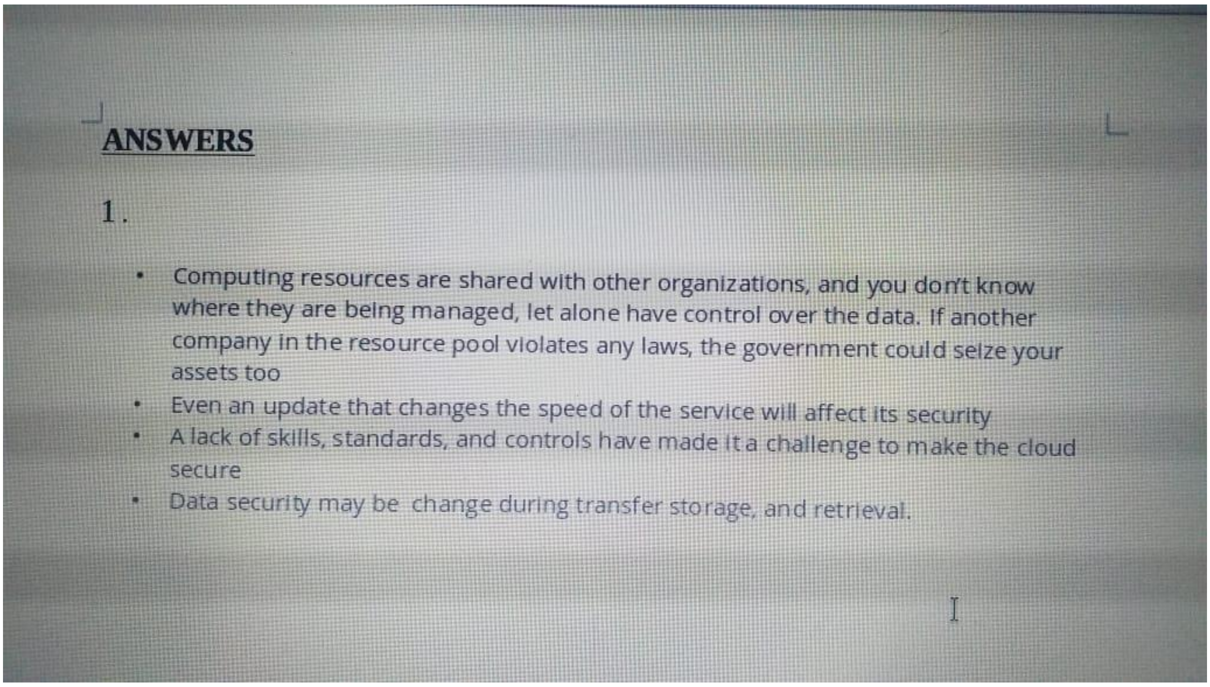
# Part a

1.

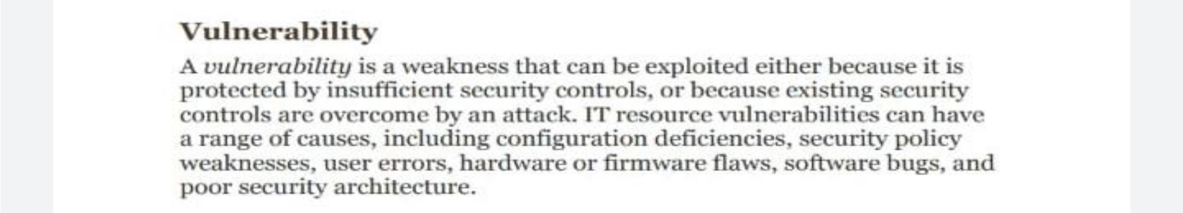


**ANSWERS**

1.

