KodeKloud

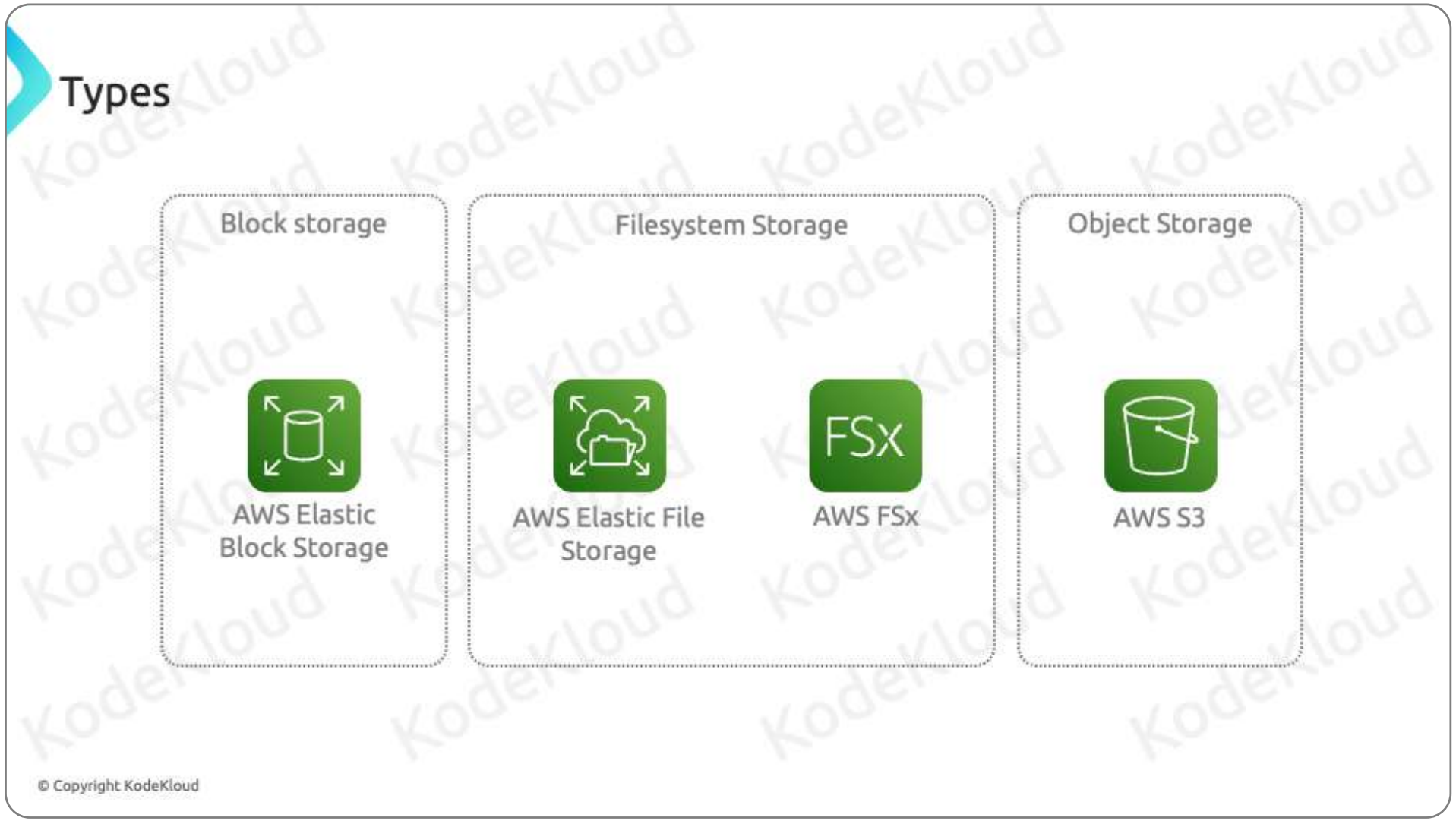
In this section we are going to be talking about storage and the different storage services provided by amazon.

Storage is all about persistent data that your application needs to use. When it comes to a storage solution there’s a variety of different features we may be interested in

Things to look for in storage solution:

- High availability – is the data replicated across multiple physical devices
    - Is data replicated across different Availability zones and/or regions
- Can the storage automatically scale up in size
- Can we adjust storage size later on if we need more?
- At what speed can we write data to the storage solution

- Can we boot & mount from the storage device
- Can we connect to the storage device over the network
- Can multiple devices/users connect to a storage device at the same time and simultaneously make changes?



There’s three main types of storage solutions and amazon has services for each storage type

Block storage
- Elastic Block Storage

Filesystem Storage
- Elastic Filesystem Storage

- FSx

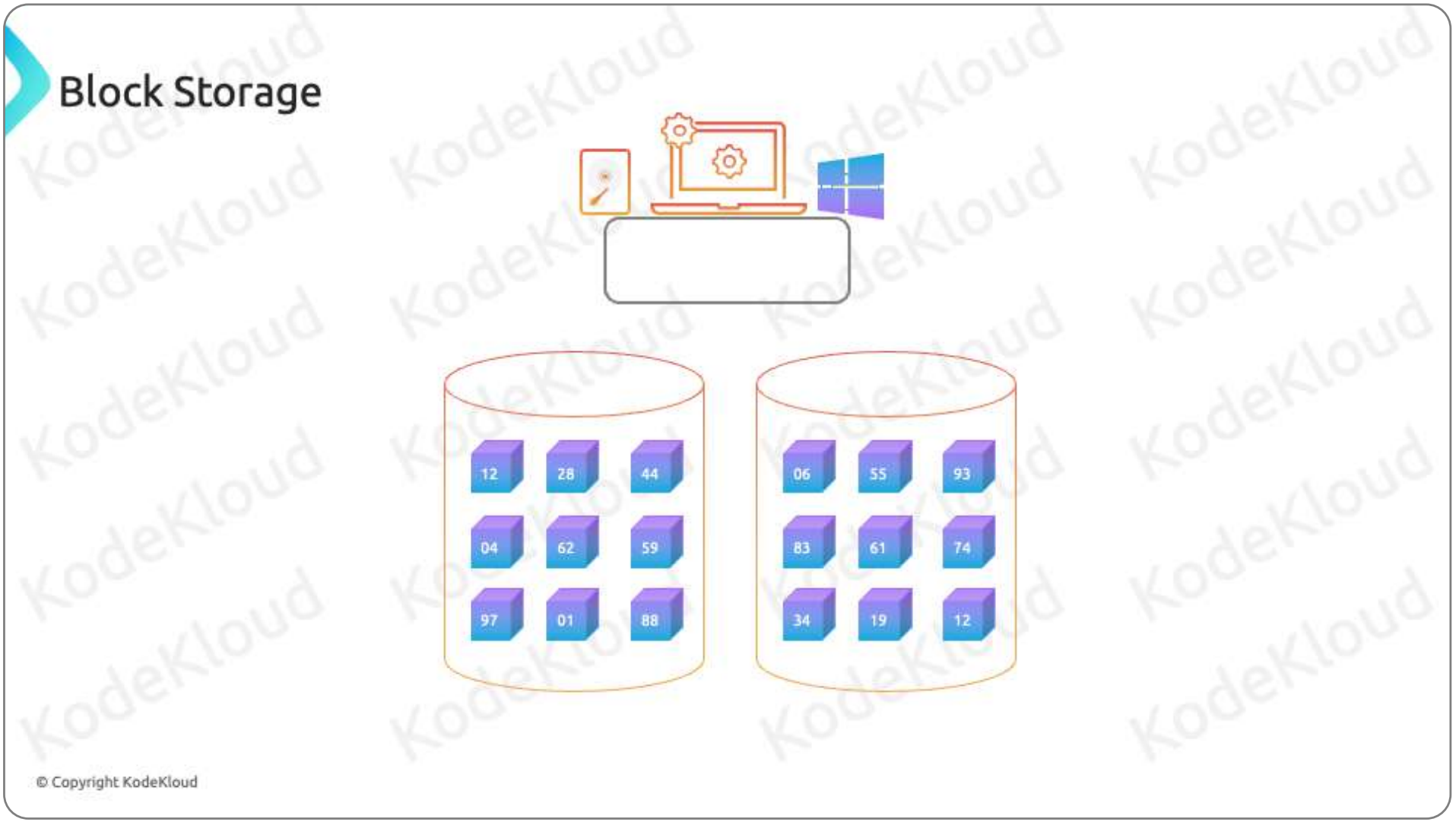
Object Storage

- S3

As a solutions architect, it's important to understand when we would pick one type of storage solution over the other. So, in this section ,we will go over the different features of each storage service and talk about the pros/cons of each one and when we would pick one over the other

Welcome to the section on Cloud Computing.

What we learned in Cloud Practitioner



Block storage breaks up data into block and then stores these block as sperate pieces each with a unique identifier. These blocks can be stored across a number of physical devices.

A collection of block can be presented to your operating system as a volume.

We can then take the volume that is presented to the OS and create a filesystem ontop of it.

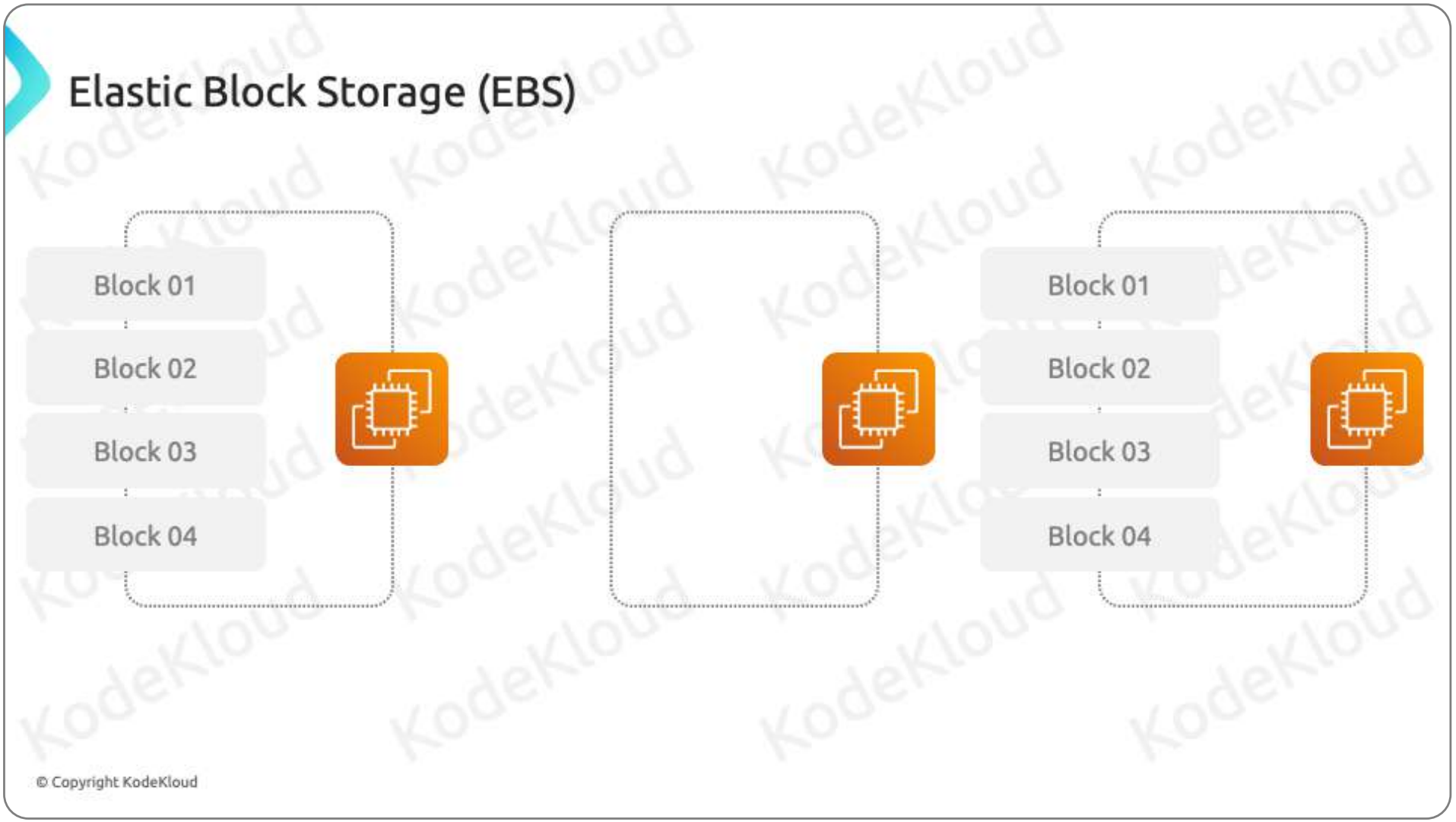
The great part about block storage is not only can we present it as a volume to the OS but we can also present it as a hard drive, which allows us to install an operating system and make it bootable, so you can install an operating system on the block device.

So with block storage can be presented as a volume so you can create a filesystem or you can present it as a hard drive bootable.

