with a false source IP address, for the purpose of impersonating another computing system
- SYN (synchronise) floods

    - is a form of denial-of-service attack in which an attacker rapidly initiates a connection to a server without finalizing the connection. The server has to spend resources waiting for half-opened connections, which can consume enough resources to make the system unresponsive to legitimate traffic. Also known as a half-open attack
- UDP (User Datagram Protocol) floods
    - is a type of Denial of Service (DoS) attack in which the attacker overwhelms random ports on the targeted host with IP packets containing UDP packets. The receiving host checks for applications associated with these datagrams and—finding none—sends back a "Destination Unreachable" packet. As more and more UDP packets are received and answered, the system becomes overwhelmed and unresponsive to other clients.
    - In the framework of a UDP flood attack, the attacker may also spoof the IP address of the packets, both to make sure that the return "Destination Unreachable" packets don't reach their host, and to anonymize the attack.
- Packet Sniffing
    - is a process of monitoring and capturing all data packets passing through given network.
        - Sniffers can be used legitimately by network/system administrators to monitor and troubleshoot network traffic.
        - Attackers use sniffers to capture data packets containing sensitive information such as password, account information etc. Sniffers can be hardware or software installed in the system. By placing a packet sniffer on a network in promiscuous mode, a malicious intruder can capture and analyse all of the network traffic.

### *VPC Flow Logs*

- VPC Flow Logs is a feature that enables you to capture information about the IP traffic going to and from network interfaces in your VPC. Flow log data can be published to Amazon CloudWatch Logs or Amazon S3. After you've created a flow log, you can retrieve and view its data in the chosen destination.
- Flow logs can help you with a number of tasks, such as:
    - Diagnosing overly restrictive security group rule
    - Monitoring the traffic that is reaching your instance
    - Determining the direction of the traffic to and from the network interfaces
- Flow log data is collected outside of the path of your network traffic, and therefore does not affect network throughput or latency. You can create or delete flow logs without any risk of impact to network performance.

### *AWS Global Accelerator*

- If local and global traffic to your application's single Region is left on the public internet, it can be negatively impacted by internet congestion and local outages. AWS Global Accelerator is a networking service that sends your user's traffic through Amazon Web Service's global network infrastructure, through 80+ global edge locations, then directed to your application origins, improving your internet user performance by up to 60%. When the internet is congested, Global Accelerator's automatic routing optimizations will help keep your packet loss, jitter, and latency consistently low.
- With Global Accelerator, you are provided two global static customer facing IPs to simplify traffic management. On the back end, add or remove your AWS application origins, such as Network Load Balancers, Application Load Balancers, Elastic IPs, and EC2 Instances, without making user facing changes.
- To mitigate endpoint failure Global Accelerator continually monitors the health of your application endpoints and redirects traffic to healthy endpoints, failover between application origins happens automatically and in less than 30 seconds.
- It can be used regardless of how many AWS Regions you are deployed in.

*How Do VPCs Work?*

A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, in a virtual network that you define. You have complete control over your virtual networking environment, including **selection of your own IP address range, creation of subnets, associate security groups, modifying access control lists and configuration of route tables and network gateways**.

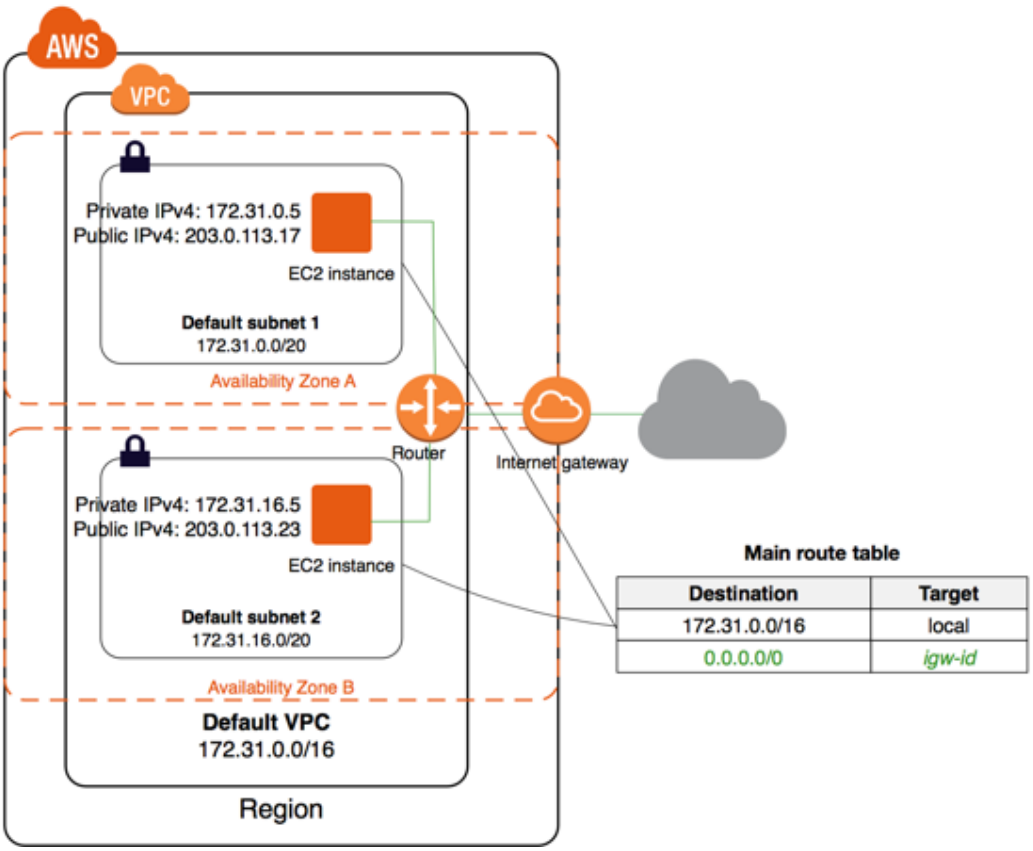
A **subnet** is a range of IP addresses in your VPC. You can launch AWS resources into a specified subnet. Use a public subnet for resources that must be connected to the internet, and a private subnet for resources that won't be connected to the internet.

You can easily customize the network configuration for your Amazon VPC. For example, you can create a public-facing subnet for your web servers that has access to the Internet, and place your backend systems such as databases or application servers in a private-facing subnet with no Internet access.

Additionally, with AWS, you can choose how network routing is delivered between Amazon VPC and your networks, leveraging either AWS or user-managed network equipment and routes.

To protect the AWS resources in each subnet, you can use multiple layers of security, including security groups and network access control lists (ACL).

You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications. To assign IPv6 address you must associate an IPv6 CIDR block with your VPC.




(https://docs.aws.amazon.com/vpc/latest/userguide/how-it-works.html)

*How to VPCs Access the Internet?*

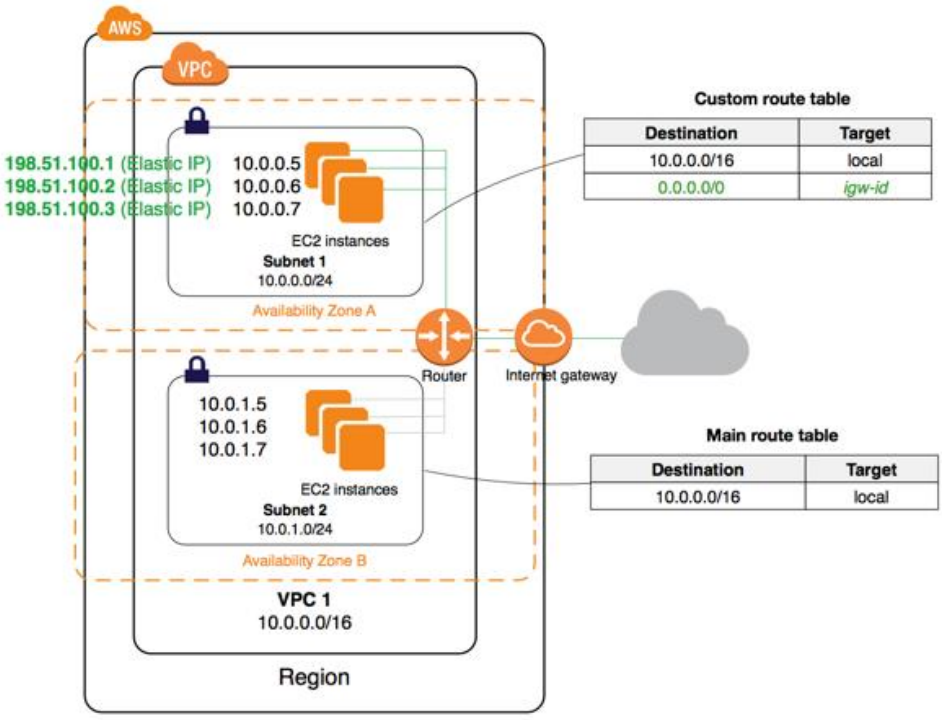
You control how the instances that you launch into a VPC access resources outside the VPC.

Your default VPC includes an **internet gateway**. An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the internet.

Each default subnet is a public subnet. Each instance that you launch into a default subnet has a private IPv4 address and a public IPv4 address. These instances can communicate with the internet through the internet gateway.

By default, each instance that you launch into a non-default subnet has a private IPv4address, but no public IPv4 address, unless you specifically assign one at launch, or you modify the subnet's public IP address attribute. These instances can communicate with each other, but can't access the internet.

You can enable internet access for an instance launched into a non-default subnet by attaching an internet gateway to its VPC (if its VPC is not a default VPC) and associating an Elastic IP address with the instance.




Alternatively, to allow an instance in your VPC to initiate outbound connections to the internet but prevent unsolicited inbound connections from the internet, you can use a network address translation (NAT) device for IPv4 traffic. NAT maps multiple private IPv4 addresses to a single public IPv4 address. A NAT device has an Elastic IP address and is connected to the internet through an internet gateway. You can connect an instance in a private subnet to the internet through the NAT device, which routes traffic from the instance to the internet gateway, and routes any responses to the instance.
(https://docs.aws.amazon.com/vpc/latest/userguide/how-it-works.html)

### *What is VPC Peering?*

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. Instances in either VPC can communicate with each other as if they are within the same network. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. The VPCs can be in different regions (also known as an inter-region VPC peering connection).

### *What is a Transit Gateway?*

AWS Transit Gateway connects VPCs and on-premises networks through a central hub using a hub-and-spoke (star) connection model. 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 abstracts away the complexity of maintaining VPN connections with hundreds of VPCs.

As you expand globally, inter-Region peering connects AWS Transit Gateways together using the AWS global network. Your data is automatically encrypted, and never travels over the public internet. And, because of its central position, AWS Transit Gateway Network Manager has a unique view over your entire network, even connecting to Software-Defined Wide Area Network (SD-WAN) devices.

Max limit is 125 peering connections per VPC currently.