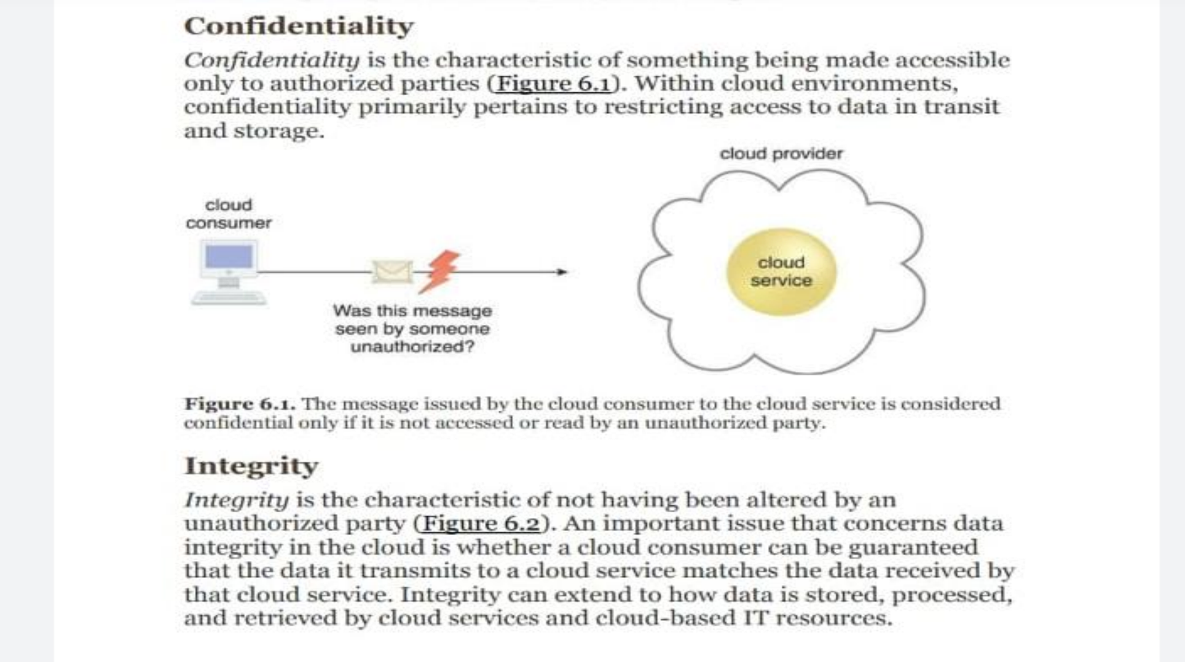
- Computing resources are shared with other organizations, and you don't know where they are being managed, let alone have control over the data. If another company in the resource pool violates any laws, the government could seize your assets too
- Even an update that changes the speed of the service will affect its security
- A lack of skills, standards, and controls have made it a challenge to make the cloud secure
- Data security may be  change during transfer storage, and retrieval.

2



### Vulnerability

A *vulnerability* is a weakness that can be exploited either because it is protected by insufficient security controls, or because existing security controls are overcome by an attack. IT resource vulnerabilities can have a range of causes, including configuration deficiencies, security policy weaknesses, user errors, hardware or firmware flaws, software bugs, and poor security architecture.



### Confidentiality

*Confidentiality* is the characteristic of something being made accessible only to authorized parties (Figure 6.1). Within cloud environments, confidentiality primarily pertains to restricting access to data in transit and storage.

cloud provider
cloud consumer
cloud service
Was this message seen by someone unauthorized?

**Figure 6.1.** The message issued by the cloud consumer to the cloud service is considered confidential only if it is not accessed or read by an unauthorized party.

### Integrity

*Integrity* is the characteristic of not having been altered by an unauthorized party (Figure 6.2). An important issue that concerns data integrity in the cloud is whether a cloud consumer can be guaranteed that the data it transmits to a cloud service matches the data received by that cloud service. Integrity can extend to how data is stored, processed, and retrieved by cloud services and cloud-based IT resources.