So block storage is both mountable and bootable

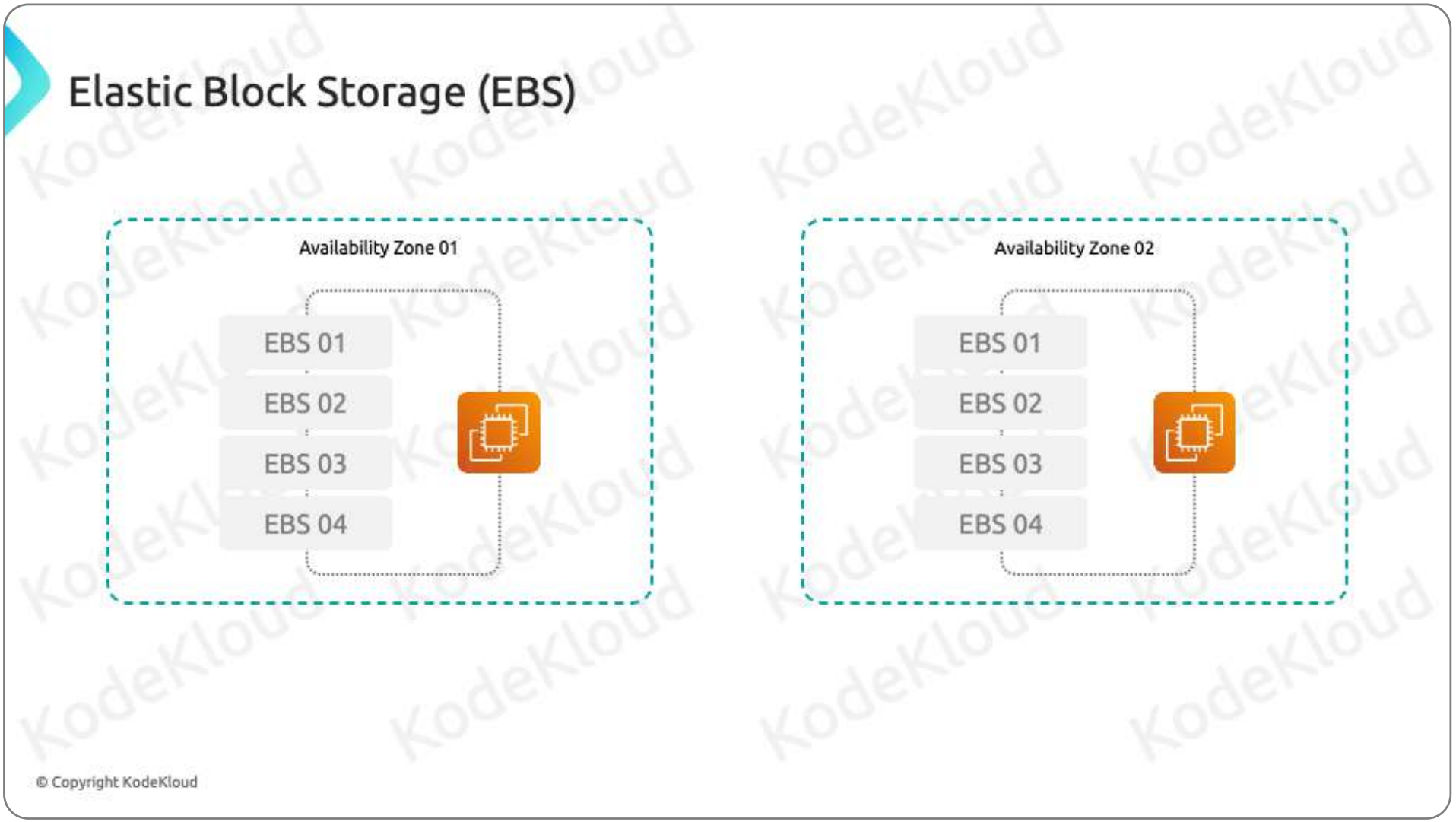
Amazon has a block storage service called elastic Block storage.

Elastic Block Storage



- So Elastic Block storage(EBS) provides block level storage volumes for use with EC2 instances The ec2 instances will see the attached block device and from there you can create a filesystem ontop of the device xfs/ext3/ext4
- The great part of Elastic block stores is that they are separate from ec2 instances. So we can attch it to an ec2 instance, then detach it and attach it to another ec2 instance. So the data is separate from the lifecycle of an ec2 instance
- EBS volumes can normally only attach to a single ec2 instance. However certain volume types allow multi-attach where multiple instances can attach to the same volume. However if you do this, your application must be intelligent enough to

not have multiple ec2 instances write to the same data at the same time as that would corrupt the data. For example database clusters are configured to only have one instance write to a volume at a given time to prevent this issue

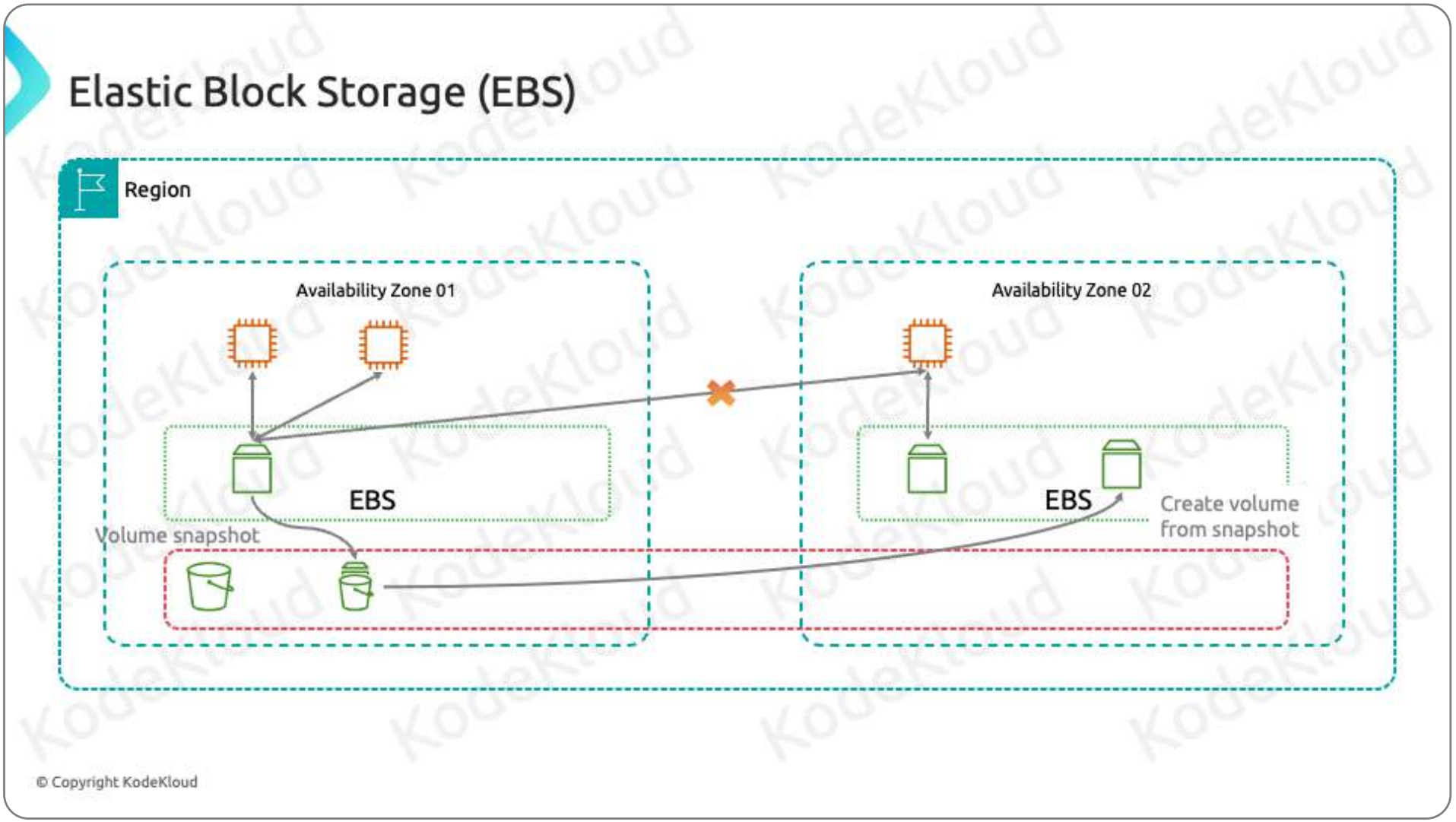


An elastic block storage volume is provisioned in an availabilizy zone

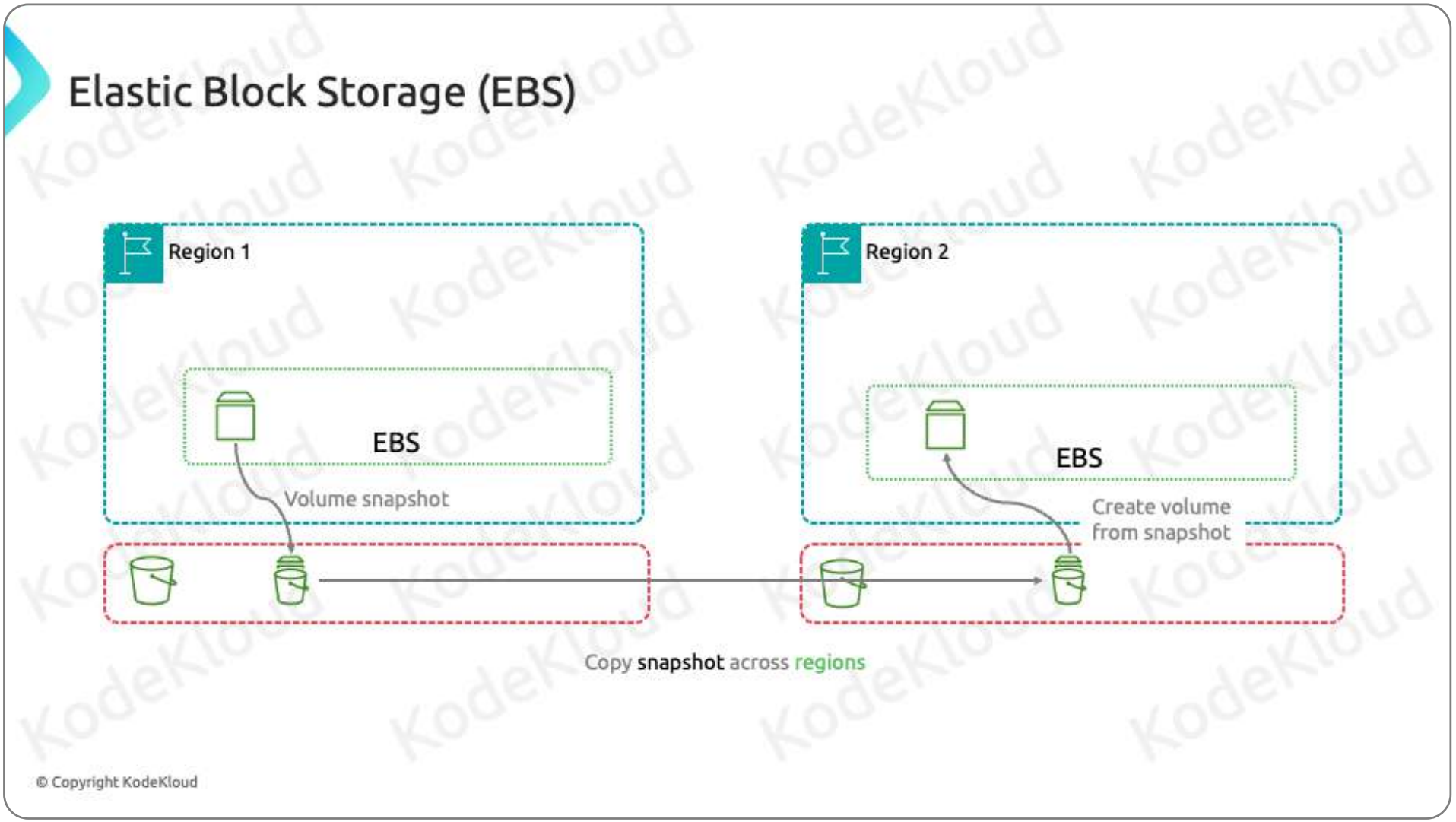
EBS volumes are resilient within an AZ. So it can handle a physical device going down but the data is lost if an entire AZ goes down.

This also means that to attach an ebs volume to an ec2 instance, both the ec2 instance and ebs volume must be in the same

AZ



So let’s take a look at how ebs works



So how do we replicate data from one region to another?

Well we take a snapshot and then we copy the snapshot from one region to another

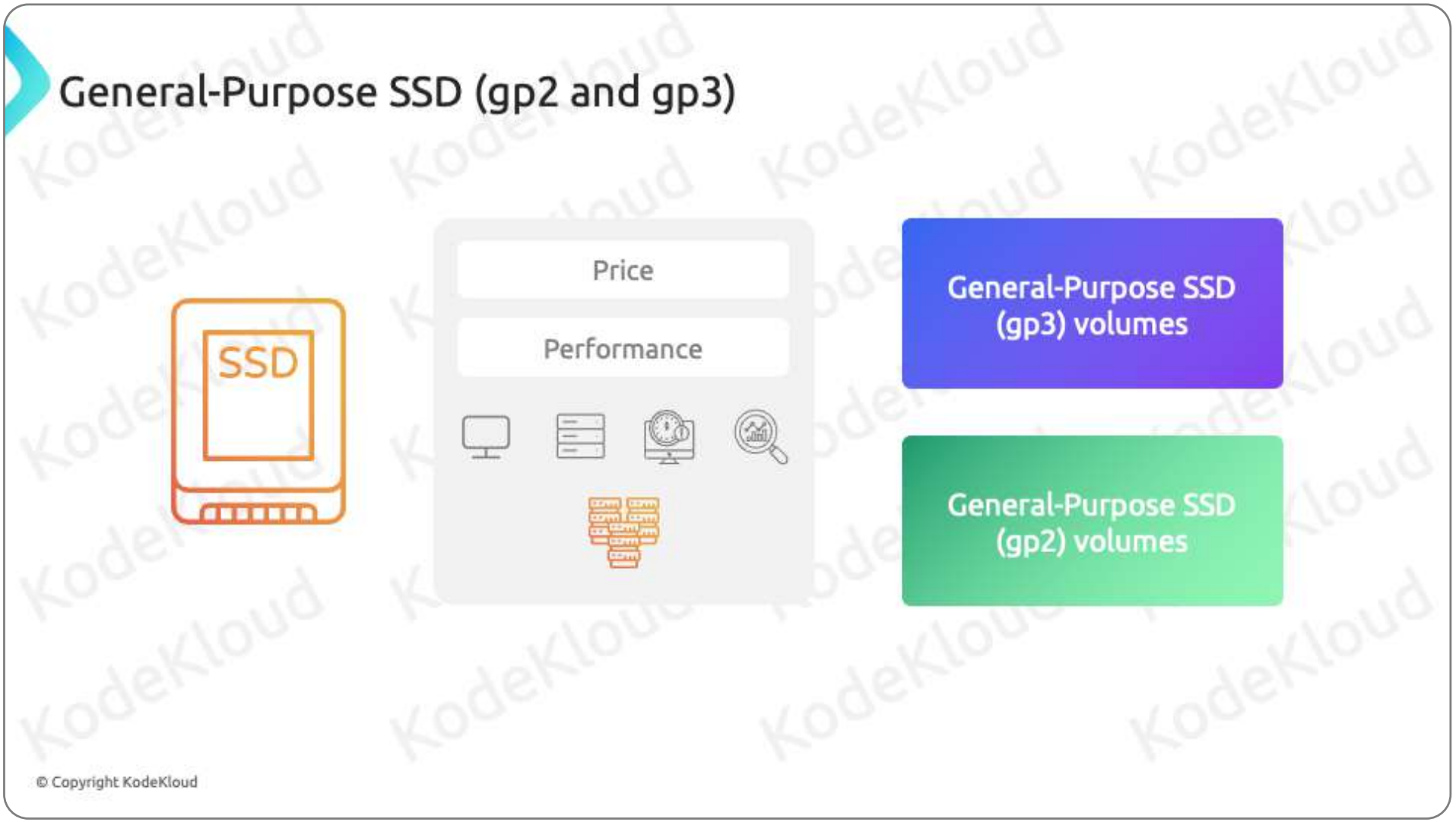
And then from the second region we can deploy a volume from the snapshot copy in that region