*What is AWS VPN?*

AWS Virtual Private Network (VPN) solutions establish secure connections via the public internet between your on-premises networks, remote offices, client devices, and the AWS global network. You can connect your Amazon VPC to remote networks and users using the following VPN connectivity options:

- AWS Site-to-Site VPN:  creates encrypted tunnels between your network and your Amazon Virtual Private Clouds. A VPN Connection utilizes IPSec to establish encrypted network connectivity between your intranet and Amazon VPC.
    - On the AWS side of the Site-to-Site VPN connection, a **virtual private gateway** or **transit gateway** provides two VPN endpoints (tunnels) for automatic failover.
    - You configure your **customer gateway** device on the remote side of the Site-to-Site VPN connection.
- AWS Client VPN: a managed client-based VPN service that enables you to securely access your AWS resources or your on-premises network. With AWS Client VPN, you configure an endpoint to which your users can connect to establish a secure TLS VPN session. This enables clients to access resources in AWS or a non-premises from any location using an Open VPN-based VPN client.
- AWS VPN CloudHub: If you have more than one remote network (for example, multiple branch offices), you can create multiple AWS Site-to-Site VPN connections via your virtual private gateway to enable communication between these networks
- Third party software VPN appliance: You can create a VPN connection to your remote network by using an Amazon EC2 instance in your VPC that's running a third party software VPN appliance. AWS does not provide or maintain third party software VPN appliances; however, you can choose from a range of products provided by partners and open source communities. You can find third party software VPN appliances on the AWS Marketplace.

Together, they deliver a highly-available, managed, and elastic cloud VPN solution to protect your network traffic.

## *What is AWS 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. Using AWS Direct Connect, you can establish private connectivity between AWS and your datacentre, office, or colocation environment, which in many cases can reduce your network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based connections.

AWS Direct Connect lets you establish a dedicated network connection between your network and one of the AWS Direct Connect locations. Using industry standard 802.1q VLANs, this dedicated connection can be partitioned into multiple virtual interfaces. This allows you to use the same connection to access public resources such as objects stored in Amazon S3 using public IP address space, and private resources such as Amazon EC2 instances running within an Amazon Virtual Private Cloud (VPC) using private IP space, while maintaining network separation between the public and private environments. Virtual interfaces can be reconfigured at any time to meet your changing needs.

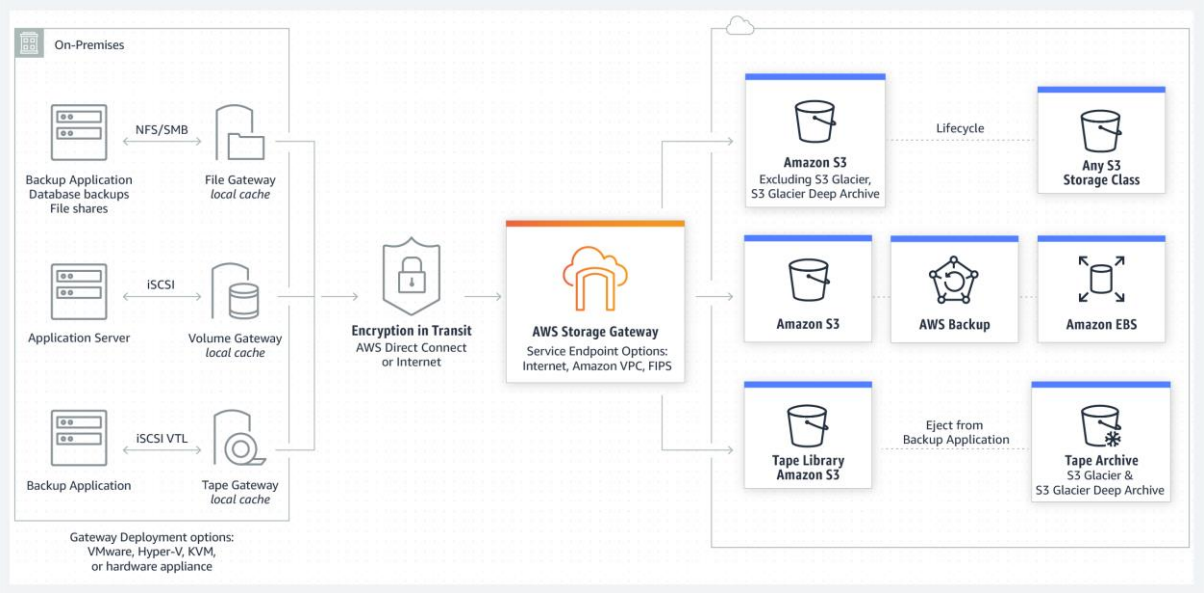
## *How does AWS Direct Connect differ from an AWS VPN Connection?*

A VPN Connection utilizes IPSec to establish encrypted network connectivity between your intranet and Amazon VPC over the Internet. VPN Connections can be configured in minutes and are a good solution if you have an immediate need, have low to modest bandwidth requirements, and can tolerate the inherent variability in Internet-based connectivity.

AWS Direct Connect does not involve the Internet; instead, it uses dedicated, private network connections between your intranet and Amazon VPC.

## *(AWS) Storage Gateway*

- is a hybrid cloud storage service that gives you on-premises access to virtually unlimited cloud storage.
- Customers use Storage Gateway to simplify storage management and reduce costs for key hybrid cloud storage use cases. These include moving backups to the cloud, using on-premises file shares backed by cloud storage, and providing low latency access to data in AWS for on-premises applications, as well as various migration, backup, archiving, processing, moving data to S3 for in-cloud workloads and tiered storage; and disaster recovery use cases.
- It seamlessly integrates on-premises enterprise applications and workflows with Amazon's block and object cloud storage services through industry standard file-storage protocols.
- It provides low-latency performance by caching frequently accessed data on premises, while storing data securely and durably in Amazon cloud storage services. It provides an optimized data transfer mechanism and bandwidth management, which tolerates unreliable networks and minimizes the amount of data being transferred.
- It brings the security, manageability, durability, and scalability of AWS to existing enterprise environments through native integration with AWS encryption, identity management, monitoring, and storage services.



## Users, Groups & Roles

### *What is the Root Account?*

Your root account is the email address you used to set up your AWS account. It has full admin access. Don't give away these account credentials. You should instead create a user for other individuals. Always secure the root account using multi-factor authentication.

(http://kayleigholiver.com/aws-cloud-practitioner-iam/)

### *What tasks require root user credentials?*

- Change your account settings. This includes the account name, email address, root user password, and root user access keys. Other account settings, such as contact information, payment currency preference, and Regions, do not require root user credentials.
- View certain tax invoices. An IAM user with the aws-portal:ViewBilling permission can view and download VAT invoices from AWS Europe, but not AWS Inc or Amazon Internet Services Pvt. Ltd (AISPL).
- Close your AWS account.
- Restore IAM user permissions. If the only IAM administrator accidentally revokes their Own permissions, you can sign in as the root user to edit policies and restore those permissions.
- Change your AWS Support plan or Cancel your AWS Support plan. For more information, see IAM for AWS Support.
- Register as a seller in the Reserved Instance Marketplace.
- Configure an Amazon S3 bucket to enable MFA (multi-factor authentication) Delete.
- Edit or delete an Amazon S3 bucket policy that includes an invalid VPC ID or VPC endpoint ID.
- Sign up for GovCloud.

### *How can you access the AWS platform?*

You can access the AWS platform in 3 ways:

1. Using the Console - Graphical interface to access AWS features
2. Using the CLI (command line interface) - Lets you control AWS services programmatically from command line
3. Using the SDK - Enable you to access AWS using a variety of popular programming languages

### *What is IAM?*

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. IAM is a feature of your AWS account offered at no additional charge. You will be charged only for use of other AWS services by your users.

### *IAM Users*

An IAM user has permanent named operator with long-term credentials and is used to directly interact with AWS services, can be a human or machine. Users can be used globally.

### *IAM Roles*

IAM roles allow you to delegate access with defined permissions to trusted entities without having to share long-term access keys, as such an IAM role does not have any credentials.
An IAM role is an AWS Identity and Access Management (IAM) entity with permissions to make AWS service requests. IAM roles cannot make direct requests to AWS services; they are meant to be assumed by authorized entities, such as IAM users, applications, or AWS services such as EC2. Roles can be used globally.

### *IAM Groups*

An IAM group is a collection of IAM users. Groups let you specify permissions for multiple users, which can make it easier to manage the permissions for those users. All the users in an IAM group inherit the permissions assigned to the group. For example, you could have a group called Admins and give that group the types of permissions that administrators typically need. To set permissions in a group you can change a centrally stored access control policy and all users in the group will immediately inherit any changes.

(http://kayleigholiver.com/aws-cloud-practitioner-iam/)

*What is an IAM access control policy?*

You manage access in AWS by creating policies and attaching them to IAM identities (users, groups of users, or roles) or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. AWS evaluates these policies when an IAM principal (user or role) makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents.

By default, IAM users, groups, and roles have no permissions; users with sufficient permissions must use a policy to grant the desired permissions.

Most policies are stored as JSON documents, for example:


The component parts of an IAM policy (i.e. Effect, Action, and Resource) are defined in a similar manner to an S3 bucket policy.

Managed policies are IAM resources that express permissions using the IAM policy language. You can create, edit, and manage separately from the IAM users, groups, and roles to which they are attached. After you attach a managed policy to multiple IAM users, groups, or roles, you can update that policy in one place and the permissions automatically extend to all attached entities. Managed policies are managed either by you (these are called customer managed policies) or by AWS (these are called AWS managed policies).

### *What is a Managed Policy?*

An AWS managed policy is a access control policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases.

- Full access AWS managed policies such as AmazonDynamoDBFullAccess and IAMFullAccess define permissions for service administrators by granting full access to a service. Power-user AWS managed policies such as AWSCodeCommitPowerUser and AWSKeyManagementServicePowerUser are designed for power users.
- Partial-access AWS managed policies such as AmazonMobileAnalyticsWriteOnlyAccess and AmazonEC2ReadOnlyAccess provide specific levels of access to AWS services without allowing permissions management access level permissions. AWS managed policies make it easier for you to assign appropriate permissions to users, groups, and roles than if you had to write the policies yourself.
- One particularly useful category of AWS managed policies are those designed for job functions. These policies align closely to commonly used job functions in the IT industry. The intent is to make granting permissions for these common job functions easy. One key advantage of using job function policies is that they are maintained and updated by AWS as new services and API operations are introduced.
- You cannot change the permissions defined in AWS managed policies. AWS occasionally updates the permissions defined in an AWS managed policy.

### *What is a Standalone Policy?*

You can create standalone policies that you administer in your own AWS account, which we refer to as customer managed policies. Standalone policies have their own Amazon Resource Name (ARN) that includes the policy name. A great way to create a customer managed policy is to start by copying an existing AWS managed policy. That way you know that the policy is correct at the beginning and all you need to do is customize it to your environment.

### *What is an Inline Policy?*

An inline policy is a policy that's embedded in an IAM identity (a user, group, or role). That is, the policy is an inherent part of the identity. You can create a policy and embed it in an identity, either when you create the identity or later.

For more than one user, group or role to include the same policy, the policy must be copied to that user, group or role. Duplicating it and separating it from the original entirely.

### *What is an AWS account alias?*

The account alias is a name you define to make it more convenient to identify your account, instead of using your AWS ID which is a twelve digit number. You can have one alias per AWS account. You can create an alias using the IAM APIs, AWS Command Line Tools, or the IAM console.

### *What is a federated user?*

With identity federation, external identities are granted secure access to resources in your AWS account without having to create IAM users. These external identities can come from your corporate identity provider (such as Microsoft Active Directory or from the AWS Directory Service) or from a web IdP (identity provider), such as Amazon Cognito, Login with Amazon, Facebook, Google, or any OpenID Connect-compatible provider.

Federated users (external identities) are users you manage outside of AWS in your corporate directory, but to whom you grant access to your AWS account using temporary security credentials. They differ from IAM users, which are created and maintained in your AWS account.

### *IAM Best Practices*

To help secure your AWS resources, follow these recommendations for the AWS Identity and Access Management (IAM) service.