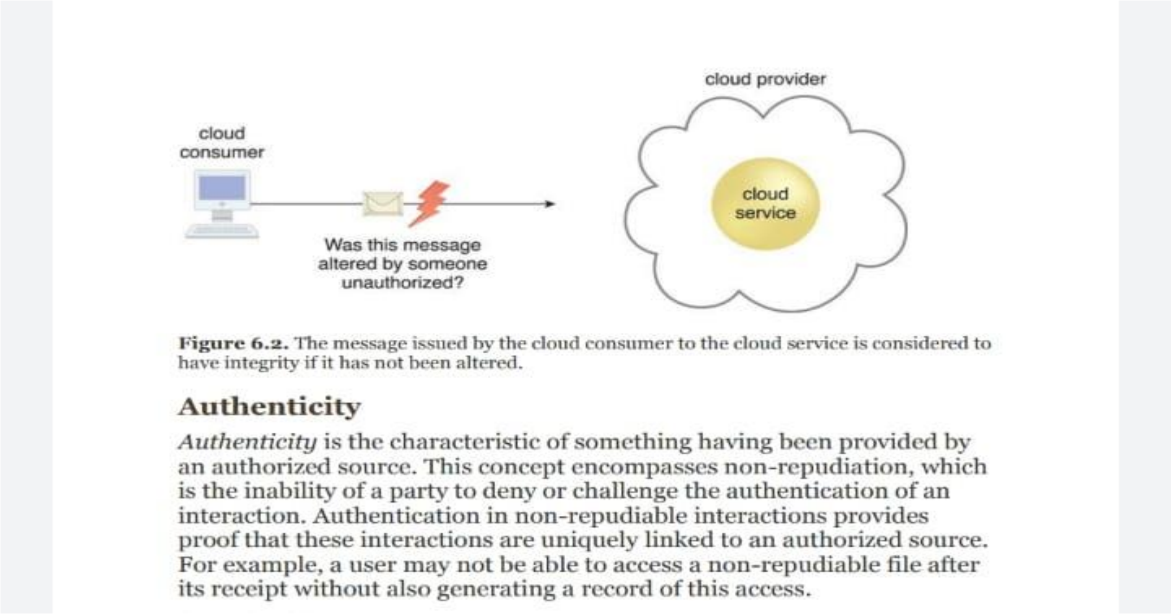
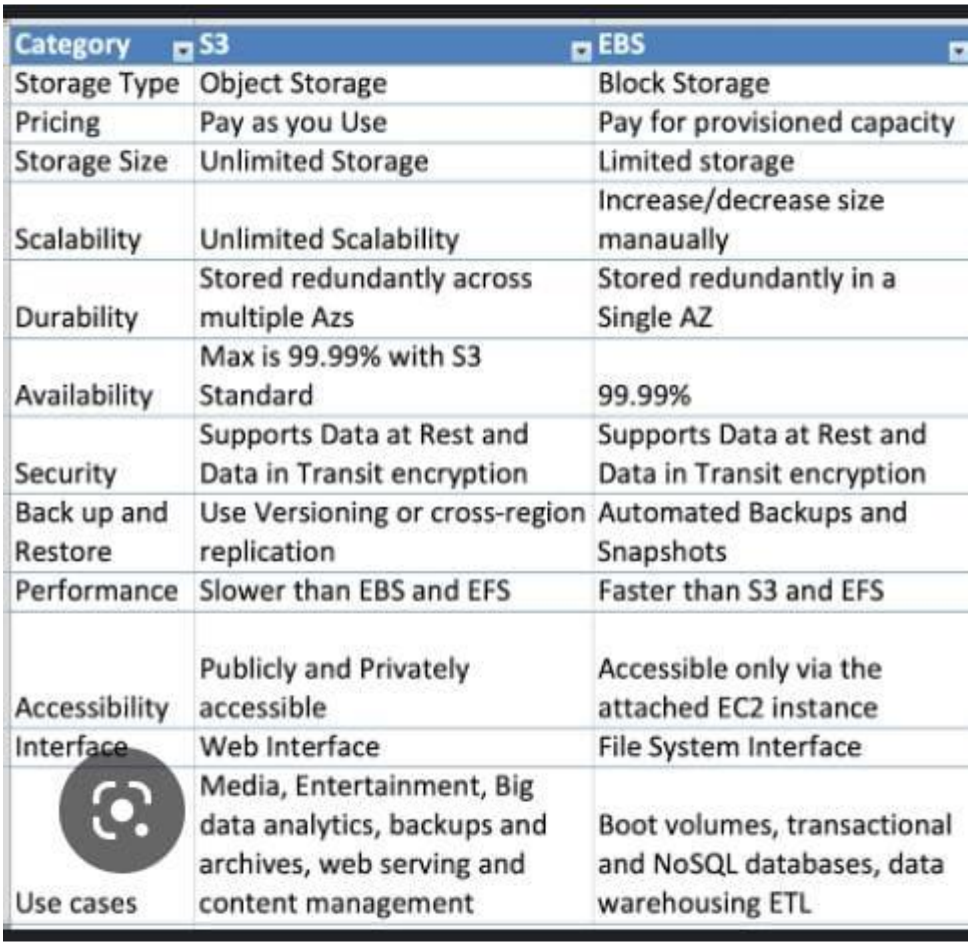

cloud provider
cloud consumer
cloud service
Was this message altered by someone unauthorized?