EBS supports multiple volume type

Amazon EBS provides the following volume types, which differ in performance characteristics and price, so that you can tailor your storage performance and cost to the needs of your applications

- General purpose SSD gp2/gp3
- Provisioned IOPS SSD volumes

- Throughput Optimized HDD volumes
- Cold HDD volumes
- Magnetic



General Purpose SSD (gp2 and gp3) volumes are backed by solid-state drives (SSDs).

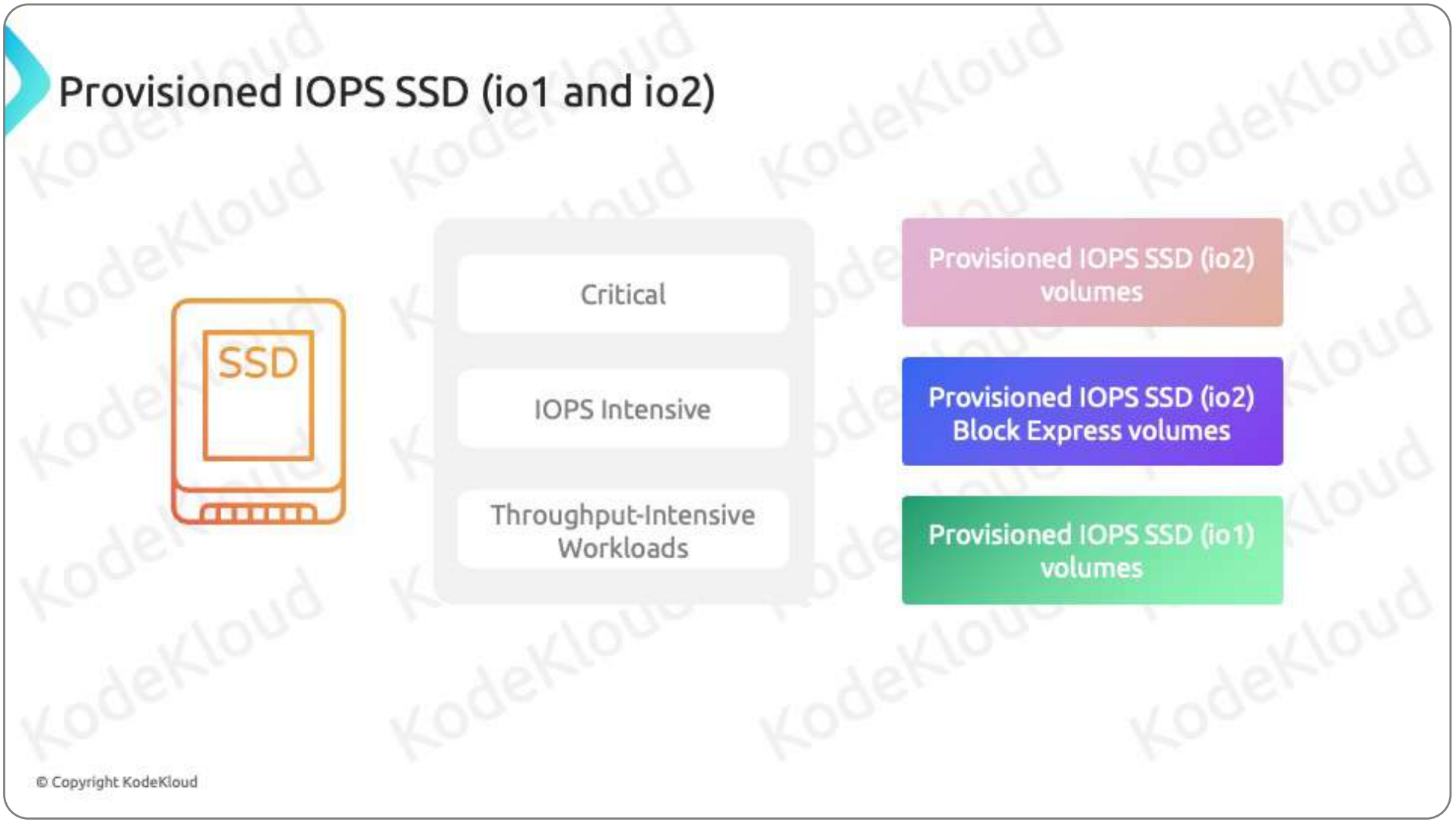
They balance price and performance for a wide variety of transactional workloads

These include virtual desktops, medium-sized single instance databases, latency sensitive interactive applications, development and test environments, and boot volumes. We recommend these volumes for most workloads.

Amazon EBS offers two types of General Purpose SSD volumes:
•General Purpose SSD (gp3) volumes—latest generation General Purpose SSD volume
•General Purpose SSD (gp2) volumes

General Purpose SSD (gp3) volumes are the latest generation of General Purpose SSD volumes, and the lowest cost SSD volume offered by Amazon EBS. This volume type helps to provide the right balance of price and performance for most applications. It also helps you to scale volume performance independently of volume size. This means that you can provision the required performance without needing to provision additional block storage capacity. Additionally, gp3 volumes offer a 20 percent lower price per GiB than General Purpose SSD (gp2) volumes.

General Purpose SSD (gp2) volumes are the default Amazon EBS volume type for Amazon EC2 instances. They offer cost-effective storage that is ideal for a broad range of transactional workloads. With gp2 volumes, performance scales with volume size.



Provisioned IOPS SSD volumes are backed by solid-state drives (SSDs).

They are the highest performance Amazon EBS storage volumes designed for critical, IOPS-intensive, and throughput-intensive workloads that require low latency.

Amazon EBS offers three types of Provisioned IOPS SSD volumes:

•Provisioned IOPS SSD (io2) volumes
•Provisioned IOPS SSD (io2) Block Express volumes
•Provisioned IOPS SSD (io1) volumes

Provisioned IOPS SSD (io1 and io2) volumes are designed to meet the needs of I/O-intensive workloads, particularly database workloads, that are sensitive to storage performance and consistency. Provisioned IOPS SSD volumes use a consistent IOPS rate, which you specify when you create the volume, and Amazon EBS delivers the provisioned performance 99.9 percent of the time.

io1 volumes are designed to provide 99.8 percent to 99.9 percent volume durability
io2 volumes are designed to provide 99.999 percent volume durability

io2 Block Express volumes is the next generation of Amazon EBS storage server architecture. It has been built for the purpose of meeting the performance requirements of the most demanding I/O intensive applications that run on Nitro-based Amazon EC2 instances.

io2 Block Express volumes are suited for workloads that benefit from a single volume that provides sub-millisecond latency, and supports higher IOPS, higher throughput, and larger capacity than io2 volumes.
io2 Block Express volumes support the same features as io2 volumes, including Multi-Attach and encryption.

SSD Volumes

	General Purpose SSD volumes		Provisioned IOPS SSD volumes		
Volume type	gp3	gp2	io2 Block Express ‡	io2	io1
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)		99.999% durability (0.001% annual failure rate)		99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)
Use cases	• Transactional workloads • Virtual desktops • Medium-sized, single-instance databases • Low-latency interactive applications • Boot volumes • Development and test environments		Workloads that require: • Sub-millisecond latency • Sustained IOPS performance • More than 64,000 IOPS or 1,000 MiB/s of throughput		• Workloads that require sustained IOPS performance or more than 16,000 IOPS • I/O-intensive database workloads
Volume size	1 GiB - 16 TiB		4 GiB - 64 TiB		4 GiB - 16 TiB
Max IOPS per volume (16 KiB I/O)	16,000		256,000		64,000 †
Max throughput per volume	1,000 MiB/s	250 MiB/s *	4,000 MiB/s		1,000 MiB/s †
Amazon EBS Multi-attach	Not supported		Supported		
Boot volume	Supported				

Provisioned IOPS SSD (io1 and io2) volumes are designed to meet the needs of I/O-intensive workloads, particularly database workloads, that are sensitive to storage performance and consistency. Provisioned IOPS SSD volumes use a consistent IOPS rate, which you specify when you create the volume, and Amazon EBS delivers the provisioned performance 99.9 percent of the time.

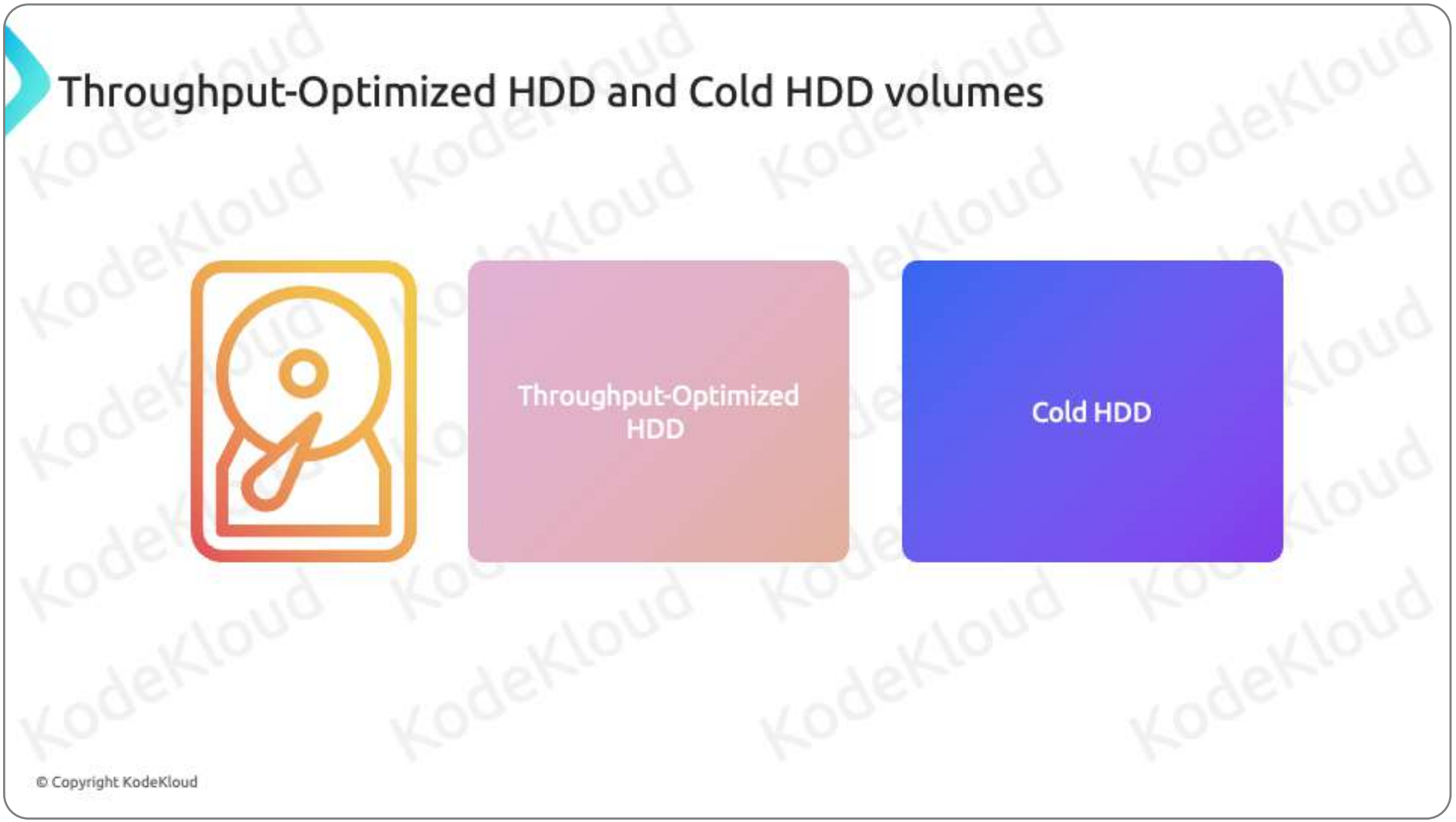
io1 volumes are designed to provide 99.8 percent to 99.9 percent volume durability
io2 volumes are designed to provide 99.999 percent volume durability

io2 Block Express volumes is the next generation of Amazon EBS storage server architecture. It has been built for the purpose of meeting the performance requirements of the most demanding I/O intensive applications that run on Nitro-based Amazon EC2 instances.

io2 Block Express volumes are suited for workloads that benefit from a single volume that provides sub-millisecond latency, and supports higher IOPS, higher throughput, and larger capacity than io2 volumes.
io2 Block Express volumes support the same features as io2 volumes, including Multi-Attach and encryption.