- Lock away your AWS account root user access keys
- Create individual IAM users
- Use groups to assign permissions to IAM users
- Grant least privilege
- Get started using permissions with AWS managed policies
- Use customer managed policies instead of inline policies
- Use access levels to review IAM permissions
- Configure a strong password policy for your users
- Enable MFA – These are not physical MFA tokens typically
- Use roles for applications that run on Amazon EC2 instances
- Use roles to delegate permissions
- Do not share access keys
- Rotate credentials regularly
- Remove unnecessary credentials
- Use policy conditions for extra security
- Monitor activity in your AWS account

### *Password Policies*

In addition to manually creating individual passwords for your IAM users, you can create a password policy that applies to all IAM user passwords in your AWS account.

You can use a password policy to do these things:

- Set a minimum password length.
- Require specific character types, including uppercase letters, lowercase letters, numbers, and non-alphanumeric characters. Be sure to remind your users that passwords are case sensitive.
- Allow all IAM users to change their own passwords.
- Require IAM users to change their password after a specified period of time (enable password expiration).
- Prevent IAM users from reusing previous passwords.
- Force IAM users to contact an account administrator when the user has allowed his or her password to expire.

*(AWS) Directory Service*

- AWS Directory Service for Microsoft Active Directory, also known as AWS Managed Microsoft Active Directory (AD), enables your directory-aware workloads and AWS resources to use managed Active Directory (AD) in AWS.
- AWS Managed Microsoft AD is built on actual Microsoft AD and does not require you to synchronize or replicate data from your existing Active Directory to the cloud. You can use the standard AD administration tools and take advantage of the built-in AD features, such as Group Policy and single sign-on. With AWS Managed Microsoft AD, you can easily join Amazon EC2 and Amazon RDS for SQL Server instances to your domain, and use AWS End User Computing (EUC) services, such as Amazon WorkSpaces, with AD users and groups.
- AWS Managed Microsoft AD makes it easy to extend your existing Active Directory to AWS. It enables you to leverage your existing on-premises user credentials to access cloud resources such as the AWS Management Console, Amazon Workspaces, Amazon Chime, and Windows workloads in the cloud.
- Enable your users to enable single sign-on (SSO) to the AWS Console. This enables your users to sign in with their existing AD credentials, assume one of their assigned roles at sign-in, and to access and take action on the resources according to the permissions defined for the role.

## Security

### *Security Group*

- A security group acts as a virtual firewall for your instance to control inbound and outbound traffic.
- When you launch an instance in a VPC, you can assign up to five security groups to the instance.
- Security groups act at the instance level, not the subnet level. Therefore, each instance in a subnet in your VPC can be assigned to a different set of security groups.
- If you launch an instance and don't specify a security group, the instance is automatically assigned to the default security group for the VPC
- For each security group, you add rules that control the inbound traffic to instances, and a separate set of rules that control the outbound traffic

### *Network Access Control List (Network ACL)*

- 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 an additional layer of security to your VPC.
- Your VPC automatically comes with a modifiable default network ACL. By default, it allows all inbound and outbound IPv4 traffic and, if applicable, IPv6 traffic.

### *What is the difference between security groups and ACLs?*

| Security group | Network ACL |
|---|---|
| Operates at the instance level | Operates at the subnet level |
| Supports allow rules only | Supports allow rules and deny rules |
| Is stateful: Return traffic is automatically allowed, regardless of any rules | Is stateless: Return traffic must be explicitly allowed by rules |
| We evaluate all rules before deciding whether to allow traffic | We process rules in order, starting with the lowest numbered rule, when deciding whether to allow traffic |
| Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on | Automatically applies to all instances in the subnets that it's associated with (therefore, it provides an additional layer of defense if the security group rules are too permissive) |

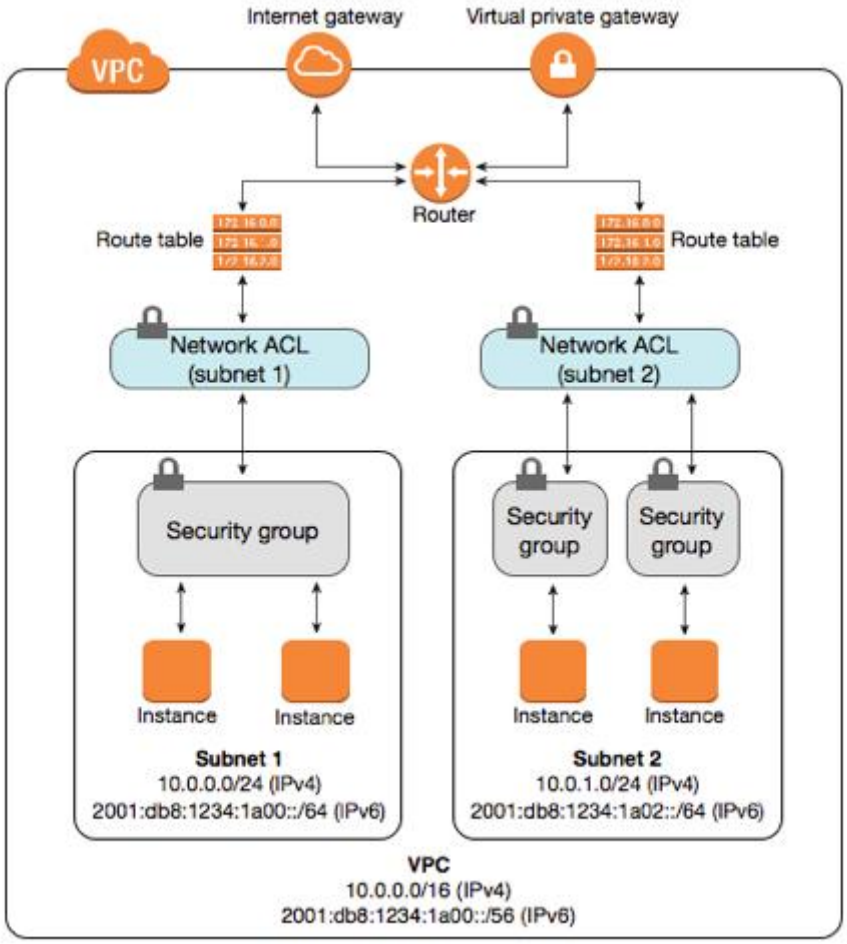



*Figure 2: The following diagram illustrates the layers of security provided by security groups and network ACLs. For example, traffic from an internet gateway is routed to the appropriate subnet using the routes in the routing table. The rules of the network ACL that is associated with the subnet control which traffic is allowed to the subnet. The rules of the security group that is associated with an instance control which traffic is allowed to the instance.*

### *What is the Principle of Least Privilege?*

The Principle of Least Privilege states that a subject should be given only those privileges needed for it to complete its task. If a subject does not need an access right, the subject should not have that access right.

Determine what users (and roles) need to do and then craft policies that allow them to perform only those tasks.

Start with a minimum set of permissions and grant additional permissions as necessary. Doing so is more secure than starting with permissions that are too lenient and then trying to tighten them later.
This principle limits the damage that can result from an accident or error. It also reduces the number of potential interactions among privileged programs to the minimum for correct operation, so that unintentional, unwanted, or improper uses of privilege are less likely to occur.

### *What is data integrity?*

Data integrity is the accuracy, completeness, consistency (validity) and reliability of data throughout its lifecycle.

Compromised data, after all, is of little use to enterprises, not to mention the dangers presented by sensitive data loss. For this reason, maintaining data integrity is a core focus of many enterprise security solutions.

Data integrity can be compromised in several ways. Each time data is replicated or transferred, it should remain intact and unaltered between updates. Error checking methods and validation procedures are typically relied on to ensure the integrity of data that is transferred or reproduced without the intention of alteration.

Data integrity is also related to a security best practice of requiring that secret data remains secret (confidentiality) and unmodified (integrity/authenticity). This is related to data encryption which is the customer's responsibility.

### *Access keys*

Access keys are long-term credentials for an IAM user or the AWS account root user. You can use access keys to sign programmatic requests to the AWS CLI (Command Line Interface), SDK (Software Development Kit), and other development tools.

IAM policies don't have access keys. The only way you will ever get an Access key is to create them from an IAM user.

Access keys consist of an **access key ID** and **secret access key**, which are used to sign programmatic requests that you make to AWS. If you don't have access keys, you can create them from the AWS Management Console. The only time that you can view or download the secret access key is when you create the keys. You cannot recover them later. However, you can create new access keys at any time.

The AWS CLI requires four pieces of information to be used:

- Access key ID
- Secret access key
- AWS Region
- Output format

### *SSH (Secure Shell) Keys*

SSH keys are needed to direct connect and login into an EC2 instance and not to access AWS services. SSH is not required to use AWS CLI.

### *Encryption Keys*

The Public and Private key pair comprise of two uniquely related cryptographickeys (basically long random numbers) known as a **key pair**. Below is an example of a Public Key:

3048 0241 00C9 18FA CF8D EB2D EFD5 FD37 89B9 E069 EA97 FC20 5E35 F577 EE31C4FB C6E4 4811 7D86 BC8F BAFA 362F 922B F01B 2F40 C744 2654 C0DD 2881 D673 CA2B4003 C266 E2CD CB02 0301 0001

The Public Key is what its name suggests - Public. It is made available toeveryone via a publicly accessible repository or directory. On the other hand,the Private Key must remain confidential to its respective owner.

Because the key pair is mathematically related, whatever is encrypted witha Public Key may only be decrypted by its corresponding Private Key and viceversa.

For example, if Bob wants to send sensitive data to Alice, and wants to besure that only Alice may be able to read it, he will encrypt the data with Alice'sPublic Key. Only Alice has access to her corresponding Private Key and as aresult is the only person with the capability of decrypting the encrypted databack into its original form.

As only Alice has access to her Private Key, it is possible that only Alicecan decrypt the encrypted data. Even if someone else gains access to the encrypteddata, it will remain confidential as they should not have access to Alice'sPrivate Key.

### *Penetration Testing Procedures*

AWS customers are welcome to carry out security assessments or penetration tests against their AWS infrastructure without prior approval for 8 services, listed here:

- Amazon EC2 instances, NAT Gateways, and Elastic Load Balancers
- Amazon RDS
- Amazon CloudFront
- Amazon Aurora
- Amazon API Gateways
- AWS Lambda and Lambda Edge functions
- Amazon Lightsail resources
- Amazon Elastic Beanstalk environments

Please ensure that these activities are aligned with the policy set out below. Note: Customers are not permitted to conduct any security assessments of AWS infrastructure, or the AWS services themselves. If you discover a security issue within any AWS services in the course of your security assessment, please contact AWS Security immediately.

Prohibited Activities

- DNS zone walking via Amazon Route 53 Hosted Zones
- Denial of Service (DoS), Distributed Denial of Service (DDoS), Simulated DoS, Simulated DDoS (These are subject to the DDoS Simulation Testing policy)
- Port flooding
- Protocol flooding
- Request flooding (login request flooding, API request flooding)

Things like customized security tests require you to fill out a Simulated Events form telling AWS what it is you want to do. Be sure to include dates, accounts involved, assets involved, and contact information, including phone number and detailed description of planned events. You should expect to receive a non-automated response to your initial contact within 2 business days confirming receipt of your request.

### *AWS Security Bulletins*

No matter how carefully engineered the services are, from time to time it may be necessary to notify customers of security and privacy events with AWS services. We will publish security bulletins online to update our customers of any changes.