**Figure 6.2.** The message issued by the cloud consumer to the cloud service is considered to have integrity if it has not been altered.

### Authenticity

*Authenticity* is the characteristic of something having been provided by an authorized source. This concept encompasses non-repudiation, which is the inability of a party to deny or challenge the authentication of an interaction. Authentication in non-repudiable interactions provides proof that these interactions are uniquely linked to an authorized source. For example, a user may not be able to access a non-repudiable file after its receipt without also generating a record of this access.

3



| Category | S3 | EBS |
|---|---|---|
| Storage Type | Object Storage | Block Storage |
| Pricing | Pay as you Use | Pay for provisioned capacity |
| Storage Size | Unlimited Storage | Limited storage |
| Scalability | Unlimited Scalability | Increase/decrease size manaually |
| Durability | Stored redundantly across multiple Azs | Stored redundantly in a Single AZ |
| Availability | Max is 99.99% with S3 Standard | 99.99% |
| Security | Supports Data at Rest and Data in Transit encryption | Supports Data at Rest and Data in Transit encryption |
| Back up and Restore | Use Versioning or cross-region replication | Automated Backups and Snapshots |
| Performance | Slower than EBS and EFS | Faster than S3 and EFS |
| Accessibility | Publicly and Privately accessible | Accessible only via the attached EC2 instance |
| Interface | Web Interface | File System Interface |
| Use cases | Media, Entertainment, Big data analytics, backups and archives, web serving and content management | Boot volumes, transactional and NoSQL databases, data warehousing ETL |

## Part b

Module3

6A

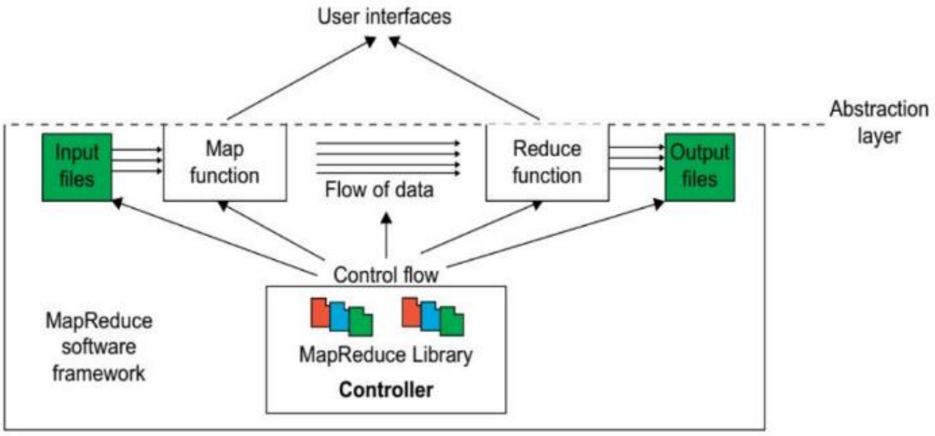
# MapReduce

MapReduce, as introduced  is a **software framework** which supports parallel and distributed computing on large data sets

It abstracts the data flow of running a parallel program on a distributed computing system by providing users with **two interfaces** in the form of two functions: **Map and Reduce.**

Users can **override** these two functions to interact with and manipulate the data flow of running their programs.

In this framework the **"value"** part of the data, (key, value), is **the actual data,** and the **"key"** part is only used by the MapReduce controller **to control the data flow.**



# MapReduce Logical Data Flow

The input data to the Map function is in the form of a (key, value) pair.

The key is the line offset within the input file and the value is the content of the line.