SSD Volumes

	General Purpose SSD volumes		Provisioned IOPS SSD volumes		
Volume type	gp3	gp2	io2 Block Express ‡	io2	io1
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)		99.999% durability (0.001% annual failure rate)		99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)
Use cases	• Transactional workloads • Virtual desktops • Medium-sized, single-instance databases • Low-latency interactive applications • Boot volumes • Development and test environments		Workloads that require: • Sub-millisecond latency • Sustained IOPS performance • More than 64,000 IOPS or 1,000 MiB/s of throughput		• Workloads that require sustained IOPS performance or more than 16,000 IOPS • I/O-intensive database workloads
Volume size	1 GiB - 16 TiB		4 GiB - 64 TiB		4 GiB - 16 TiB
Max IOPS per volume (16 KiB I/O)	16,000		256,000		64,000 †
Max throughput per volume	1,000 MiB/s	250 MiB/s *	4,000 MiB/s		1,000 MiB/s †
Amazon EBS Multi-attach	Not supported		Supported		
Boot volume	Supported				



HDD-backed volumes – HDD or hard disk drives which means they have moving parts, which means they are slower than ssd based volume types

HDD-backed volumes are optimized for large streaming workloads where the dominant performance attribute is throughput.
The HDD-backed volumes provided by Amazon EBS fall into these categories:
•Throughput Optimized HDD — A low-cost HDD designed for frequently accessed, throughput-intensive workloads.

•Cold HDD — The lowest-cost HDD design for less frequently accessed workloads.

HDD Volumes

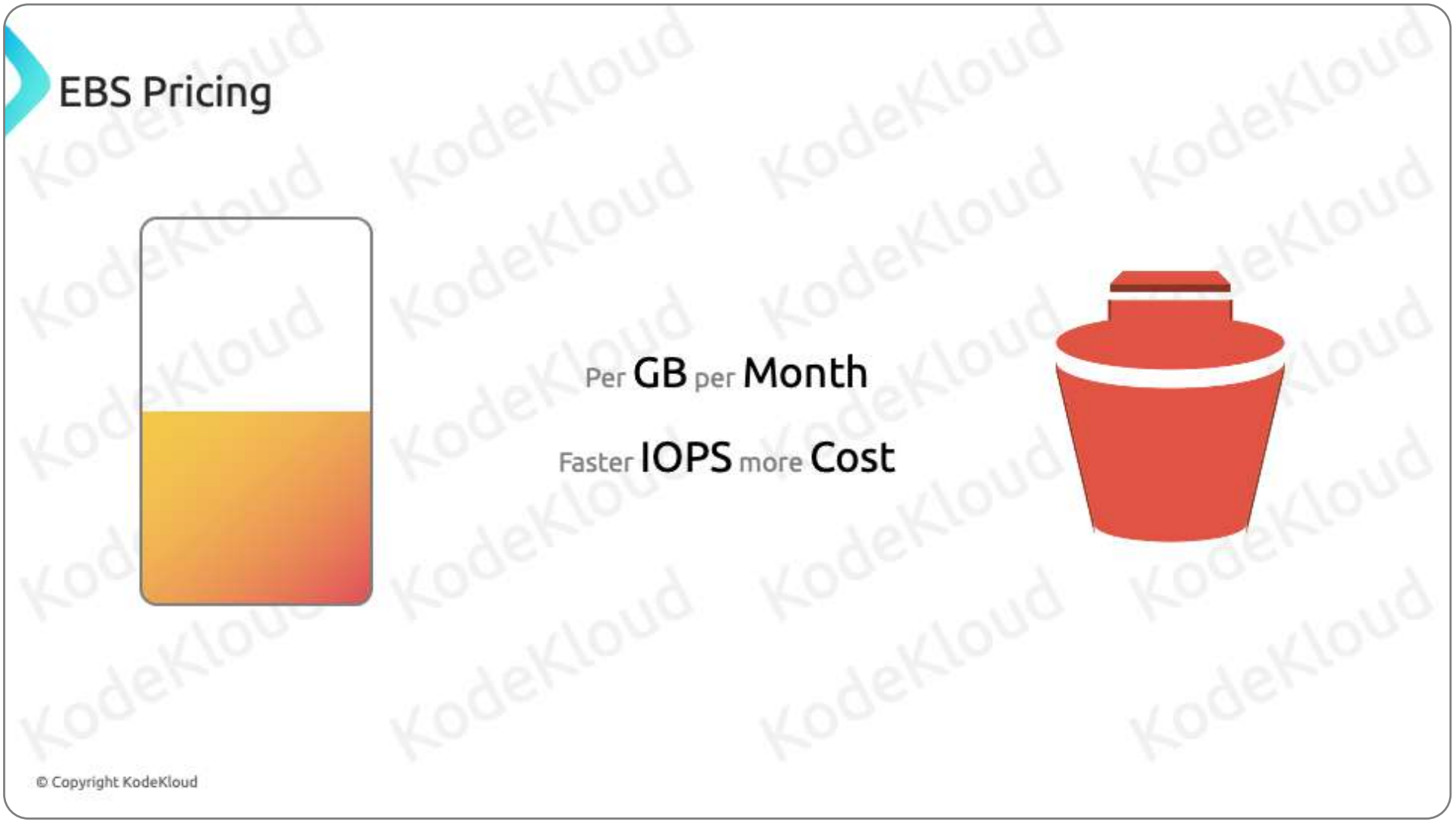
	Throughput Optimized HDD volumes	Cold HDD volumes
Volume type	`st1`	`sc1`
Durability	99.8% - 99.9% durability (0.1% - 0.2% annual failure rate)	
Use cases	• Big data • Data warehouses • Log processing	• Throughput-oriented storage for data that is infrequently accessed • Scenarios where the lowest storage cost is important
Volume size	125 GiB - 16 TiB	
Max IOPS per volume (1 MiB I/O)	500	250
Max throughput per volume	500 MiB/s	250 MiB/s
Amazon EBS Multi-attach	Not supported	
Boot volume	Not supported	

Magnetic Volumes (previous generation)

	Magnetic
Volume type	`standard`
Use cases	Workloads where data is infrequently accessed
Volume size	1 GiB-1 TiB
Max IOPS per volume	40–200
Max throughput per volume	40–90 MiB/s
Boot volume	Supported

Magnetic (standard) volumes are previous generation volumes that are backed by magnetic drives. They are suited for workloads with small datasets where data is accessed infrequently and performance is not of primary importance. These volumes deliver approximately 100 IOPS on average, with burst capability of up to hundreds of IOPS, and they can range in size from 1 GiB to 1 TiB.

EBS supports multiple volume type



With Amazon Elastic Block Store (EBS), you pay only for what you provision. You are charged per Gb per month
So the more gigabytes you provision the more you pay

The cost per GB varies depending on the volume type selected. Volume types with faster IOPS will cost more

EBS snapshots are also charged per gb per month
keep in mind that these are full snapshots and not just incremental snapshots that only track changes. So your paying for full cost

Summary

01 Block storage breaks up data into blocks and then stores these blocks as separate pieces, each with a unique identifier

02 A collection of blocks can be presented to your operating system as a volume

03 You can boot and mount from block storage

04 Elastic Block Storage (EBS) provides block-level storage volumes for use with EC2 instances

Summary

05 EBS volume is provisioned in an availability zone

06 To copy data from an EBS volume to another AZ, you can create an EBS snapshot, followed by creating a volume from snapshot in the desired AZ

07 EBS offers different volume types for different storage needs

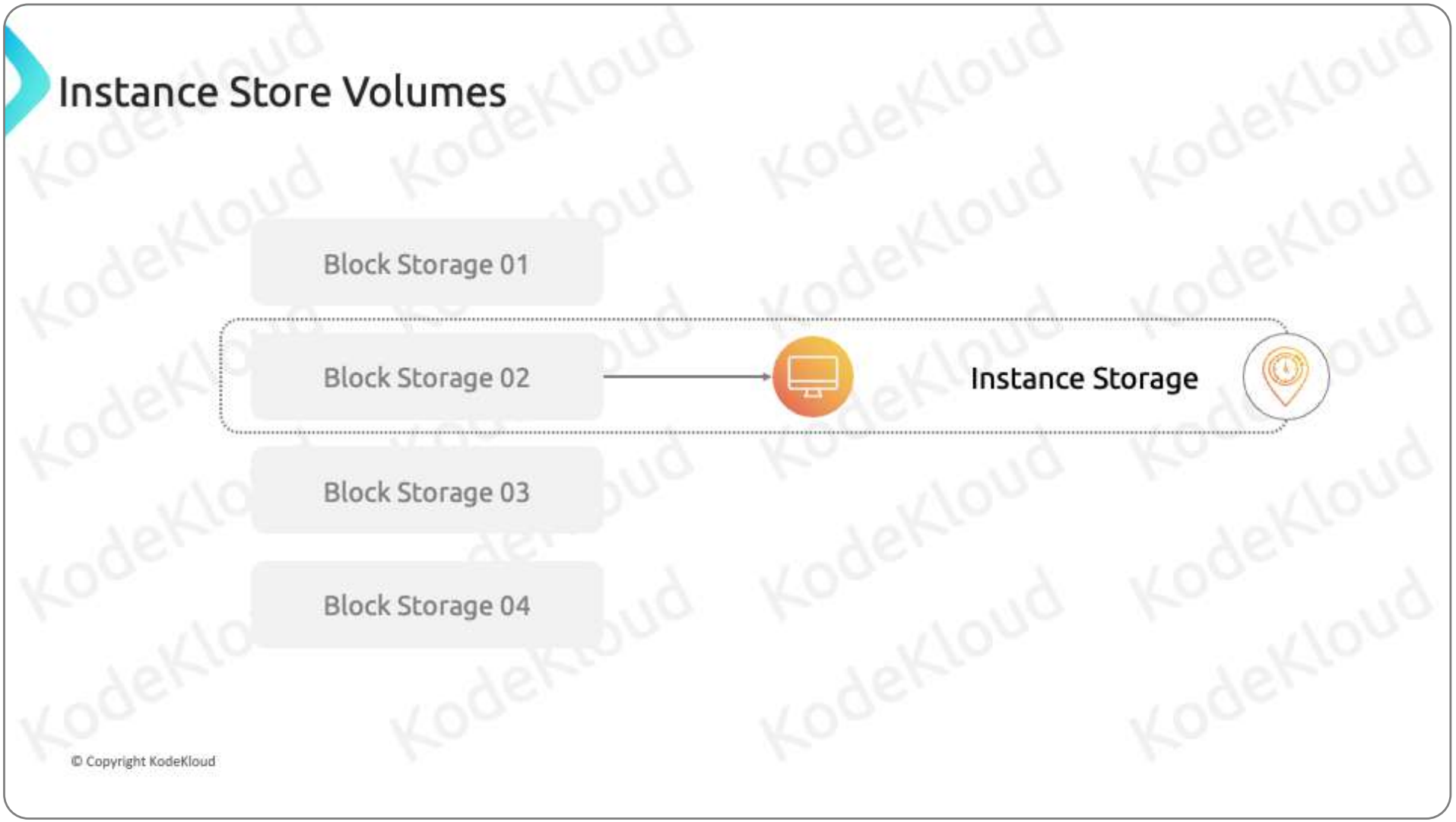
08 You pay for only what you are provisioned – charges are per GB per month

Service Section

The Power of Block Storage

Instance Storage

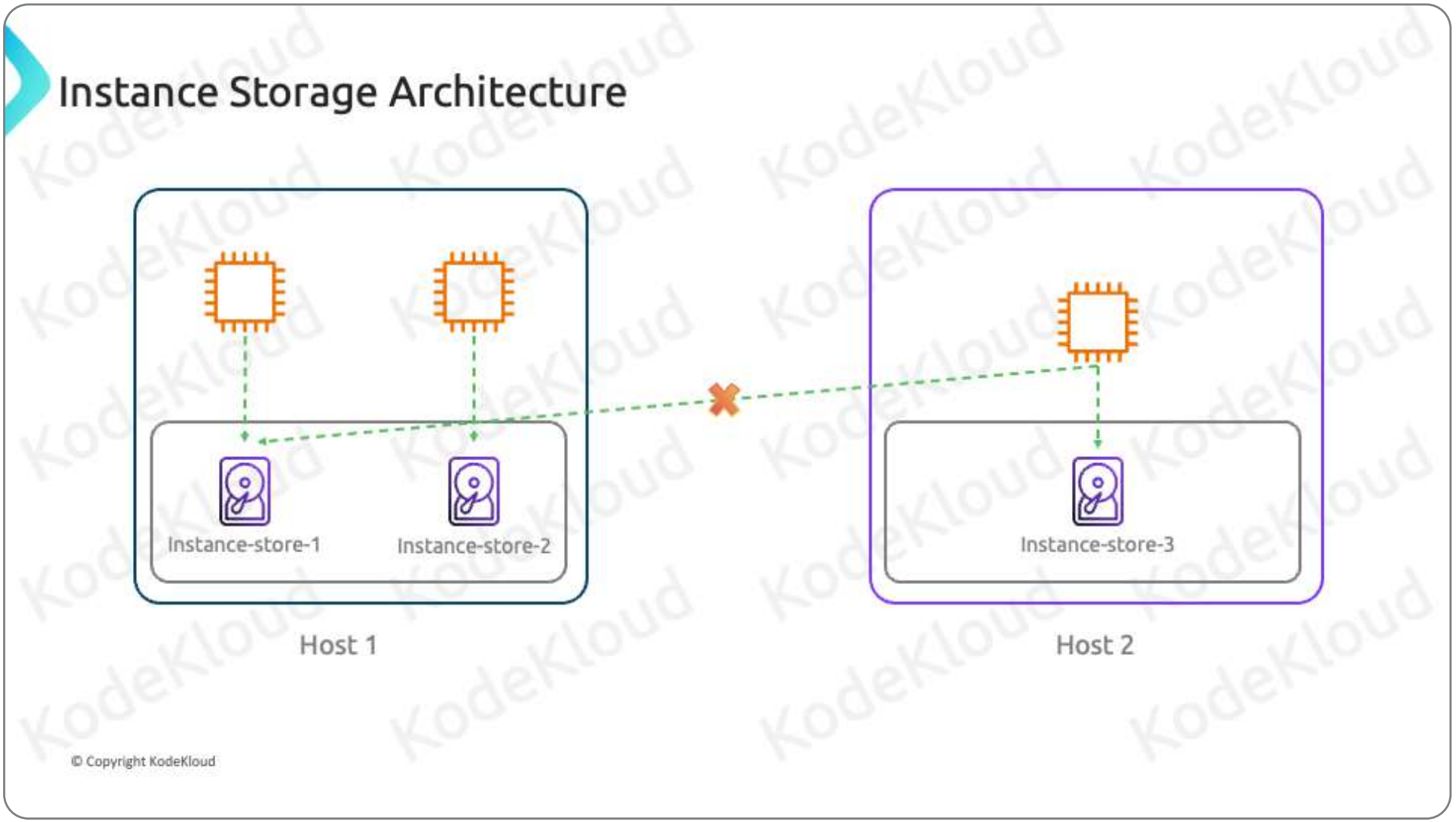
What Is Instance Storage?