### *What do I do if I notice unauthorized activity in my AWS account?*

If you observe unauthorized activity within your AWS account, or you believe that an unauthorized party has accessed your account, then do the following:

- Change your AWS account root user password.
- Rotate and delete all root and AWS Identity and Access Management (IAM) access keys.
- Delete any potentially unauthorized IAM users, and then change the password for all other IAM users.
- Delete any resources on your account that you didn't create, such as Amazon Elastic Compute Cloud (Amazon EC2) instances and AMIs, Amazon Elastic Block Store (Amazon EBS) volumes and snapshots, and IAM users.
- Respond to the notifications that you received from AWS Support through the AWS Support Center.

## *Encryption*

- **Encryption of Data at Rest:**
    - You can create an encrypted file system so all your data and metadata is encrypted at rest using an industry-standard AES-256 (Advanced Encryption Standard) encryption algorithm. Encryption and decryption is handled automatically and transparently, so you don't have to modify your applications. If your organization is subject to corporate or regulatory policies that require encryption of data and metadata at rest, we recommend creating an encrypted file system.
    - You have the following options for protecting data at rest in Amazon S3:
        - Server-Side Encryption – Request Amazon S3 to encrypt your object before saving it on disks in its data centres and then decrypt it when you download the objects.
            - AWS "Server-side encryption means that if you send unencrypted raw data to AWS, on the AWS infrastructure, the raw data is encrypted and finally stored on disk. When you retrieve data, AWS reads the encrypted data from the disk, decrypts the data, and sends raw data back to you. The encryption /decryption is transparent to the AWS user.
        - Client-Side Encryption – Encrypt data client-side and upload the encrypted data to Amazon S3. In this case, you manage the encryption process, the encryption keys, and related tools.
- **Encryption of Data in Transit:**
    - You can mount a file system so all NFS traffic is encrypted in transit using Transport Layer Security 1.2 (TLS, formerly called Secure Sockets Layer [SSL]) with an industry-standard AES-256 cipher. TLS is a set of industry-standard cryptographic protocols used for encrypting information that is exchanged over the wire. AES-256 is a 256-bit encryption cipher used for data transmission in TLS. If your organization is subject to corporate or regulatory policies that require encryption of data and metadata in transit, we recommend setting up encryption in transit on every client accessing the file system.

### *(AWS) Compliance*

Enables customers to understand the robust controls in place at AWS to maintain security and data protection in the cloud. As systems are built on top of AWS cloud infrastructure, compliance responsibilities are shared. By tying together governance-focused, audit friendly service features with applicable compliance or audit standards, AWS Compliance enablers build on traditional programs; helping customers to establish and operate in an AWS security control environment. The IT infrastructure that AWS provides to its customers is designed and managed in alignment with security best practices and a variety of IT security standards, including:

- System and Organization Controls (SOC) Reports - independent third-party examination reports that demonstrate how AWS achieves key compliance controls and objectives. The purpose of these reports is to help you and your auditors understand the AWS controls established to support operations and compliance
    - SOC 1/SSAE 16/ISAE 3402 (formerly SAS 70)
    - SOC 2 - Security, Availability & Confidentiality Report
    - SOC 2 - Privacy Type I Report
    - SOC 3 - Security, Availability & Confidentiality Report
- FISMA, DIACAP, and FedRAMP
- DOD CSM Levels 1-5
- PCI DSS Level 1
- ISO 9001 / ISO 27001 / ISO 27017 / ISO 27018
- ITAR
- FIPS 140-2
- MTCS Level 3
- HITRUST

In addition, the flexibility and control that the AWS platform provides allows customers to deploy solutions that meet several industry-specific standards, including:

- Criminal Justice Information Services (CJIS)
- Cloud Security Alliance (CSA)
- Family Educational Rights and Privacy Act (FERPA)
- Health Insurance Portability and Accountability Act (HIPAA)
- Motion Picture Association of America (MPAA)

### *(AWS) Artifact*

A no cost, self-service portal for on-demand access to for compliance-related information that matters to AWS customers. It provides on-demand access to AWS' security and compliance reports and select online agreements. Reports available in AWS Artifact include our Service Organization Control (SOC) reports, Payment Card Industry (PCI) reports, and certifications from accreditation bodies across geographies and compliance verticals that validate the implementation and operating effectiveness of AWS security controls. Agreements available in AWS Artifact include the Business Associate Addendum (BAA) and the Nondisclosure Agreement (NDA). The AWS SOC 2 report is particularly helpful for completing questionnaires because it provides a comprehensive description of the implementation and operating effectiveness of AWS security controls. Another useful document is the Executive Briefing within the AWS FedRAMP Partner Package.

# Glossary

## A

- Alias Record – See CNAME
- (Amazon) API Gateway –
    - is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs (application programming interface) at any scale.
    - APIs act as the "front door" for applications to access data, business logic, or functionality from your backend services.
    - Using API Gateway, you can create:
        - REST APIs (short for representational state transfer, an architectural style for an API that uses HTTP requests to access and use data). Use HTTP as the underlying protocol for communication, which in turn follows the request and response model where a client sends a request to a service and the service responds back synchronously. This kind of model is suitable for many different kinds of applications that depend on synchronous communication. Essentially, REST is an architectural style which puts a set of constraints on HTTP to create web services.
        - WebSocket APIs (protocol which makes it possible to open a two-way interactive communication session between the user's browser and a server. With this API, you can send messages to a server and receive event-driven responses without having to poll the server for a reply). WebSocket protocol starts off over HTTP, although it is further elevated to follow the WebSockets protocol if both the server and the client are compliant. It is a bidirectional protocol, a client can send messages to a service, and services can independently send messages to clients. Websocket APIs enable real-time two-way communication, this bidirectional behavior enables richer client/service interactions because services can push data to clients without requiring clients to make an explicit request. WebSocket APIs are often used in real-time applications such as chat applications, collaboration platforms, multiplayer games, GPS location tracking, Push Notifications and stock market prices updating in realtime.
    - API Gateway supports containerized and serverless workloads, as well as web applications. It has a collection of API routes that are integrated with backend HTTP endpoints, Lambda functions, or other AWS services.
    - API Gateway handles all the tasks involved in accepting and processing up to hundreds of thousands of concurrent API calls, including traffic management, CORS support, authorization and access control, throttling, monitoring, and API version management.
    - API Gateway has no minimum fees or startup costs. You pay for the API calls you receive and the amount of data transferred out.
- Amazon AppStream 2.0
    - is a fully managed non-persistent application and desktop streaming service. You centrally manage your desktop applications on AppStream 2.0 and securely deliver them to any computer. You can easily scale to any number of users across the globe without acquiring, provisioning, and operating hardware or infrastructure.

AppStream 2.0 is built on AWS, so you benefit from a data center and network architecture designed for the most security-sensitive organizations. Each end user has a fluid and responsive experience because your applications run on virtual machines optimized for specific use cases and each streaming sessions automatically adjust to network conditions.

- ARN (Amazon Resource Name) –
    - uniquely identify AWS resources. We require an ARN whenyouneed to specify a resource unambiguously across all of AWS, such as in IAM policies, Amazon Relational Database Service (Amazon RDS) tags, and API calls.
    - Can be either qualified or unqualified ARN. Qualified ARNs contain a version suffix, while unqualified ARNs do not.
        - Qualified ARN: arn:aws:lambda:aws-region:acct-id:function:helloworld:42
        - Unqualified ARN: arn:aws:lambda:aws-region:acct-id:function:helloworld
- Asynchronous Integration – See Loose Coupling
- (Amazon) Athena –
    - Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run.
    - Queries use standard SQL. Most results are delivered within seconds.

B

## C

- (AWS) Certificate Manager (ACM)
    - AWS Certificate Manager is a service that lets you easily provision, manage, and deploy public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for use with AWS services and your internal connected resources. SSL/TLS certificates are used to secure network communications and establish the identity of websites over the Internet as well as resources on private networks. AWS Certificate Manager removes the time-consuming manual process of purchasing, uploading, and renewing SSL/TLS certificates.
- (AWS) CloudHSM (Hardware Security Module)
    - service helps you meet corporate, contractual, and regulatory compliance requirements for data security by using dedicated Hardware Security Module (HSM) instances within the AWS cloud. AWS and AWS Marketplace partners offer a variety of solutions for protecting sensitive data within the AWS platform, but for some applications and data subject to contractual or regulatory mandates for managing cryptographic keys, additional protection may be necessary.
    - CloudHSM complements existing data protection solutions and allows you to protect your encryption keys within HSMs that are designed and validated to government standards for secure key management. CloudHSM allows you to securely generate, store, and manage cryptographic keys used for data encryption in a way that keys are accessible only by you.
    - A Hardware Security Module (HSM) provides secure key storage and cryptographic operations within a tamper-resistant hardware device. HSMs are designed to securely store cryptographic key material and use the key material without exposing it outside the cryptographic boundary of the hardware.
- CDMB (Configuration Management Database)
    - A CMDB is a repository that acts as a data warehouse – storing information about your IT environment, the components that are used to deliver IT services. The data stored in a CMDB include lists of assets (referred to as configuration items) and the relationships among them.
- CNAME -
    - is a Canonical Name Record or Alias Record. A type of resource record in the Domain Name System (DNS), that specifies that one domain name is an alias of another canonical domain name. Any system hosting a Web site must have an IP address in order to be connected to the World Wide Web.
    - CNAME records must always point to another domain name, never directly to an IP address.
    - A common example is when you have both example.com and www.example.com pointing to the same application and hosted by the same server. To avoid maintaining two different records, it's common to create: (1) An A record for example.com pointing to the server IP address (2) A CNAME record for www.example.com pointing to example.com. As a result, example.com points to the server IP address, and www.example.com points to the same address via example.com. If the IP address changes, you only need to update it in one place: just edit the A record for example.com, and www.example.com automatically inherits the changes.
- (Amazon) Cognito

    - Amazon Cognito lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily. Amazon Cognito scales to millions of users and supports sign-in with social identity providers, such as Facebook, Google, and Amazon, and enterprise identity providers via SAML 2.0
- Colocation center –
    - Also known as a "carrier hotel". Colocation is the practice of housing privately-owned servers and networking equipment in a third-party data center. It can also extend to renting equipment, bandwidth and other resources. It is a shared facility generally with other paying tenants. It is good for businesses that require full control over their equipment. When compared with traditional datacentre there is access to higher levels of bandwidth, higher reliability and higher levels of physical protection. This is different to AWS which manages the entire datacentre themselves and instead provides products packaged as services e.g. EC2 for compute power or S3 for object storage.
- Container –
    - A container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. A container is a uniform structure in which any application can be stored, transported and run.
    - It is named for and often compared to the standardised intermodal containers used in the shipping industry for efficient transportation.
    - In the software world, containerisation is an efficient method for deploying applications. A container encapsulates an application with its own operating environment. It can be placed on any host machine without special configuration.
    - Virtual machines (VMs) and containers are not the same. Deploying VMs is hardware virtualisation whereas containerisation is OS virtualisation. An application on a VM requires a guest OS and thus an underlying hypervisor to run. By contrast, an application in a container, doesn't require a guest OS or hypervisor. It allows an application to run in the Userspace of the OS – a segment of the computer memory that is kept separate from the critical processes of the OS kernel. This leads to improved performance, as an application's instructions do not have to pass through the guest OS and the hypervisor to reach the CPU. It also means that applications in containers are smaller and can be started up in seconds, compared to minutes for VMs. Significantly, container applications offer much more stability - they never hang on the host OS, like VM applications can do, which takes all VMs on the host offline.
    - One of the appeals of using containers is their ability to die gracefully and respawn upon demand. Whether a container's demise is caused by a crash or because it's simply no longer needed when server traffic is low, containers are cheap to start, and they're designed to seamlessly appear and disappear.
    - Containers can be thought of as necessitating three categories of software:
        - Builder: technology used to build a container (e.g. Docker)
        - Engine: technology used to run a container (e.g. Docker)
        - Orchestration: technology used to manage many container (e.g. Kubernetes)
    - Containers are used in PaaS where customer is responsible for app and data and the rest is taken care of by the cloud provider.
- (AWS) Control Tower