The output data from the Map function is structured as (key, value) pairs called **intermediate (key, value) pairs.**

The user-defined Map function processes each input (key, value) pair and produces a number of **(zero, one, or more) intermediate (key, value) pairs.**

The goal is to process all input (key, value) pairs to the Map function in parallel

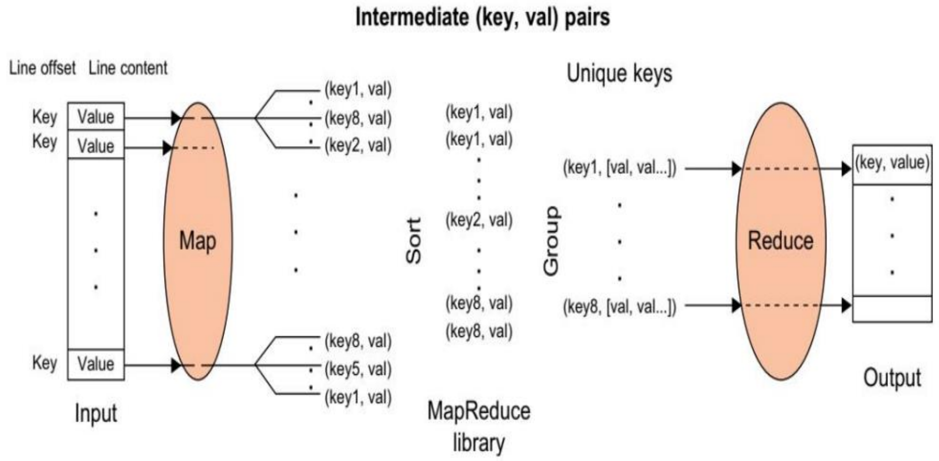
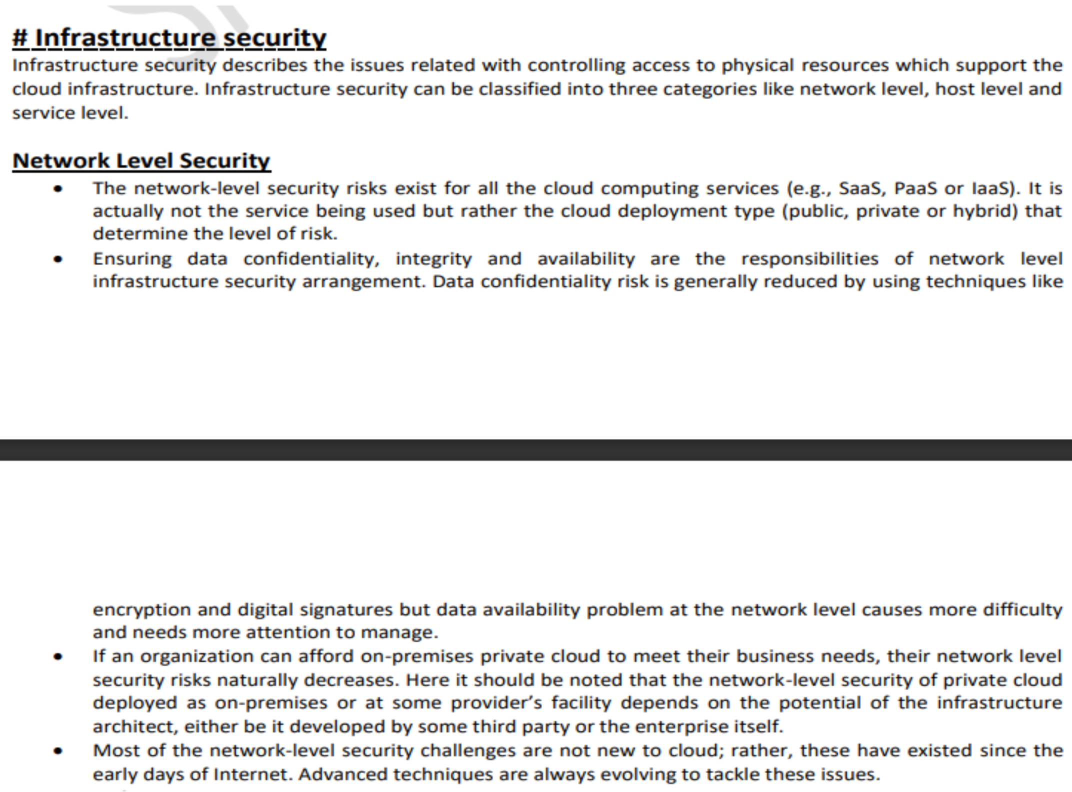


**FIGURE 6.2**

MapReduce logical data flow in 5 processing stages over successive (key, value) pairs.