An instance store provides temporary block-level storage for your instance. Now we haven't covered what is block storage, but we'll be going over that in the next topic when we cover ebs. But I want you to remember for the exam that instance storage volumes are block-level storage.

Instance storage is located on disks that are physically attached to the host computer.

Instance storage is meant to be used only for temprorary storage or information that changes frequently like temporary content, or scratch data.



As mentioned the instance storage is physically attached ot the host machine that ec2 instance is running on. SO think of it like a hard drive in the server that is running your ec2 instance.

Now if you shut your ec2 instance off and turn it back on, if the ec2 instance remains on the same physical host machine then it will still have access ot the same instance store.

However if after powering down and powering back up it moves to a different host machine. Then it now longer will have access to the original instance store. It will have access to a new instance store located on the new host mahchine host2. So all the previous data is lost.

Summary

01 Instance stores should only be used for temporary data

02 If an EC2 instance is moved from one host to another, then it will lose all the data from the original Instance store

Service Section

The Power of Network File Storage

Elastic File System (EFS)

Understanding Elastic File System (EFS)

Amazons Elastic Filesystem Storage is the first of two filesystem storage services provided by amazon. The other being FSx

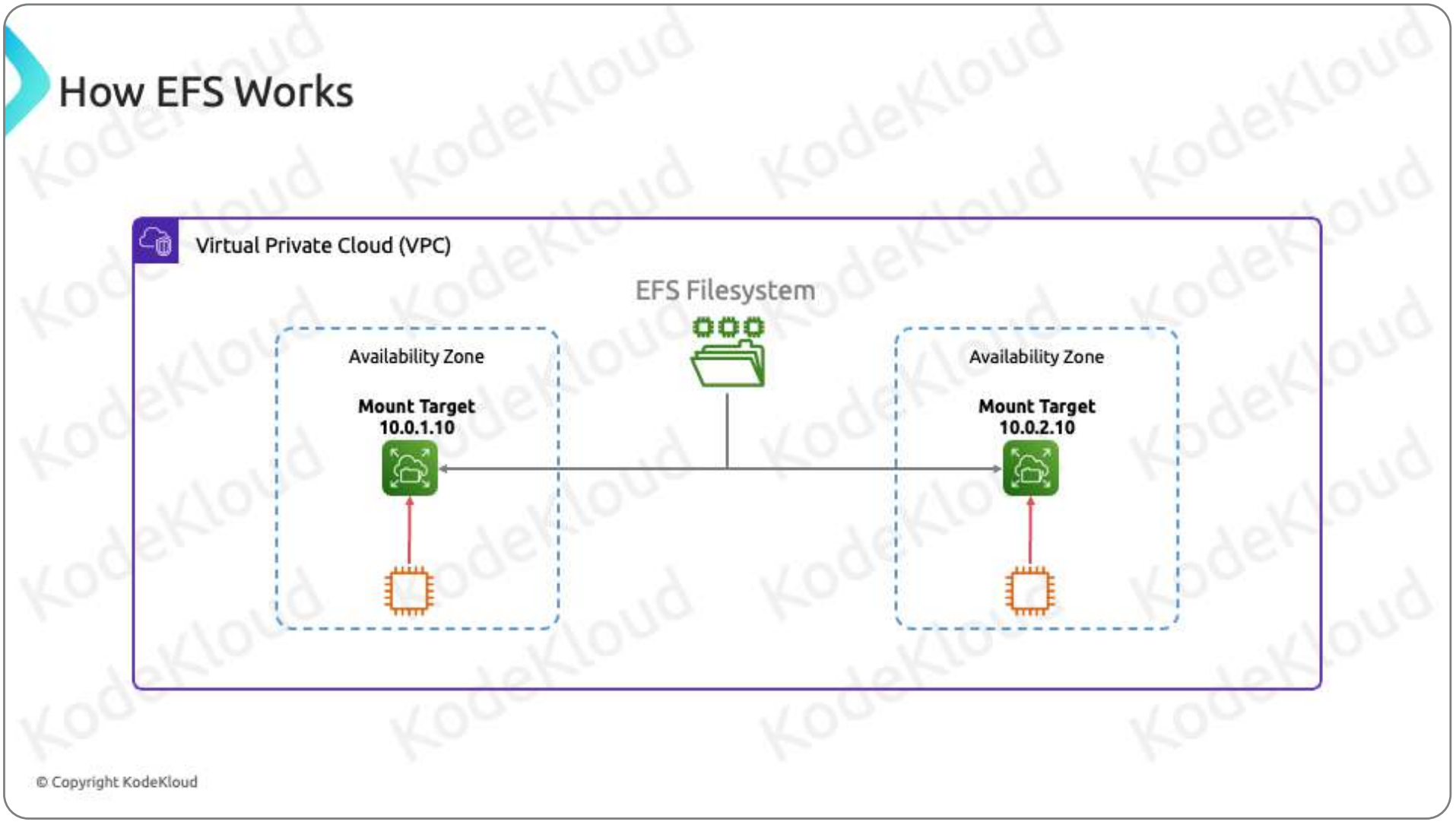
EFS supports the Network File System v4 protocol NFSv4. So applications that use the protocol can seamlessly work with EFS

With EFS you create filesystems and ec2 instances and other computer services can remotely connect to the filesystem and

mount them

EFS does not support windows based ec2 instances only linux

With EFS filesystems, they can be mounted onto multiple ec2 instances at the same time. So the data can be shared across multiple ec2 instances

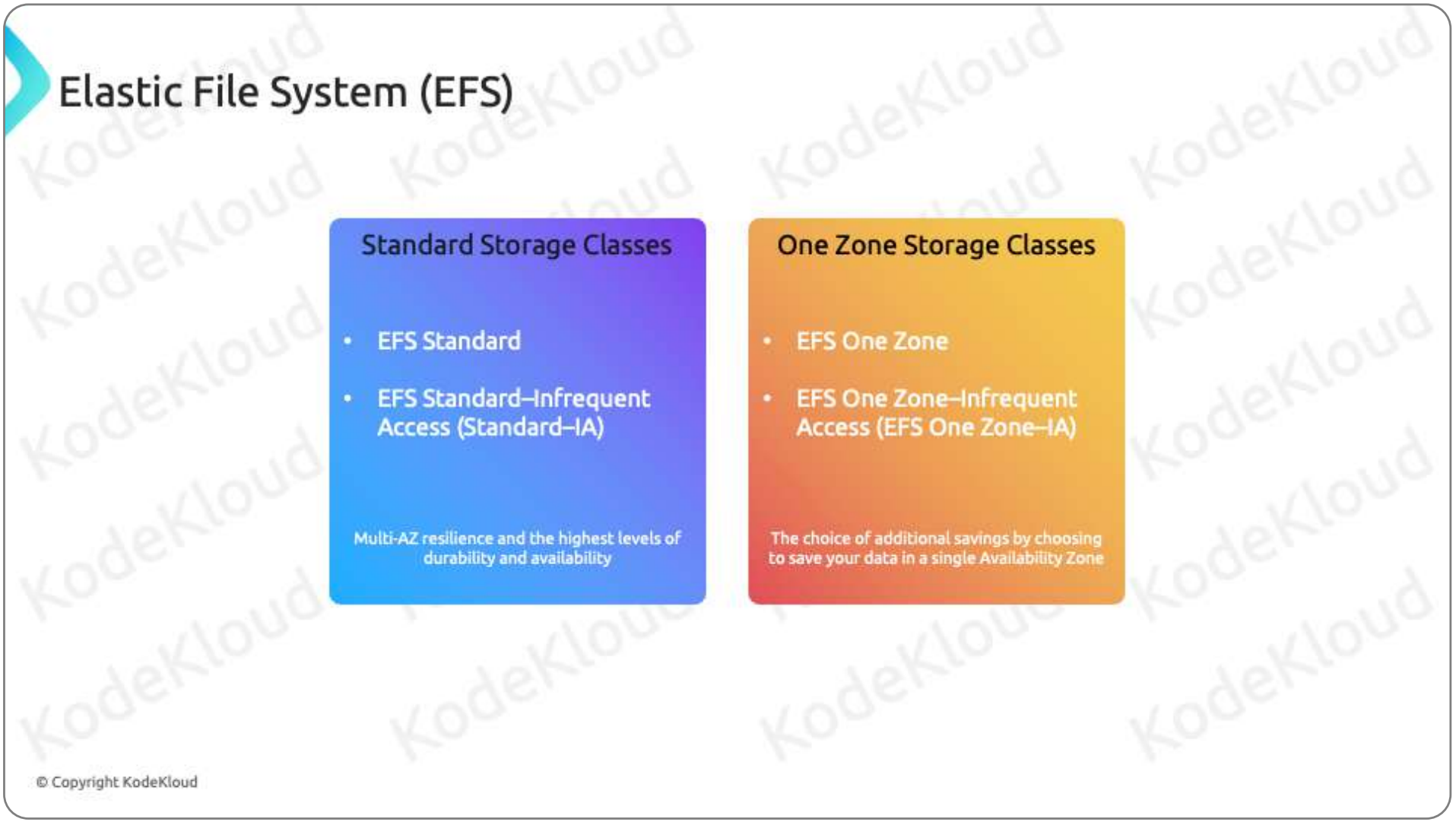


EFS is deployed to a VPC

An EFS filesystem is made available inside a VPC via mount targets. The mount targets get ip addresses from the subnets they are deployed in.

To ensure high availability, you need to make sure you have mount target in multiple availability zones.

Its these mount target that these instances use to connect to the efs filesystem



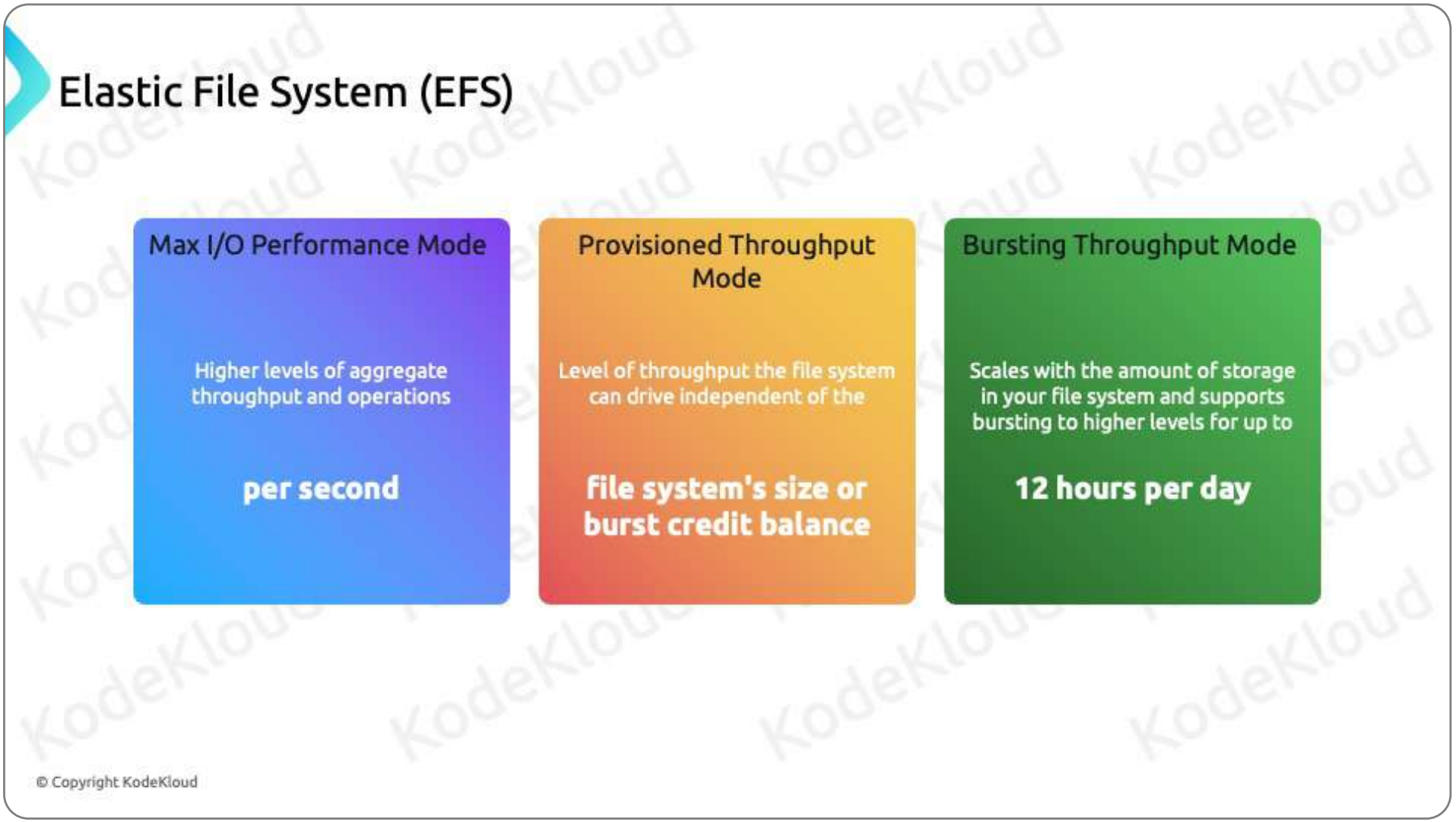
Amazon EFS offers the following storage class options for different use cases:

•**Standard storage classes** (Recommended) – EFS Standard and EFS Standard–Infrequent Access (Standard–IA), which offer Multi-AZ resilience and the highest levels of durability and availability.