    - Automates the process of setting up a new baseline multi-account AWS environment that is secure, well-architected, and ready to use.
    - If you're an enterprise with multiple AWS accounts and teams, cloud setup and governance can be complex and time consuming, slowing down the very innovation you're trying to speed up. AWS Control Tower provides the easiest way to set up and govern a new, secure, multi-account AWS environment based on best practices established through AWS' experience working with thousands of enterprises as they move to the cloud. With AWS Control Tower, builders can provision new AWS accounts in a few clicks, while you have peace of mind knowing your accounts conform to your company-wide policies. If you are building a new AWS environment, starting out on your journey to AWS, starting a new cloud initiative, or are completely new to AWS, Control Tower will help you get started quickly with governance and best practices built-in.
    - A landing zone is a well-architected, multi-account AWS environment that's based on security and compliance best practices. AWS Control Tower automates the setup of a new landing zone using best-practices blueprints for identity, federated access, and account structure.
- (Amazon) Connect
    - Amazon Connect is an easy to use omnichannel cloud contact center that helps you provide superior customer service at a lower cost. Over 10 years ago, Amazon's retail business needed a contact center that would give our customers personal, dynamic, and natural experiences. We couldn't find one that met our needs, so we built it. We've now made this available for all businesses, and today thousands of companies ranging from 10 to tens of thousands of agents use Amazon Connect to serve millions of customers daily.
    - Designed from the ground up to be omnichannel, Amazon Connect provides a seamless experience across voice and chat for your customers and agents. This includes one set of tools for skills-based routing, task management, powerful real-time and historical analytics, and intuitive management tools – all with pay-as-you-go pricing, which means Amazon Connect simplifies contact center operations, improves agent efficiency, and lowers costs. You can set up a contact center in minutes that can scale to support millions of customers from the office or as a virtual contact center.
- CI/CD (continuous integration, continuous delivery/deployment) –
    - CI - is a software development practice in which all developers merge code changes in a central repository multiple times a day
        - With CI, each change in code triggers an automated build-and-test sequence for the given project, providing feedback to the developer(s) who made the change. The entire CI feedback loop should run in less than 10 minutes.
    - CD adds the practice of automating the entire software release process. The purpose of continuous delivery is to ensure that it takes minimal effort to deploy new code.
        - Continuous Delivery includes infrastructure provisioning and deployment, which may be manual and consist of multiple stages. What's important is that all these processes are fully automated, with each run fully logged and visible to the entire team.
        - Continuous deployment (the other possible "CD") can refer to automatically releasing a developer's changes from the repository to production, where it is usable by customers. It addresses the problem of overloading operations

teams with manual processes that slow down app delivery. It builds on the benefits of continuous delivery by automating the next stage in the pipeline.

## D

- (AWS) Data Pipeline –
    - is a web service that you can use to automate the movement and transformation of data. With AWS Data Pipeline, you can define data-driven workflows, so that tasks can be dependent on the successful completion of previous tasks. You define the parameters of your data transformations and AWS Data Pipeline enforces the logic that you've set up.
- (AWS) Database Migration Service
    - AWS Database Migration Service helps you migrate databases to AWS quickly and securely. The source database remains fully operational during the migration, minimizing downtime to applications that rely on the database. The AWS Database Migration Service can migrate your data to and from most widely used commercial and open-source databases.
    - AWS Database Migration Service supports homogeneous migrations such as Oracle to Oracle, as well as heterogeneous migrations between different database platforms, such as Oracle or Microsoft SQL Server to Amazon Aurora. With AWS Database Migration Service, you can continuously replicate your data with high availability and consolidate databases into a petabyte-scale data warehouse by streaming data to Amazon Redshift and Amazon S3.
    - When migrating databases to Amazon Aurora, Amazon Redshift, Amazon DynamoDB or Amazon DocumentDB (with MongoDB compatibility) you can use DMS free for six months.
    - The only requirement to use AWS DMS is that one of your endpoints must be on an AWS service. You can't use AWS DMS to migrate from an on-premises database to another on-premises database.
- Defense in Depth –
    - multiple layers of security controls placed throughout an IT system. Its intent is to provide redundancy in the event a security control fails or a vulnerability is exploited that can cover aspects of personnel, procedural, technical and physical security for the duration of the system's life cycle.
- (AWS) Direct Connect –
    - a cloud service solution that 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, increase bandwidth throughput, and provide a more consistent network experience** than Internet-based connections.
    - Lets you establish a dedicated network connection between your network and one of the AWS Direct Connect locations. Using industry standard 802.1q VLANs, this dedicated connection can be partitioned into multiple virtual interfaces. This allows you to use the same connection to access public resources such as objects stored in Amazon S3 using public IP address space, and private resources such as Amazon EC2 instances running within an Amazon Virtual Private Cloud (VPC) using private IP space, while maintaining network separation between the public and private environments. Virtual interfaces can be reconfigured at any time to meet your changing needs.
- Docker –

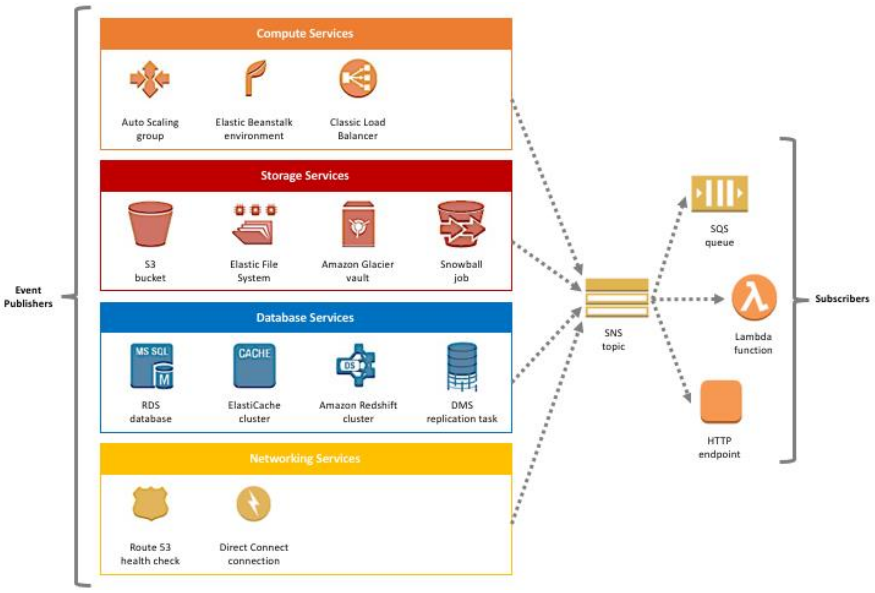
    - an open source project launched in 2013
    - Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. It has became the de facto standard program in this area.
    - It uses OS-level virtualization to deliver software in packages called containers.
    - Docker allows applications to use the same OS kernel as the system leading to more efficient operation
- DNS (Domain Name Service)
    - Translates human readable domain names into numeric IP addresses

## E

- (Amazon) Elastic Container Registry (ECR)
    - is a fully managed container registry that makes it easy to store, manage, share, and deploy your container images and artifacts anywhere. Amazon ECR eliminates the need to operate your own container repositories or worry about scaling the underlying infrastructure. Amazon ECR hosts your images in a highly available and high-performance architecture, allowing you to reliably deploy images for your container applications.
    - You can share container software privately within your organization or publicly worldwide for anyone to discover and download. For example, developers can search the ECR public gallery for an operating system image that is geo-replicated for high availability and faster downloads.
    - Amazon ECR works with Amazon Elastic Kubernetes Service (EKS), Amazon Elastic Container Service (ECS), and AWS Lambda, simplifying your development to production workflow, and AWS Fargate for one-click deployments. Or you can use ECR with your own containers environment. Integration with AWS Identity and Access Management (IAM) provides resource-level control of each repository. With ECR, there are no upfront fees or commitments. You pay only for the amount of data you store in your repositories and data transferred to the Internet.
- (Amazon) Elastic Container Service (ECS)
    - An Amazon ECS cluster is a logical grouping of tasks or services. If you are running tasks or services that use the EC2 launch type, a cluster is also a grouping of container instances. If you are using capacity providers, a cluster is also a logical grouping of capacity providers. When you first use Amazon ECS, a default cluster is created for you, but you can create multiple clusters in an account to keep your resources separate.
- (AWS) Elastic Beanstalk –
    - AWS Elastic Beanstalk is an easy-to-use service for deploying and scaling web applications and services developed with Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker on familiar servers such as Apache, Nginx, Passenger, and IIS.
    - You can simply upload your code and Elastic Beanstalk automatically handles the deployment, from capacity provisioning, load balancing, auto-scaling to application health monitoring. Within minutes, your application will be ready to use without any infrastructure or resource configuration work on your part. At the same time, you retain full control over the AWS resources powering your application and can access the underlying resources at any time.
    - Elastic Beanstalk provisions and operates the infrastructure and manages the application stack (platform) for you, so you don't have to spend the time or develop the expertise. It will also keep the underlying platform running your application up-to-date with the latest patches and updates. Instead, you can focus on writing code rather than spending time managing and configuring servers, databases, load balancers, firewalls, and networks.
    - There is no additional charge for Elastic Beanstalk - you pay only for the AWS resources needed to store and run your applications.
    - Cannot be used in on-premises situations, can only be used for AWS contexts. For on-premises situations AWS OpsWorks and AWS CodeDeploy are suitable.
    - You can configure event notifications for your Elastic Beanstalk environment so that notable events can be automatically published to an SNS topic, then pushed to topic

subscribers. As an example, you may use this event-driven architecture to coordinate your continuous integration pipeline (such as Jenkins CI). That way, whenever an environment is created, Elastic Beanstalk publishes this event to an SNS topic, which triggers a subscribing Lambda function, which then kicks off a CI job against your newly created Elastic Beanstalk environment.

- (Amazon) Elastic Kubernetes Service (EKS) –
    - a fully managed Kubernetes (an open-source container-orchestration system for automating computer application deployment) service.
    - You can choose to run your EKS clusters using AWS Fargate, which is serverless compute for containers. Fargate removes the need to provision and manage servers, lets you specify and pay for resources per application, and improves security through application isolation by design.
    - EKS is deeply integrated with services such as Amazon CloudWatch, Auto Scaling Groups, AWS Identity and Access Management (IAM), and Amazon Virtual Private Cloud (VPC), providing you a seamless experience to monitor, scale, and load-balance your applications.
    - EKS integrates with AWS App Mesh and provides a Kubernetes native experience to consume service mesh features and bring rich observability, traffic controls and security features to applications
    - EKS provides a scalable and highly-available control plane that runs across multiple availability zones to eliminate a single point of failure.
    - Containers are used in PaaS where customer is responsible for app and data and the rest is taken care of by the cloud provider.
- Endpoint –
    - The URL of the entry point for an AWS web service. It may include a region code if the service supports regions
    - For example: https://awsexamplebucket/s3-us-west-2.amazonaws.com/docs/hello.txt
- Event-driven computing
    - Given the context of microservices, event-driven computing is a model in which subscriber services automatically perform work in response to events triggered by publisher services. This paradigm can be applied to automate workflows while decoupling the services that collectively and independently work to fulfil these workflows. Amazon SNS is an event-driven computing hub, in the AWS Cloud, that has native integration with several AWS publisher and subscriber services.