7.



# Infrastructure security

Infrastructure security describes the issues related with controlling access to physical resources which support the cloud infrastructure. Infrastructure security can be classified into three categories like network level, host level and service level.

## Network Level Security

- The network-level security risks exist for all the cloud computing services (e.g., SaaS, PaaS or IaaS). It is actually not the service being used but rather the cloud deployment type (public, private or hybrid) that determine the level of risk.
- Ensuring data confidentiality, integrity and availability are the responsibilities of network level infrastructure security arrangement. Data confidentiality risk is generally reduced by using techniques like encryption and digital signatures but data availability problem at the network level causes more difficulty and needs more attention to manage.
- If an organization can afford on-premises private cloud to meet their business needs, their network level security risks naturally decreases. Here it should be noted that the network-level security of private cloud deployed as on-premises or at some provider's facility depends on the potential of the infrastructure architect, either be it developed by some third party or the enterprise itself.
- Most of the network-level security challenges are not new to cloud; rather, these have existed since the early days of Internet. Advanced techniques are always evolving to tackle these issues.

## Host Level Security

At cloud service provider's end, the 'host' refers to the physical machines. weak implementation of access control mechanism to the hypervisor may create trouble for physical hosts. VM escape problem may also cause damage to physical hosts as virtual machines are a little prone to this particular security threat as associated to virtualization technology.

The responsibilities of the host-level security management are:

- *For SaaS and PaaS consumers*: Service providers would not publicly share details regarding their host platforms like operating systems or security management mechanisms to secure the hosts. Otherwise, hackers may exploit those details to break the security.
- One difference between PaaS and SaaS consumers arises from the difference in access right to the abstraction layer that covers the OS on which applications they run. This abstraction layer is not accessible by the SaaS consumers but is accessible by the developers who are actually the PaaS consumers. But, the PaaS users cannot access this abstraction layer directly; rather they are given indirect access to the layer through the application program interface (API) of PaaS application.
- In general, the security responsibility of hosts in PaaS and SaaS services largely depends on the service providers.
- *For IaaS consumers*: Unlike PaaS and SaaS, IaaS consumers have the shares of responsibility in securing the host. Service providers use to take care of the security of physical resources through abstraction. But IaaS consumers must take care that no malicious application could try to break it.

## Application Level Security

Both the consumer and service providers have their share of responsibilities of security management at this level so that no application can harm to the infrastructure.

- **IaaS Application Security:** At IaaS level, the users are largely accountable for managing and securing the virtual servers they work with, along with the providers. At this level, the virtual servers (which are delivered by IaaS service providers) are owned by customers, and the IaaS providers blindly serve the applications running over those virtual servers with full trust without verifying any threats. Therefore, the major responsibility of security management of virtual resources at this layer is task of consumers as well.
- **PaaS Application Security:** The security issues can be divided into two stages at the PaaS application level: Security of the PaaS platform itself and Security of consumers' applications deployed on a PaaS application. PaaS service providers are responsible for securing the platform software stack on which consumers deploy or develop their applications. Security management of these applications deployed on PaaS is consumer's prime responsibility, although PaaS providers take care of any kind of dependencies.
- **SaaS Application Security:** In SaaS model, it is the responsibility of the provider to manage the complete set of applications they deliver to consumers. Therefore, the SaaS providers must take suitable measures to make their offering secure so that consumers with ill intention cannot cause harm to them. From the consumer's viewpoint, the use of SaaS reduces lots of tensions.

Module5

8a

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers and system administrators.

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 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers and system administrators the tools to build failure resilient applications and isolate themselves from common failure scenarios.

Auto Scaling: Auto Scaling allows you to scale your Amazon EC2 capacity up or down automatically according to conditions you define. With Auto Scaling, you can ensure that the number of Amazon EC2 instances you're using increases seamlessly during demand spikes to maintain performance, and decreases automatically during demand lulls to minimize costs.

Elastic Load Balancing: Elastic Load Balancing automatically distributes incoming application traffic across multiple Amazon EC2 instances. It enables you to achieve even greater fault tolerance in your applications, seamlessly providing the amount of load balancing capacity needed in response to incoming application traffic.