•**One Zone storage classes** – EFS One Zone and EFS One Zone–Infrequent Access (EFS One Zone–IA), which offer you the choice of additional savings by choosing to save your data in a single Availability Zone.

Amazon EFS provides the throughput, IOPS, and low latency needed for a broad range of workloads.

- The default *General Purpose performance mode* is ideal for latency-sensitive applications, like web-serving environments, content-management systems, home directories, and general file serving.
- The default *Elastic Throughput mode* is designed to automatically scale throughput performance up or down to meet the needs of your workload activity.

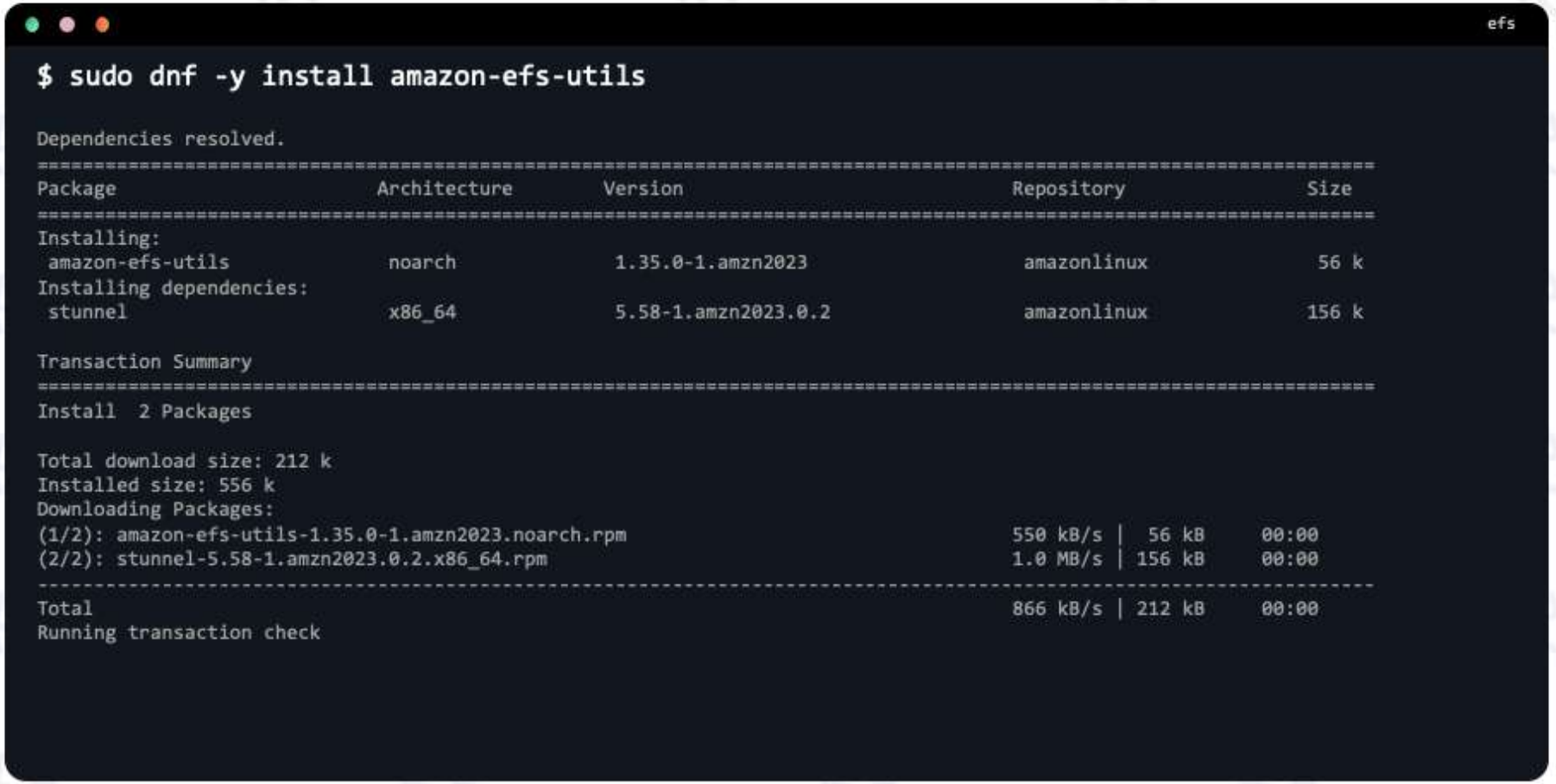


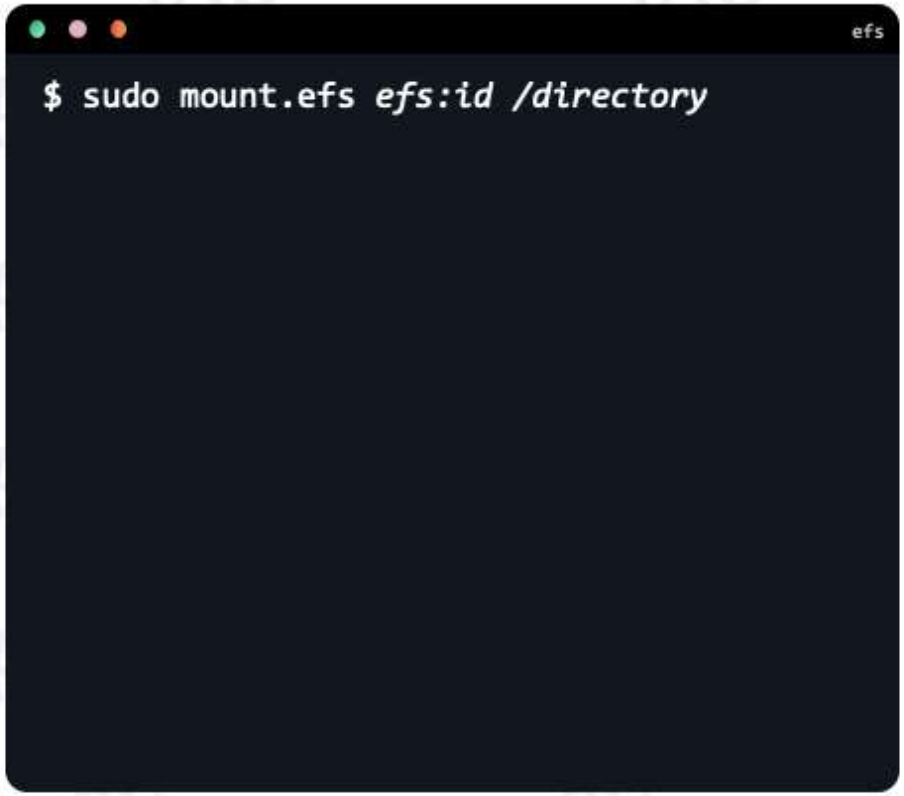
Alternatively, if you know the specific access patterns for your workloads (including throughput, latency, and storage needs), then you can choose different performance and throughput modes.

- The *Max I/O performance mode* can scale to higher levels of aggregate throughput and operations per second. However, these file systems have higher latencies for file system operations.
- With *Provisioned Throughput mode*, you specify a level of throughput that the file system can drive independent of the file system's size or burst credit balance.

- The *Bursting Throughput mode* provides throughput that scales with the amount of storage in your file system and supports bursting to higher levels for up to 12 hours per day.

Installing amazon-efs-utils





The efs:id is the ID of the efs filesystem and can be found in the AWS console

/directory – select which directory you would like to mount the filesystem at

Summary

01 Filesystem storage services provided by Amazon

02 EFS supports the Network File System v4 protocol NFSv4. So, applications that use the protocol can seamlessly work with EFS

03 EFS does not support Windows-based EC2 instances; Only Linux based

04 EFS filesystems can be mounted onto multiple EC2 instances at the same time

Summary

05 EFS filesystem is made available inside a VPC via mount targets. Mount targets get IP addresses from the subnets they are deployed in

06 EFS offers two storage classes: Standard storage classes and One Zone storage classes

07 EFS has two different modes: General purpose performance mode and Elastic throughput mode

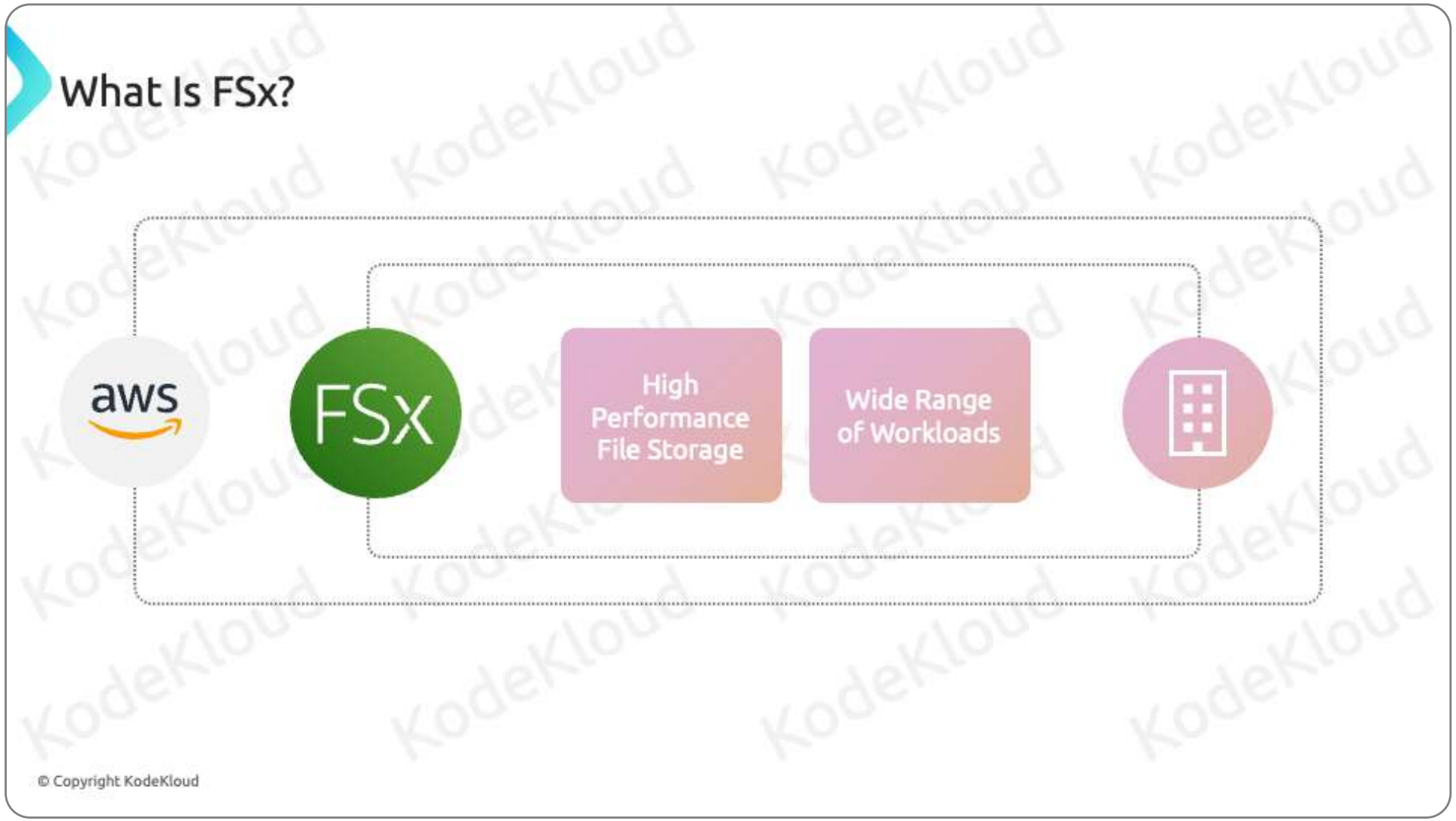
04 EFS can be mounted but cannot be booted, can't install an operating system

Service Section

The Power of File System Storage

FSx

What Is FSx?