- (Amazon) Eventsbridge
    - Amazon EventBridge is a service that provides real-time access to changes in data in AWS services, your own applications and Software-as-a-Service (SaaS) applications without writing code. To get started, you can choose an event source on the Amazon EventBridge console, and select a target from a number of AWS services including AWS Lambda, Amazon SNS, and Amazon Kinesis Data Firehose. Amazon EventBridge will automatically deliver the events in near real-time.
    - Amazon EventBridge builds upon and extends CloudWatch Events. It uses the same service API and endpoint, and the same underlying service infrastructure. However, it has new features also that enable customers to connect data from their own apps and third-party SaaS apps

F

- Failback –
  - Failback is the process of restoring operations to a primary machine or facility after they have been shifted to a secondary machine or facility during failover.
  - In a failback stage, the process uses something called change data, which represents changes made to the system under duress, or in other words, changes made only in the backup system. In failback, only the change data is sent to the original system. There is no need to copy an entire drive or set of drives; failback just adds what was recorded by the backup facility during the duration of the crisis.
  - One of the implied characteristics of a failback system is that the process is done automatically.
- Failover –
  - is a backup operational mode in which the functions of a system component (such as a processor, server, network, or database, for example) are assumed by secondary system components when the primary component becomes unavailable through either failure or scheduled down time.
  - One of the implied characteristics of a failover system is that the process is done automatically.
  - Two main types of failover:
    - Active-active failover - Use this failover configuration when you want all of your resources to be available the majority of the time. When a resource becomes unavailable, Route 53 can detect that it's unhealthy and stop including it when responding to queries. In active-active failover, all the records that have the same name, the same type (such as A or AAAA), and the same routing policy (such as weighted or latency) are active unless Route 53 considers them unhealthy. Route 53 can respond to a DNS query using any healthy record.
    - Active-passive failover - Use an active-passive failover configuration when you want a primary resource or group of resources to be available the majority of the time and you want a secondary resource or group of resources to be on standby in case all the primary resources become unavailable. When responding to queries, Route 53 includes only the healthy primary resources. If all the primary resources are unhealthy, Route 53 begins to include only the healthy secondary resources in response to DNS queries.

## G

- Granularity
    - The degree to which you have control or access over a given setting in AWS. For example:
        - In CloudWatch certain metrics are available at 1-minute granularity
        - AWS Identity and Access Management (IAM) allows customers to provide granular access control to resources in AWS

## H

- Hypervisor
    - a virtual machine monitor (VMM) that allows many virtual operating systems to run simultaneously on one computer system. These virtual machines are also called guest machines, and they all share the hardware of the physical machine, such as memory, processor, storage, and other related resources.

|

- IaaS (Infrastructure as a Service)
- IAM
    - Users
    - Roles
- IdP –
    - an identity provider, that manages your user identities outside of AWS, such as Login with Amazon, Facebook, or Google
- (AWS) Infrastructure Event Management (IEM)
    - offers architecture and scaling guidance and operational support during the preparation and execution of planned events, such as shopping holidays, product launches, and migrations. For these events, AWS Infrastructure Event Management will help you assess operational readiness, identify and mitigate risks, and execute your event confidently with AWS experts by your side. The program is included in the Enterprise Support plan and is available to Business Support customers for an additional fee.
- (Amazon) Inspector –
    - Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. Amazon Inspector automatically assesses applications for exposure, vulnerabilities, and deviations from best practices. After performing an assessment, Amazon Inspector produces a detailed list of security findings prioritized by level of severity. These findings can be reviewed directly or as part of detailed assessment reports which are available via the Amazon Inspector console or API.
    - Amazon Inspector security assessments help you check for unintended network accessibility of your Amazon EC2 instances and for vulnerabilities on those EC2 instances. Amazon Inspector assessments are offered to you as pre-defined rules packages mapped to common security best practices and vulnerability definitions. Examples of built-in rules include checking for access to your EC2 instances from the internet, remote root login being enabled, or vulnerable software versions installed. These rules are regularly updated by AWS security researchers.
- IOPS –
    - Input/Output operations per second. The operations are measured in KiB.
        - It is a performance metric used to distinguish one storage type from another.
        - The underlying drive technology determines the maximum amount of data that a volume type counts as a single I/O.
        - SSD volumes generally I/O much more efficiently than HDD volumes
- (AWS) IoT Greengrass –
    - is a service that extends Amazon Web Services functionality to Internet of Things (IoT) devices, allowing a business to perform data collection and analysis closer to its origin.
    - AWS IoT Greengrass seamlessly extends AWS to edge devices so they can act locally on the data they generate, while still using the cloud for management, analytics, and durable storage. With AWS IoT Greengrass, connected devices can run AWS Lambda functions, Docker containers, or both, execute predictions based on machine

learning models, keep device data in sync, and communicate with other devices securely – even when not connected to the Internet.

J

- JSON –
  - stands for JavaScript Object Notation
  - is an lightweight open standard file format, and data interchange format, that uses human-readable text to store and transmit data objects consisting of attribute–value pairs and array data types (or any other serializable value).
  - It is a very common data format, with a diverse range of applications

## K

- (Amazon) Kinesis -
    - Amazon Kinesis makes it easy to collect, process, and analyze real-time, streaming data so you can get timely insights and react quickly to new information.
    - Amazon Kinesis offers key capabilities to cost-effectively process streaming data at any scale, along with the flexibility to choose the tools that best suit the requirements of your application.
    - With Amazon Kinesis, you can ingest real-time data such as video, audio, application logs, website clickstreams, and IoT telemetry data for machine learning, analytics, and other applications.
    - Amazon Kinesis enables you to process and analyze data as it arrives and respond instantly instead of having to wait until all your data is collected before the processing can begin.
- (AWS) Key Management Service (KMS) –
    - Easily create and control the customer master keys (CMKs), the encryption keys used to encrypt or digitally sign your data. Makes it easy for you to create and manage cryptographic keys and control their use across a wide range of AWS services and in your applications. AWS KMS is a secure and resilient service that uses hardware security modules that have been validated under FIPS 140-2, or are in the process of being validated, to protect your keys. AWS KMS is integrated with AWS CloudTrail to provide you with logs of all key usage to help meet your regulatory and compliance needs.
- Kubernetes -
    - The name Kubernetes originates from Greek, meaning helmsman or pilot.
    - It was originally designed by Google and is now maintained by the Cloud Native Computing Foundation
    - is an open-source container-orchestration system for automating computer application deployment, networking, load-balancing, security, scaling, and management
    - It will orchestrate the running of containers across a potentially large number of hosts (also called nodes), these can be Docker hosts, bare-metal servers or virtual machines.
    - A collection of nodes that is managed by a single Kubernetes instance is referred to as a Kubernetes cluster.
    - Kubernetes can control its clusters from a single command line or dashboard.
- KPI (Key performance indicator) –
    - a type of performance measurement. KPIs evaluate the success of an organization or of a particular activity in which it engages

L

- (AWS) Lake Formation –
    - a service that makes it easy to set up a secure data lake in days.
    - A data lake is a centralized, curated, and secured repository that stores all your data, both in its original form and prepared for analysis. A data lake enables you to break down data silos and combine different types of analytics to gain insights and guide better business decisions.
    - However, setting up and managing data lakes today involves a lot of manual, complicated, and time-consuming tasks. This work includes loading data from diverse sources, monitoring those data flows, setting up partitions, turning on encryption and managing keys, defining transformation jobs and monitoring their operation, re-organizing data into a columnar format, configuring access control settings, deduplicating redundant data, matching linked records, granting access to data sets, and auditing access over time.
    - Creating a data lake with Lake Formation is as simple as defining data sources and what data access and security policies you want to apply. Lake Formation then helps you collect and catalog data from databases and object storage, move the data into your new Amazon S3 data lake, clean and classify your data using machine learning algorithms, and secure access to your sensitive data. Your users can access a centralized data catalog which describes available data sets and their appropriate usage. Your users then leverage these data sets with their choice of analytics and machine learning services, like Amazon Redshift, Amazon Athena, and (in beta) Amazon Elastic MapReduce (EMR) for Apache Spark. Lake Formation builds on the capabilities available in AWS Glue.
- Amazon Lex –
    - is a service for building conversational interfaces into any application using voice and text. Amazon Lex provides the advanced deep learning functionalities of automatic speech recognition (ASR) for converting speech to text, and natural language understanding (NLU) to recognize the intent of the text, to enable you to build applications with highly engaging user experiences and lifelike conversational interactions. With Amazon Lex, the same deep learning technologies that power Amazon Alexa are now available to any developer, enabling you to quickly and easily build sophisticated, natural language, conversational bots (“chatbots”).
    - With Amazon Lex, you can build bots to increase contact center productivity, automate simple tasks, and drive operational efficiencies across the enterprise. As a fully managed service, Amazon Lex scales automatically, so you don’t need to worry about managing infrastructure.


*Figure 3: An example Amazon Lex implementation*

- (Amazon) Lightsail
    - is the easiest way to get started with AWS for developers, small businesses, students, and other users who need a solution to build and host their applications on cloud. Lightsail provides developers compute, storage, and networking capacity and capabilities to deploy and manage websites and web applications in the cloud. Lightsail includes everything you need to launch your project quickly – virtual machines, containers, databases, CDN, load balancers, DNS management etc. – for a low, predictable monthly price.
    - You can get preconfigured virtual private server plans that include everything to easily deploy and manage your application. Lightsail is best suited to projects that require a few virtual private servers and users who prefer a simple management interface. Common use cases for Lightsail include running websites, web applications, blogs, e-commerce sites, simple software, and more.
- Local Zone - an AWS infrastructure deployment that places select services closer to your end users. A Local Zone is an extension of a Region that is in a different location from your Region. It provides a high-bandwidth backbone to the AWS infrastructure and is ideal for latency-sensitive applications, for example machine learning.