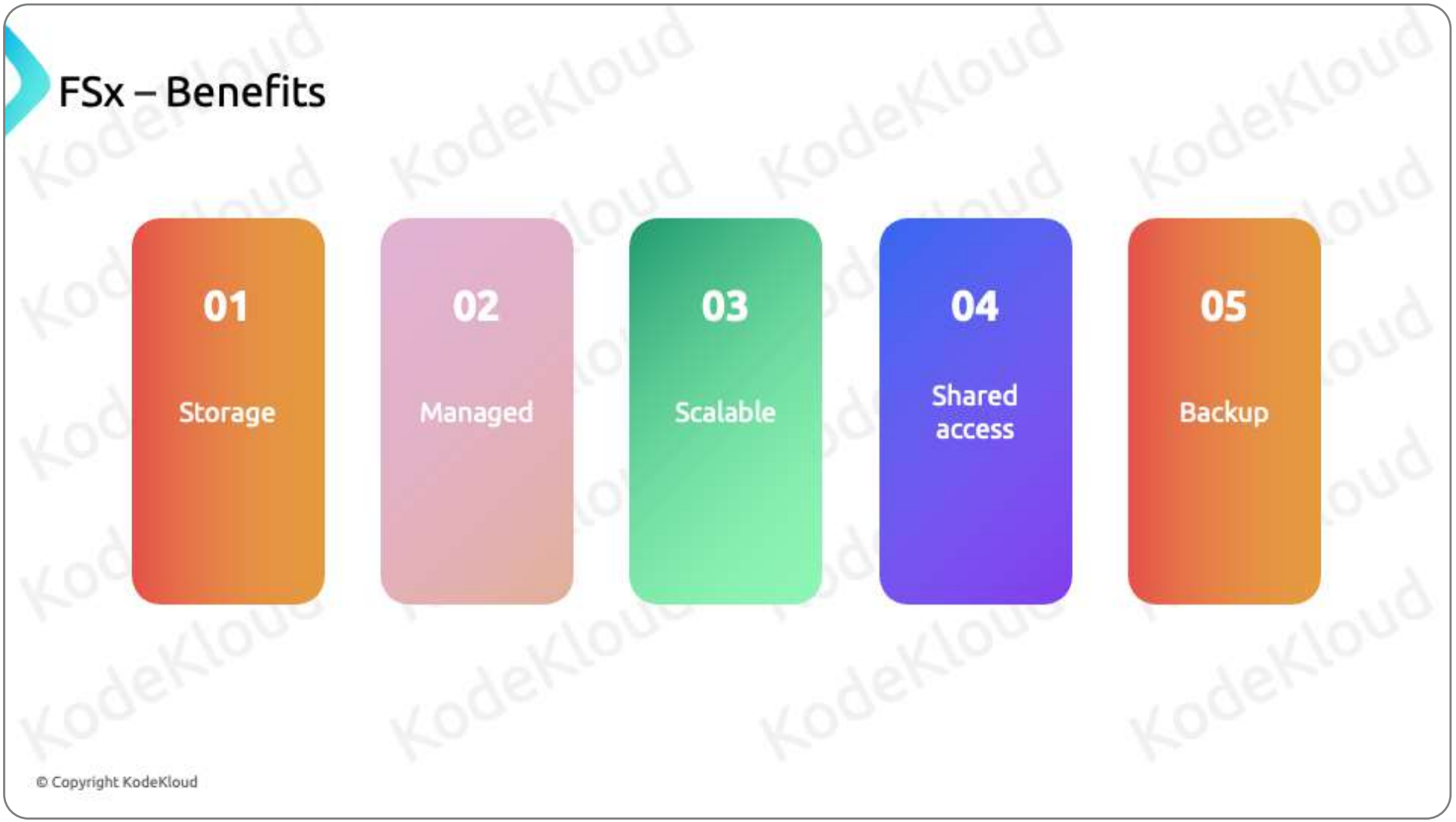


Amazon FSx is a fully managed service that provides high performance file storage for a wide range of workloads designed to make it easier for businesses and orginaztion to setup and mange file storage in the cloud

So instead of you having to deal with the headaches of provisioning and managing file servers, AWS will handle the work for ou

So you no longer need to worry about:

- Provisioning file servers and storag volumes
- Replicating data
- Patching file server
- Addressing hardware issues
- Performing manual backups



FSx provides the following benefits

1.Storage: FSx gives you a place to store your files, just like how you use your computer's hard drive to save documents, photos, and videos.
2.Managed: AWS takes care of all the technical stuff for you. You don't need to worry about setting up servers or dealing with

complex storage systems.

3. **Scalable**: You can easily make your storage bigger or smaller depending on your needs. It's like adding more shelves to your filing cabinet when you have more files.
4. **Shared Access**: FSx allows multiple people or computers to access the same files at the same time, making it great for collaborative work.
5. **Backup**: It also offers automatic backup options, so you don't lose your files in case something goes wrong.

Overall, Amazon FSx simplifies the process of setting up and managing file storage in the AWS cloud, whether you're running Windows-based workloads or need high-performance file storage for compute-intensive tasks.

FSx comes in 4 different version

1. Amazon FSx for Windows File Server - This type of FSx file system is designed to be fully compatible with Windows file servers. It is based on the Windows Server operating system and provides features that are familiar to Windows users and administrators.

- It supports the Server Message Block (SMB) protocol, which is commonly used in Windows environments for file sharing and access.
- You can easily integrate it with Microsoft Active Directory for user authentication and access control

- It supports data deduplication, which can help reduce storage costs by eliminating duplicate data.
- You can set quotas to limit the amount of storage space individual users or groups can use.

2. Amazon FSx for Lustre: This type of FSx file system is optimized for high-performance, parallel file processing, and is commonly used in scenarios like scientific computing, machine learning, and data analytics

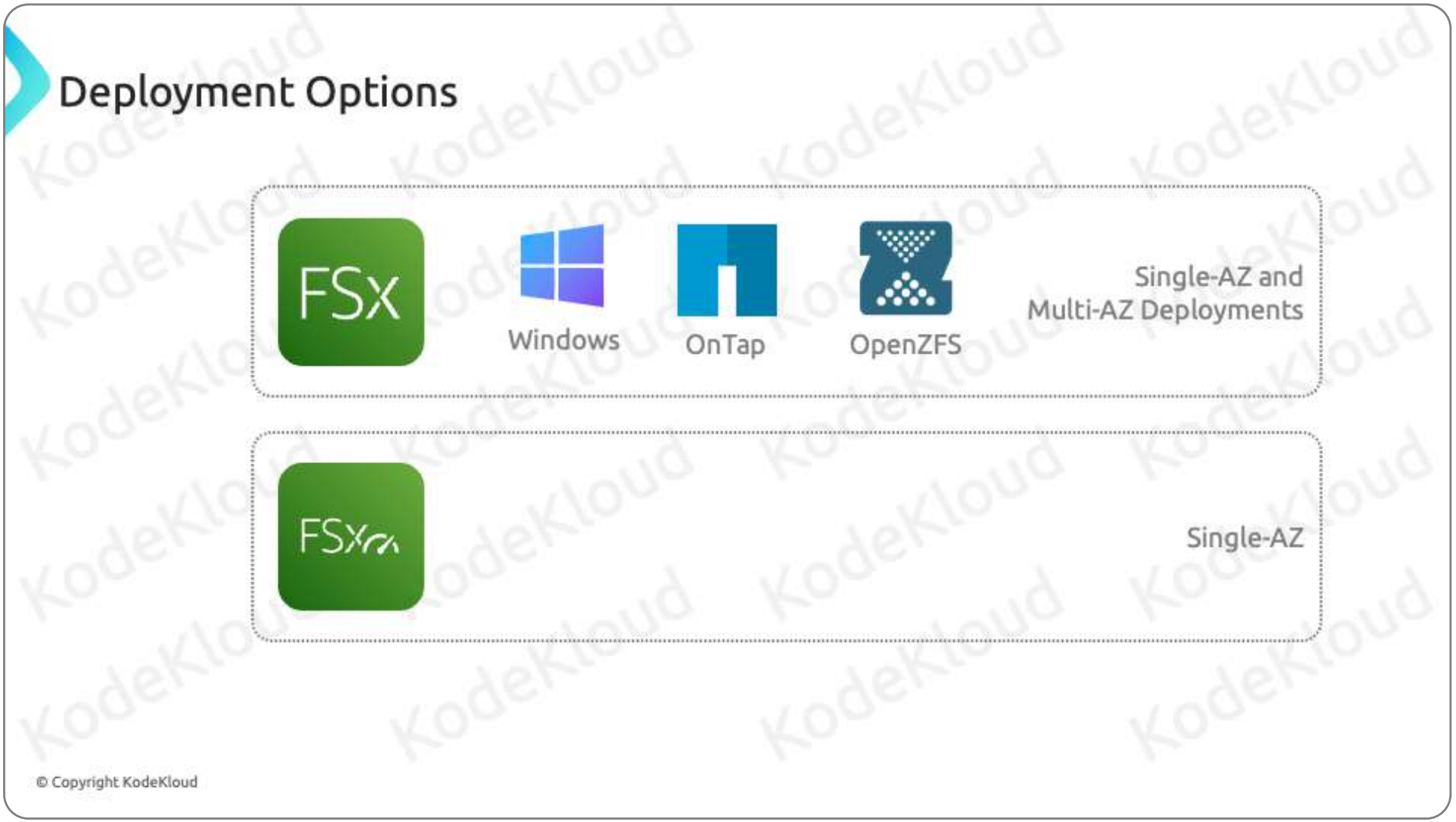
- It provides low-latency, high-throughput access to data, making it suitable for workloads that require intensive data processing.
- It is built on the Lustre file system, a popular choice for high-performance computing and data-intensive applications.
- Amazon FSx for Lustre integrates seamlessly with other AWS services like Amazon S3, AWS DataSync, and AWS Batch for

data processing and storage workflows.
- You can easily scale the file system's capacity and throughput based on your workload requirements.

3. Amazon FSx for NetAPP ONTAP: filesystem built ontop of NetApps ONTAP file system
- Offers high performance storage that's accessible from linux, windows, and macOS via NFS, SMB, and iSCSI protocols
- Can scale your file system up or down in response to workload demands
- Feature rich – can perform snapshots, clones, replications and much more

4. Amazon FSx for OpenZFS - fully managed file storage service that makes it easy to move data to AWS from on-premises ZFS or other Linux-based file servers

- Built ontop of the opensource OpenZFS file system
- Support for access from Linux, Windows, and MacOS via industry standard NFS Protocol
- Utilizes powerOpenZFS capabilities including data compression, snapshots, and data cloning
- Built in data protection and security features



FSx for windows, ONTAP, and OpenZFS support both single-AZ and multi-Az deployments
Lustre only supports Single-AZ deployments

FSx Comparison

	FSx for NetApp OnTap	FSx for Windows	FSx for Lustre	FSx for OpenZFS
Client compatibility	Windows, Linux, macOS	Windows, Linux, macOS	Linux	Windows, Linux, macOS
Protocol support	SMB, NFS, iSCSI	SMB	Custom protocol	NFS
latency	<1ms	<1ms	<1ms	<0.5s
Max throughput	4-6 GB/s	12-20 GB/s	1000 GB/s	10-21 GB/s
Max file system size	Virtually unlimited	64 TiB	Multiple PBs	512 TiB

Summary

01 Amazon FSx is a fully managed service that provides high-performance file storage for a wide range of workloads

02 Amazon FSx comes in 3 flavors – FSx for Windows, FSx for Lustre, and FSx for ONTAP

03 Amazon FSx for Windows provides a fully managed Windows file server

04 Amazon FSx for Windows is based on SMB protocol and integrates with Windows Active Directory for authentication

05 Amazon FSx for Lustre – Optimized for high-performance and parallel file processing; great for scientific computing and machine learning

Summary

06 Amazon FSx for Lustre is based on the Lustre file system

07 Amazon FSx for ONTAP – Built on top of NetApps ONTAP file system

08 Amazon FSx for ONTAP – Utilizes NFS, SMB, and iSCSI protocol

09 Amazon FSx for OpenZFS – Built on top of the open-source OpenZFS file system

10 Amazon FSx for OpenZFS – Accessible from Linux, Windows, and macOS compute instances by utilizing NFS protocol

Service Section

The Power of Object Storage

S3 Overview

What Is AWS S3?

Simple Storage Service (S3)

Scalability

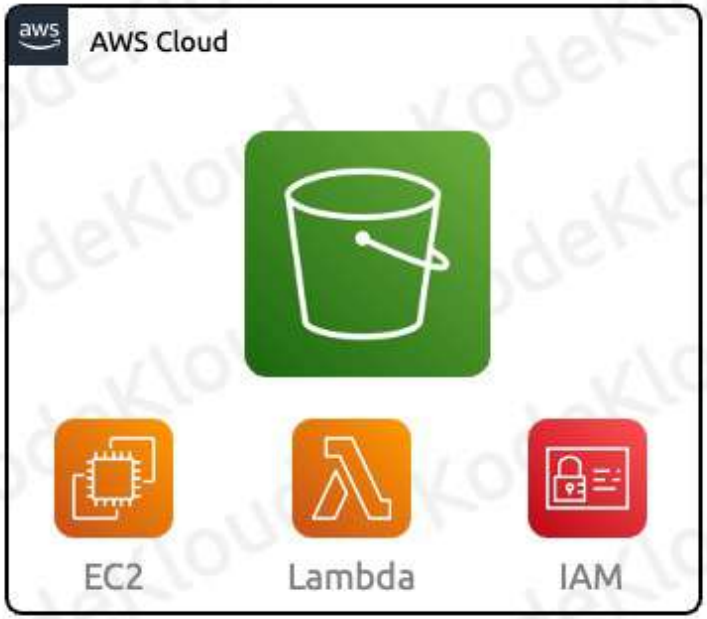
Data Availability

Security

Performance

Simple Storage Service (S3)

Simple Storage Service (S3)



Simple Storage Service (S3)

AWS Console

AWS CLI

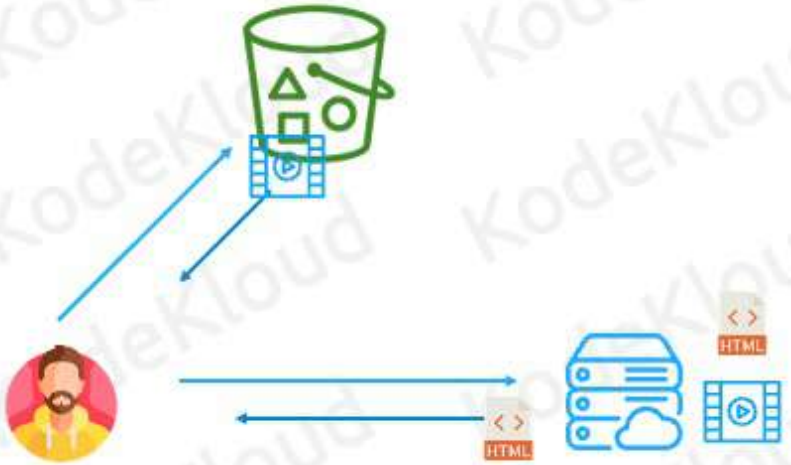
SDK

Rest API

Simple Storage Service (S3)