## M

- (Amazon) Macie
    - Amazon Macie is a fully managed data security and data privacy service that uses machine learning and pattern matching to discover, classify and protect your sensitive data in AWS.
    - Macie recognizes sensitive data such as personally identifiable information (PII) or intellectual property. It provides you with dashboards and alerts that give visibility into how this data is being accessed or moved.
    - As organizations manage growing volumes of data, identifying and protecting their sensitive data at scale can become increasingly complex, expensive, and time-consuming. Amazon Macie automates the discovery of sensitive data at scale and lowers the cost of protecting your data.
    - Macie automatically provides an inventory of Amazon S3 buckets including a list of unencrypted buckets, publicly accessible buckets, and buckets shared with AWS accounts outside those you have defined in AWS Organizations. Then, Macie applies machine learning and pattern matching techniques to the buckets you select to identify and alert you to sensitive data.
    - Macie's alerts, or findings, can be searched and filtered in the AWS Management Console and sent to Amazon EventBridge, for easy integration with existing workflow or event management systems, or to be used in combination with AWS services, such as AWS Step Functions to take automated remediation actions.
    - All this can help you meet regulations, such as the Health Insurance Portability and Accountability Act (HIPAA) and General Data Privacy Regulation (GDPR).
- Microservices -
    - are an architectural and organizational approach to software development where software is composed of small independent services that communicate over well-defined APIs. Services are built for business capabilities and each service performs a single function. Because they are independently run, each service can be updated, deployed, and scaled to meet demand for specific functions of an application. Microservices architectures are typically faster to develop, enabling innovation and accelerating time-to-market for new features.
    - Microservices contrast with monolithic architectures, where all processes are tightly coupled and run as a single service, meaning that if one process of the application experiences a spike in demand, the entire architecture must be scaled. Adding or improving a monolithic application's features becomes more complex as the code base grows. This complexity limits experimentation and makes it difficult to implement new ideas. Monolithic architectures add risk for application availability because many dependent and tightly coupled processes increase the impact of a single process failure.
- Monolithic – See Microservices

## N

- NAS –
  - Network attached storage – single storage devices that provides file-level storage to clients on the network
- NAT (Network Address Translation)
  - You can use a NAT instance in a public subnet in your VPC to enable instances in the private subnet to initiate outbound IPv4 traffic to the internet or other AWS services, but prevent the instances from receiving inbound traffic initiated by someone on the internet.
- (Amazon) Neptune
  - Amazon Neptune is a fast, reliable, fully managed graph database service that makes it easy to build and run applications that work with highly connected datasets. The core of Amazon Neptune is a purpose-built, high-performance graph database engine optimized for storing billions of relationships and querying the graph with milliseconds latency. Amazon Neptune supports popular graph models Property Graph and W3C's RDF, and their respective query languages Apache TinkerPop Gremlin and SPARQL, allowing you to easily build queries that efficiently navigate highly connected datasets. Neptune powers graph use cases such as recommendation engines, fraud detection, knowledge graphs, drug discovery, and network security.
  - Amazon Neptune is highly available, with read replicas, point-in-time recovery, continuous backup to Amazon S3, and replication across Availability Zones. Neptune is secure with support for HTTPS encrypted client connections and encryption at rest. Neptune is fully managed, so you no longer need to worry about database management tasks such as hardware provisioning, software patching, setup, configuration, or backups.

O

- OpenStack –
    - is a free open standard cloud computing platform, mostly deployed as infrastructure-as-a-service in both public and private clouds where virtual servers and other resources are made available to users

P

- PaaS (Platform as a Service)
- (Amazon) Pinpoint –
    - a flexible and scalable outbound and inbound multichannel marketing communications service. You can connect with customers over channels like email, SMS, push, or voice. Segment your campaign audience for the right customer and personalize your messages with the right content. Delivery and campaign metrics in Amazon Pinpoint measure the success of your communications. Amazon Pinpoint can grow and scales globally
- (Amazon) Polly -
    - is a service that turns text into lifelike speech, allowing you to create applications that talk, and build entirely new categories of speech-enabled products. Polly's Text-to-Speech (TTS) service uses advanced deep learning technologies to synthesize natural sounding human speech. With dozens of lifelike voices across a broad set of languages, you can build speech-enabled applications that work in many different countries.
    - In addition to Standard TTS voices, Amazon Polly offers Neural Text-to-Speech (NTTS) voices that deliver advanced improvements in speech quality through a new machine learning approach. Polly's Neural TTS technology also supports two speaking styles that allow you to better match the delivery style of the speaker to the application: a Newscaster reading style that is tailored to news narration use cases, and a Conversational speaking style that is ideal for two-way communication like telephony applications. Finally, Amazon Polly Brand Voice can create a custom voice for your organization. This is a custom engagement where you will work with the Amazon Polly team to build an NTTS voice for the exclusive use of your organization.
- Principle of Least Privilege
- (AWS) Private Link –
    - provides private connectivity between VPCs and services hosted on AWS or on-premises, securely on the Amazon network. By providing a private endpoint to access your services, AWS PrivateLink ensures your traffic is not exposed to the public internet. AWS PrivateLink makes it easy to connect services across different accounts and VPCs to significantly simplify your network architecture.

Q

- Amazon QuickSight –
    - is a fast, cloud-powered business intelligence service that makes it easy to deliver insights to everyone in your organization.
    - as a fully managed service, QuickSight lets you easily create and publish interactive dashboards that include ML Insights. Dashboards can then be accessed from any device, and embedded into your applications, portals, and websites.
    - with our Pay-per-Session pricing, QuickSight allows you to give everyone access to the data they need, while only paying for what you use.
- (AWS) Quick Starts –
    - are built by AWS solutions architects and partners to help you deploy popular technologies on AWS, based on AWS best practices for security and high availability. These accelerators reduce hundreds of manual procedures into just a few steps, so you can build your production environment quickly and start using it immediately.

## R

- (Amazon) Redshift –
    - Amazon Redshift is the most widely used cloud data warehouse. It makes it fast, simple and cost-effective to analyze all your data using standard SQL and your existing Business Intelligence (BI) tools. It allows you to run complex analytic queries against terabytes to petabytes of structured and semi-structured data, using sophisticated query optimization, columnar storage on high-performance storage, and massively parallel query execution. Most results come back in seconds.
    - Amazon Redshift manages the work needed to set up, operate, and scale a data warehouse. For example, provisioning the infrastructure capacity, automating ongoing administrative tasks such as backups, and patching, and monitoring nodes and drives to recover from failures. Redshift also has automatic tuning capabilities, and surfaces recommendations for managing your warehouse in Redshift Advisor. For Redshift Spectrum, Amazon Redshift manages all the computing infrastructure, load balancing, planning, scheduling and execution of your queries on data stored in Amazon S3.
    - The name means to shift away from Oracle, red being an allusion to Oracle.
- Root Account

## S

- SaaS (Software as a Service)
- (Amazon) Sagemaker –
    - Service which provides ability to build, train and deploy machine learning (ML) models quickly
- SAN –
    - a specialized, high-speed network that provides block-level network access to storage. Used to improve application availability (e.g., multiple data paths), enhance application performance, increase storage effectiveness and improve data protection and security.
- SCPs (Service control policies) –
    - Service control policies (SCPs) are a type of organization policy that you can use to manage permissions in your organization. SCPs offer central control over the maximum available permissions for all accounts in your organization. SCPs help you to ensure your accounts stay within your organization's access control guidelines. SCPs are available only in an organization that has all features enabled.
    - Central security administrators use SCPs with AWS Organizations to establish controls that all IAM principals (users and roles) adhere to. Now, you can use SCPs to set permission guardrails with the fine-grained control supported in the AWS Identity and Access Management (IAM) policy language. This makes it easier for you to fine-tune policies to meet the precise requirements of your organization's governance rules.
- (AWS) Secrets Manager
    - AWS Secrets Manager helps you protect secrets needed to access your applications, services, and IT resources. The service enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle. Users and applications retrieve secrets with a call to Secrets Manager APIs, eliminating the need to hardcode sensitive information in plain text. Secrets Manager offers secret rotation with built-in integration for Amazon RDS, Amazon Redshift, and Amazon DocumentDB. Also, the service is extensible to other types of secrets, including API keys and OAuth tokens. In addition, Secrets Manager enables you to control access to secrets using fine-grained permissions and audit secret rotation centrally for resources in the AWS Cloud, third-party services, and on-premises.
- Sharding –
    - also known as horizontal partitioning, is a popular scale-out approach for relational databases to achieve high scalability, high availability, and fault tolerance for data storage.
    - Sharding is a technique that splits data into smaller subsets and distributes them across a number of physically separated database servers. Each server is referred to as a database shard. All database shards usually have the same type of hardware, database engine, and data structure to generate a similar level of performance. However, they have no knowledge of each other, which is the key characteristic that differentiates sharding from other scale-out approaches such as database clustering or replication.
    - The share-nothing model offers the sharded database architecture unique strengths in scalability and fault tolerance. There is no need to manage communications and

contentions among database members. The complexities and overhead involved in doing so don't exist. If one database shard has a hardware issue or goes through failover, no other shards are impacted because a single point of failure or slowdown is physically isolated.

- (AWS) Shield –
    - a managed Distributed Denial of Service (DDoS) protection service that safeguards applications running on AWS. AWS Shield provides always-on detection and automatic inline mitigations that minimize application downtime and latency, so there is no need to engage AWS Support to benefit from DDoS protection. There are two tiers of AWS Shield - Standard and Advanced.
    - All AWS customers benefit from the automatic protections of AWS Shield Standard, at no additional charge. AWS Shield Standard defends against most common, frequently occurring network and transport layer DDoS attacks that target your web site or applications. When you use AWS Shield Standard with Amazon CloudFront and Amazon Route 53, you receive comprehensive availability protection against all known infrastructure (Layer 3 and 4) attacks.
    - For higher levels of protection against attacks targeting your applications running on Amazon Elastic Compute Cloud (EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator and Amazon Route 53 resources, you can subscribe to AWS Shield Advanced. In addition to the network and transport layer protections that come with Standard, AWS Shield Advanced provides additional detection and mitigation against large and sophisticated DDoS attacks, near real-time visibility into attacks, and **integration with AWS WAF, a web application firewall**. AWS WAF is included with AWS Shield Advanced at no additional cost. AWS Shield Advanced also gives you 24x7 access to the AWS DDoS Response Team (DRT) and protection against DDoS related spikes in your Amazon Elastic Compute Cloud (EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator and Amazon Route 53 charges.
    - AWS Shield Advanced is available globally on all Amazon CloudFront, AWS Global Accelerator, and Amazon Route 53 edge locations. You can protect your web applications hosted anywhere in the world by deploying Amazon CloudFront in front of your application. Your origin servers can be Amazon S3, Amazon Elastic Compute Cloud (EC2), Elastic Load Balancing (ELB), or a custom server outside of AWS. You can also enable AWS Shield Advanced directly on an Elastic IP or Elastic Load Balancing (ELB) in certain regions
- (Amazon) Simple Notification Service -
    - is a fully managed messaging service for both application-to-application (A2A) and application-to-person (A2P) communication.
        - The A2A pub/sub functionality provides topics for high-throughput, push-based, many-to-many messaging between distributed systems, microservices, and event-driven serverless applications. Using Amazon SNS topics, your publisher systems can fanout messages to a large number of subscriber systems including Amazon SQS queues, AWS Lambda functions and HTTPS endpoints, for parallel processing.
        - The A2P functionality enables you to send messages to users at scale via SMS, mobile push, and email.
- (Amazon) Simple Queue Service (SQS) -

    - is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. SQS eliminates the complexity and overhead associated with managing and operating message oriented middleware, and empowers developers to focus on differentiating work.
    - Using SQS, you can send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available.
    - SQS offers two types of message queues:
        - Standard queues offer maximum throughput, best-effort ordering, and at-least-once delivery.
        - SQS FIFO queues are designed to guarantee that messages are processed exactly once, in the exact order that they are sent
- SSH (Secure Shell)
    - is a network protocol that gives users, particularly system administrators, a secure way to access a computer over an unsecured network
    - Typical applications include remote command-line, login, and remote command execution, but any network service can be secured with SSH
- SSL (Secure Sockets Layer) – See TLS
- Stateful / Stateless –
    - A stateful web service will keep track of the "state" of a client's connection and data over several requests. So for example, the client might login, select a users account data, update their address, attach a photo, and change the status flag, then disconnect.
    - In a stateless web service, the server doesn't keep any information from one request to the next. The client needs to do it's work in a series of simple transactions, and the client has to keep track of what happens between requests. So in the above example, the client needs to do each operation separately: connect and update the address, disconnect. Connect and attach the photo, disconnect. Connect and change the status flag, disconnect.
    - To handle the removal of instances without impacting your service, you need to ensure that your application instances are stateless. This means that all system and application state is stored and managed outside of the instances themselves.
    - The essence of a stateless installation is that the scalable components are disposable, and configuration is stored away from the disposable components. A stateless web service is much simpler to implement, and can handle greater volume of clients.
- (AWS) Snowcone -
    - smallest member of the AWS Snow Family of edge computing, edge storage, and data transfer devices, weighing in at 4.5 pounds (2.1 kg) with 8 terabytes of usable storage.
    - Snowcone is ruggedized, secure, and purpose-built for use outside of a traditional data center. Its small form factor makes it a perfect fit for tight spaces or where portability is a necessity. You can use Snowcone in backpacks on first responders, or for IoT, vehicular, and even drone use cases.
    - You can execute compute applications at the edge, and you can ship the device with data to AWS for offline data transfer, or you can transfer data online with AWS DataSync from edge locations.

    - Snowcone has multiple layers of security and encryption. You can use either of these services to run edge computing workloads that use AWS IoT Greengrass or Amazon EC2 instances, or to collect, process, and transfer data to AWS. Snowcone is designed for data migration needs up to dozens of terabytes (with up to 8 terabytes per device) and from space-constrained environments where AWS Snowball devices will not fit.
- (AWS) Snowball -
    - Petabyte-scale data transport with on-board storage and compute capabilities
    - part of the AWS Snow Family, is an edge computing, data migration, and edge storage device that comes in two options.
        - Snowball Edge Storage Optimized devices provide both block storage and Amazon S3-compatible object storage, and 40 vCPUs. They are well suited for local storage and large scale-data transfer.
        - Snowball Edge Compute Optimized devices provide 52 vCPUs, block and object storage, and an optional GPU for use cases like advanced machine learning and full motion video analysis in disconnected environments.
    - You can use these devices for data collection, machine learning and processing, and storage in environments with intermittent connectivity (like manufacturing, industrial, and transportation) or in extremely remote locations (like military or maritime operations) before shipping them back to AWS. These devices may also be rack mounted and clustered together to build larger temporary installations.
    - Snowball supports specific Amazon EC2 instance types and AWS Lambda functions, so you can develop and test in the AWS Cloud, then deploy applications on devices in remote locations to collect, pre-process, and ship the data to AWS. Common use cases include data migration, data transport, image collation, IoT sensor stream capture, and machine learning.

- (AWS) Snowmobile –
    - Part of AWS Snow Family. An **Exabyte-scale data transfer service used to move extremely large amounts of data to AWS**. You can transfer **up to 100PB per Snowmobile, a 45-foot long ruggedized shipping container, pulled by a semi-trailer truck**. Snowmobile makes it easy to move massive volumes of data to the cloud, including video libraries, image repositories, or even a complete data center migration. Transferring data with Snowmobile is more secure, fast and cost effective.
    - After an initial assessment, a Snowmobile will be transported to your data center and AWS personnel will configure it for you so it can be accessed as a network storage target. When your Snowmobile is on site, AWS personnel will work with your team to connect a removable, high-speed network switch from Snowmobile to your local network and you can begin your high-speed data transfer from any number of sources within your data center to the Snowmobile. After your data is loaded, Snowmobile is driven back to AWS where **your data is imported into Amazon S3.**
    - Snowmobile uses multiple layers of security to help protect your data including dedicated security personnel, GPS tracking, alarm monitoring, 24/7 video surveillance, and an optional escort security vehicle while in transit. All data is encrypted with 256-bit encryption keys you manage through the AWS Key Management Service (KMS) and designed for security and full chain-of-custody of your data.
- (AWS) Step Functions -
    - a serverless function orchestrator that makes it easy to sequence AWS Lambda functions and multiple AWS services into business-critical applications.
    - Step Functions allow us to design and build the flow of execution of AWS serverless modules in our application in a simplified manner
    - Through its visual interface, you can create and run a series of checkpointed and event-driven workflows that maintain the application state.
    - The output of one step acts as an input to the next. Each step in your application executes in order, as defined by your business logic…
    - This enables a developer to focus solely on ensuring that each module performs its intended task, without having to worry about connecting each module with others.
- (AWS) Systems Manager –
    - gives you visibility and control of your infrastructure on AWS. Provides a unified user interface so you can view operational data from multiple AWS services and allows you to automate operational tasks across your AWS resources.
    - With dista, you can group resources, like Amazon EC2 instances, Amazon S3 buckets, or Amazon RDS instances, by application, view operational data for monitoring and troubleshooting, and take action on your groups of resources.
    - Systems Manager simplifies resource and application management, shortens the time to detect and resolve operational problems, and makes it easy to operate and manage your infrastructure securely at scale.
    - AWS Systems Manager Maintenance Windows let you define a schedule for when to perform potentially disruptive actions on your instances such as patching an operating system, updating drivers, or installing software or patches.

- Use Maintenance Windows to set up recurring schedules for managed instances to run administrative tasks like installing patches and updates without interrupting business-critical operations.

# T

- (Amazon) Transcribe –
    - makes it easy for developers to add speech to text capabilities to their applications. Audio data is virtually impossible for computers to search and analyze. Therefore, recorded speech needs to be converted to text before it can be used in applications. Historically, customers had to work with transcription providers that required them to sign expensive contracts and were hard to integrate into their technology stacks to accomplish this task. Many of these providers use outdated technology that does not adapt well to different scenarios, like low-fidelity phone audio common in contact centers, which results in poor accuracy.
    - Amazon Transcribe uses a deep learning process called automatic speech recognition (ASR) to convert speech to text quickly and accurately. Amazon Transcribe can be used to transcribe customer service calls, to automate closed captioning and subtitling, and to generate metadata for media assets to create a fully searchable archive. You can use Amazon Transcribe Medical to add medical speech to text capabilities to clinical documentation applications.
- (TLS) Transport Layer Security –
    - a cryptographic protocol designed to provide communications security over a computer network. When sender and recipient computers send data they both agree to encrypt the information in a way they both understand. If either machine cannot support an encrypted connection, both services will default to a less secure Secure Sockets Layer (SSL) connection, a non-encrypted connection or may simply refuse to connect, all depending on the rules in place.
    - A public encryption key is used to encrypt data while a private key only held by the data recipient is used to decrypt the data
    - Uses include secure web browsing (and in particular the padlock icon that appears in web browsers when a secure session is established) and sending and receiving emails securely.

## U

- Unified SAN –
    - NAS & SAN used together

V

- VMware –
    - is an American publicly traded software company from California, USA. It provides cloud computing and virtualization software and services. Based in Palo Alto, California. Founded in 1998, VMware is a subsidiary of Dell Technologies

## W

- (AWS) 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. AWS WAF gives you control over how traffic reaches your applications by **enabling you to create security rules that block common attack patterns, such as SQL injection or cross-site scripting, and rules that filter out specific traffic patterns you define**. You can monitor many attributes of traffic, such as, IP addresses, URI strings, HTTP headers and HTTP methods (more details).
    - You can get started quickly using Managed Rules for AWS WAF, a pre-configured set of rules managed by AWS or AWS Marketplace Sellers. The Managed Rules for WAF address issues like the OWASP Top 10 security risks. These rules are regularly updated as new issues emerge. AWS WAF includes a full-featured API that you can use to automate the creation, deployment, and maintenance of security rules.
    - With AWS WAF, **you pay only for what you use. The pricing is based on how many rules you deploy and how many web requests your application receives**. There are no upfront commitments.
    - **You can deploy AWS WAF on Amazon CloudFront** as part of your CDN solution, the Application Load Balancer that fronts your web servers or origin servers running on EC2, or Amazon API Gateway for your APIs.
    - AWS WAF is included with AWS Shield Advanced at no additional cost.
- Websocket –
    - is a computer communications protocol, providing full-duplex communication channels over a single TCP connection. This contrasts with HTTP which is unidirectional where the client sends the request and the server then sends the response

X

## Y

- YAML –
  - is a recursive acronym for "YAML Ain't Markup Language"
  - is a human-readable data-serialization language. It is commonly used for configuration files and in applications where data is being stored or transmitted
  - uses key-value pairs to store data

Z

*To Be Done*

1. Read through all other AWS services and ensure they are in the guide
2. Acronym guide / glossary complete
3. Organise every topic into its correct section
4. Misc.
    a. NACL or ACL?
    b. Amazon EC2 Auto Scaling or AWS Autoscaling

*Completed*

1. ~~Transfer answers into AWS Guide format (up to #512 done so far)~~
2. ~~Check syllabus to see if anything is missing from the guide~~
3. ~~Information from~~ ~~http://kayleigholiver.com/aws-cloud-practitioner-preparation-exam-notes/~~)
    a. ~~Cloud Computing and the Topics To Cover~~
    b. ~~AWS Global Infrastructure~~
    c. ~~AWS Cost Management~~
    d. ~~Identity Access Management (IAM)~~
    e. ~~Simple Storage Service (S3)~~
    f. ~~CloudFront~~
    g. ~~Elastic Compute Cloud (EC2)~~
    h. ~~Roles~~
    i. ~~Load Balancers~~
    j. ~~Databases~~
    k. ~~Domain Name System~~
    l. ~~Elastic Beanstalk~~
    m. ~~CloudFormation~~
    n. ~~Architecting for the Cloud Best Practices: Part 1~~
    o. ~~Architecting for the Cloud Best Practices: Part 2~~
    p. ~~Global and On Premises AWS Services~~
    q. ~~CloudWatch 101~~
    r. ~~Systems Manager~~
    s. ~~How AWS Pricing Works Whitepaper~~
    t. ~~EC2 Pricing~~
    u. ~~AWS Budgets vs AWS Cost Explorer~~
    v. ~~AWS Support Plans~~
    w. ~~Tagging and Resource Groups~~
    x. ~~AWS Organizations & Consolidated Billing~~
    y. ~~AWS Calculators~~
    z. ~~Compliance On AWS~~
    aa. ~~AWS Web Application Firewall (WAF) & AWS Shield~~
    bb. ~~AWS Inspector vs AWS Trusted Advisor vs CloudTrail~~
    cc. ~~CloudWatch vs AWS Config~~
    dd. ~~Athena vs Macie~~
    ee. ~~AWS Shared Responsibility Model~~
4. ~~Better compact well architected framework best practices~~
5. ~~Ensure no work lost in document corruption~~
6. ~~Fill in 'To Be Added' section~~

*Discarded*

1. ~~What are pain points?~~
2. ~~POC proof of concept~~
3. ~~Polyglot Development~~
4. ~~What is a stack?~~
5. ~~MVP minimum viable product~~
6. ~~Real-Time Messaging Protocol (RTMP) – protocol for streaming audio, video and data over the Internet, between a Flash player and a server. Used in CloudFront~~
7. ~~SOAP APIs – Simple object access protocol~~
8. ~~What is Security as Code~~
9. ~~Questions need verified that answers are right and explained (up to #34 done so far)(scrapped this)~~
10. ~~Routing algorithms / Round robin, X-forwarded for header; Routine load balancer error messages / general error messages~~
11. ~~HTTP Request types~~
12. ~~AWS Pinpoint Campaigns~~
13. ~~APN Campaigns / APN Sponsorships~~
14. ~~Apn partner central~~
15. ~~Partner development manager~~
16. ~~Partner development representative~~
17. ~~data in use encryption~~
18. ~~aws global reach~~
19. ~~Elastic network interfaces~~
20. ~~point of presence~~
21. ~~appspec file~~
22. ~~lifecycle event hooks~~
23. ~~Lambda for scaling up & down vs autoscaling~~
24. ~~Request for Proposal~~
25. ~~Methods of attending instructor based training~~
26. ~~Make acronyms for design principles and best practices~~

# Change Log

V1.0 – Mon 02 August 2021 – Most recent updates completed