S3 – Use Cases

S3 – Use Cases



Bucket

Bucket

Bucket

App File

App 2

Media File

Objects

Objects are files that are uploaded to S3

An object has:

Key – The file name

Value – File data

VersionID/Metadata/Other information

S3 File Structure

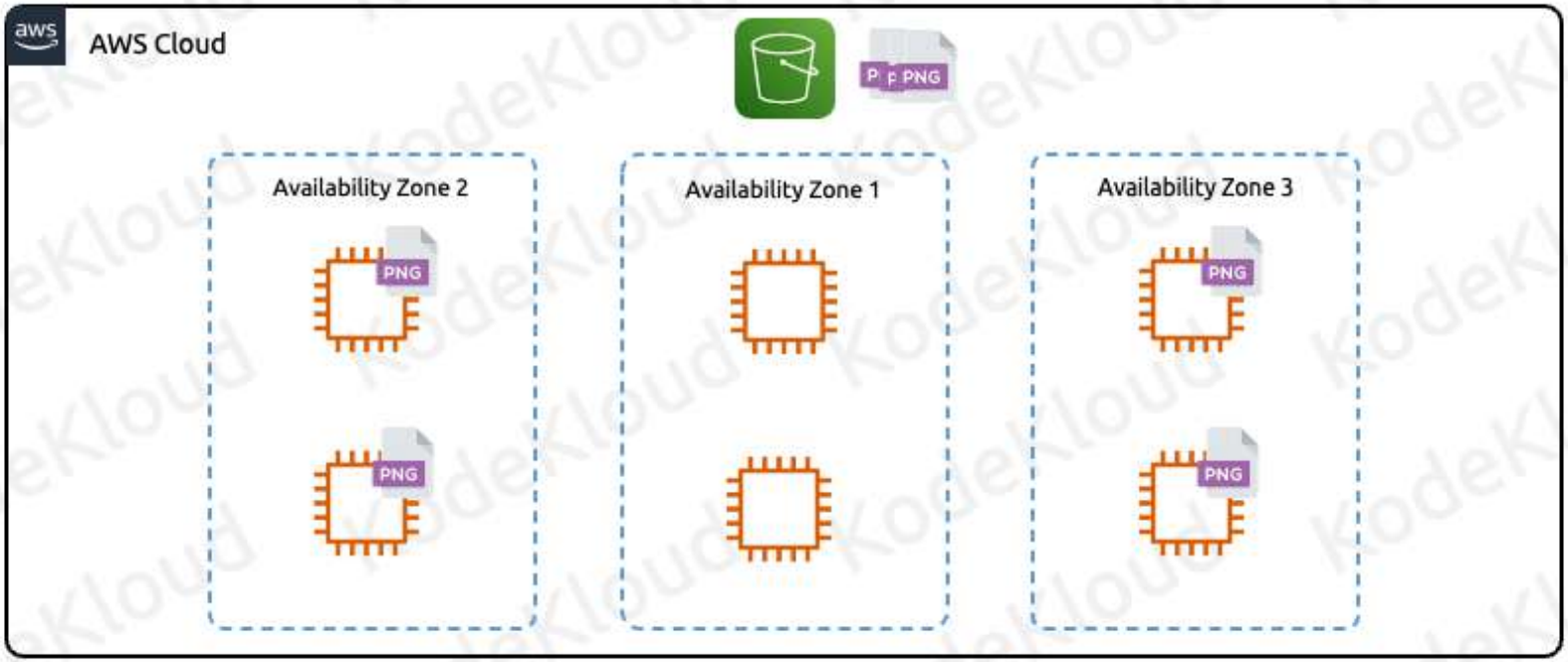
S3 Buckets have a flat file structure

- File1.txt
- File2.txt
- File3.txt
- File4.txt

music/

- /music/song1
- /music/song2
- /music/song3

Availability



S3 Bucket Names

Important

S3 bucket names must be unique globally across all AWS accounts

https://kodekloud.s3.amazonaws.com

Bucket Name

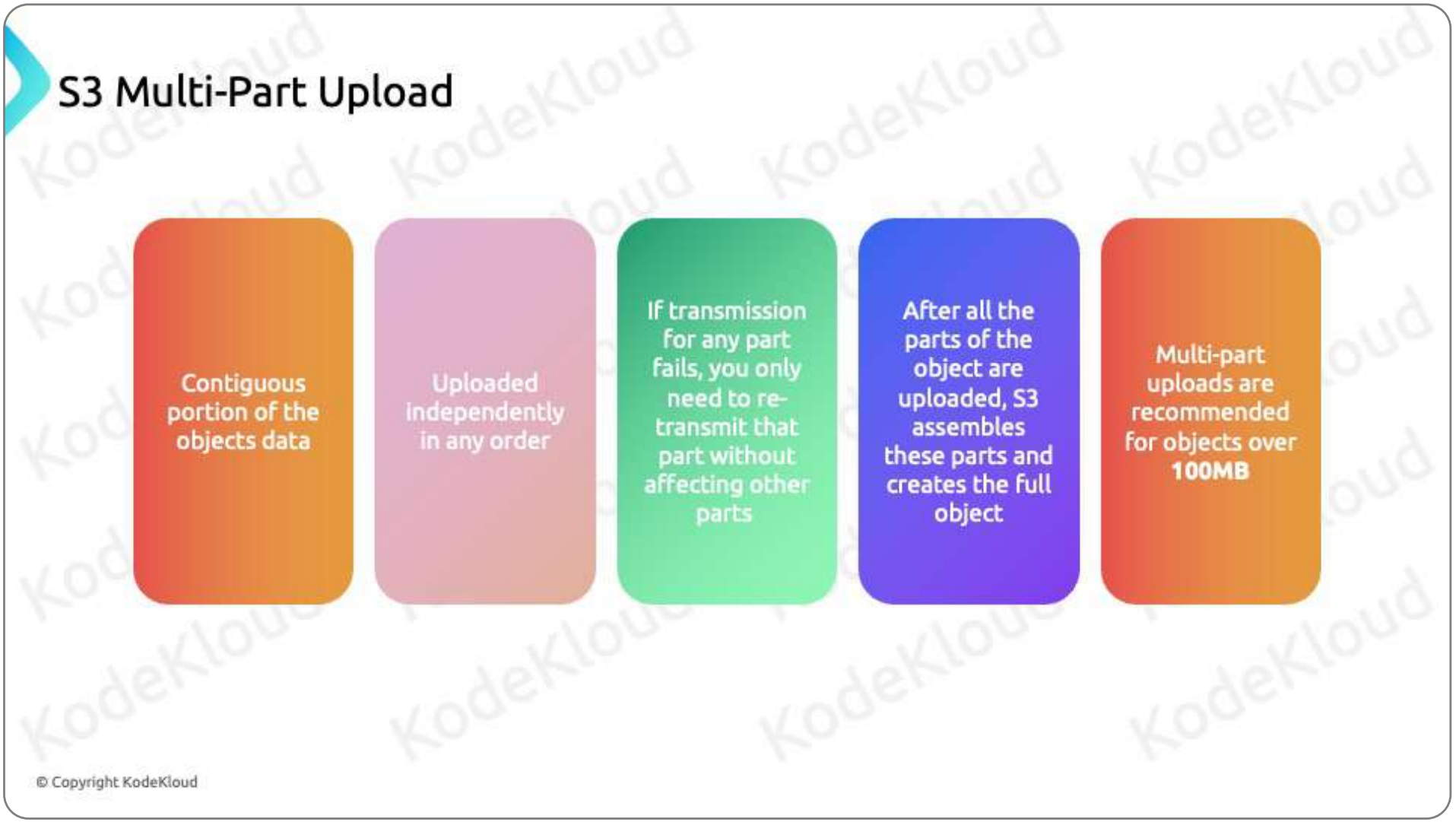
https://kodekloud.s3.amazonaws.com

S3 Restrictions

S3 can handle unlimited number of objects

Maximum size of a single file is 5 TB

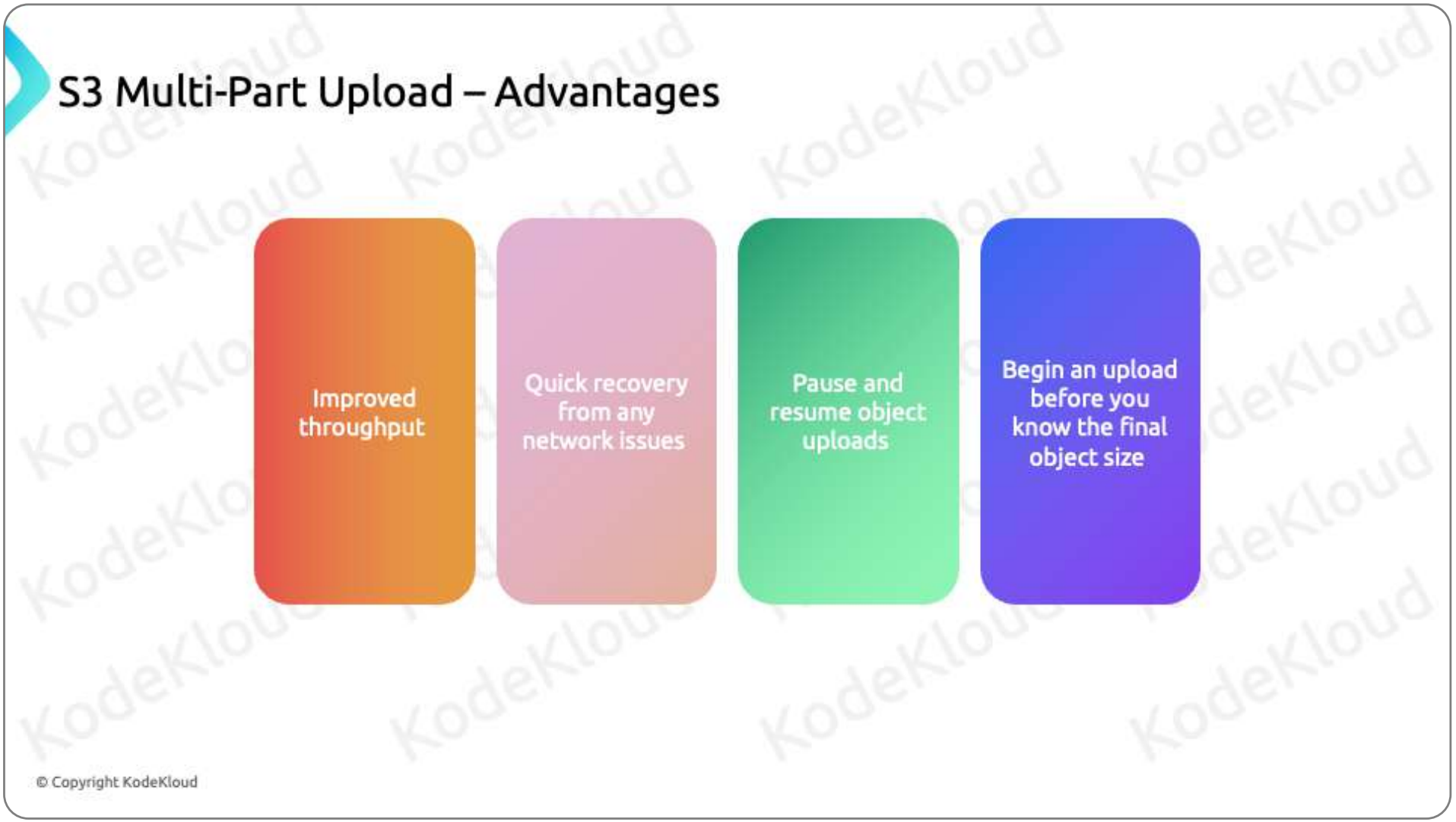
An AWS account supports 100 buckets by default, but this number can be increased to 1,000 by requesting a service limit increase



Instead of uploading a single object at once, you can take advantage of a feature called multipart upload which allowd you to upload object as a set of parts.

- Each part is a contiguous portion of the objects data
- Each part can be uploaded independently and in any order
- If transmission for any part fails, you only need to re-transmit that part without affecting other parts.
- After all the parts of the object are uploaded, S3 assembles these parts and creates the full object

- Multipart uploads are recommended for objects over 100MB



Using multipart upload provides the following advantages:

- **Improved throughput** – You can upload parts in parallel to improve throughput.
- **Quick recovery from any network issues** – Smaller part size minimizes the impact of restarting a failed upload due to a network error.
- **Pause and resume object uploads** – You can upload object parts over time. After you initiate a multipart upload, there is no expiry; you must explicitly complete or stop the multipart upload.

•**Begin an upload before you know the final object size** – You can upload an object as you are creating it.

I mentioned in the previous slide that the maximum object size s3 supports is 5TB. With S3 if you upload a single object at once the maximum file size is 5GB. To achieve the 5TB object size, you must use a multipart uplaod

Summary

01 Object storage service from Amazon that provides industry-leading scalability, data availability, security, and performance

02 Use cases include storing static websites, media files, logs, and traces

03 S3 and object storage have a flat file structure (single folder), so you cannot boot or mount from it

04 Objects are nothing more than files and they have a key, which is the name of the object, and the value which is the content of the file plus some other metadata

Summary

05 Buckets are a container (folder) for objects

06 Bucket names must be unique globally across all AWS accounts

07 S3 can handle an unlimited number of objects

08 Maximum size of an individual object is 5 TB

Summary

09 Multi-part upload allows you to break up an object into parts before uploading

Service Section

The Power of Object Storage

S3 Storage Classes

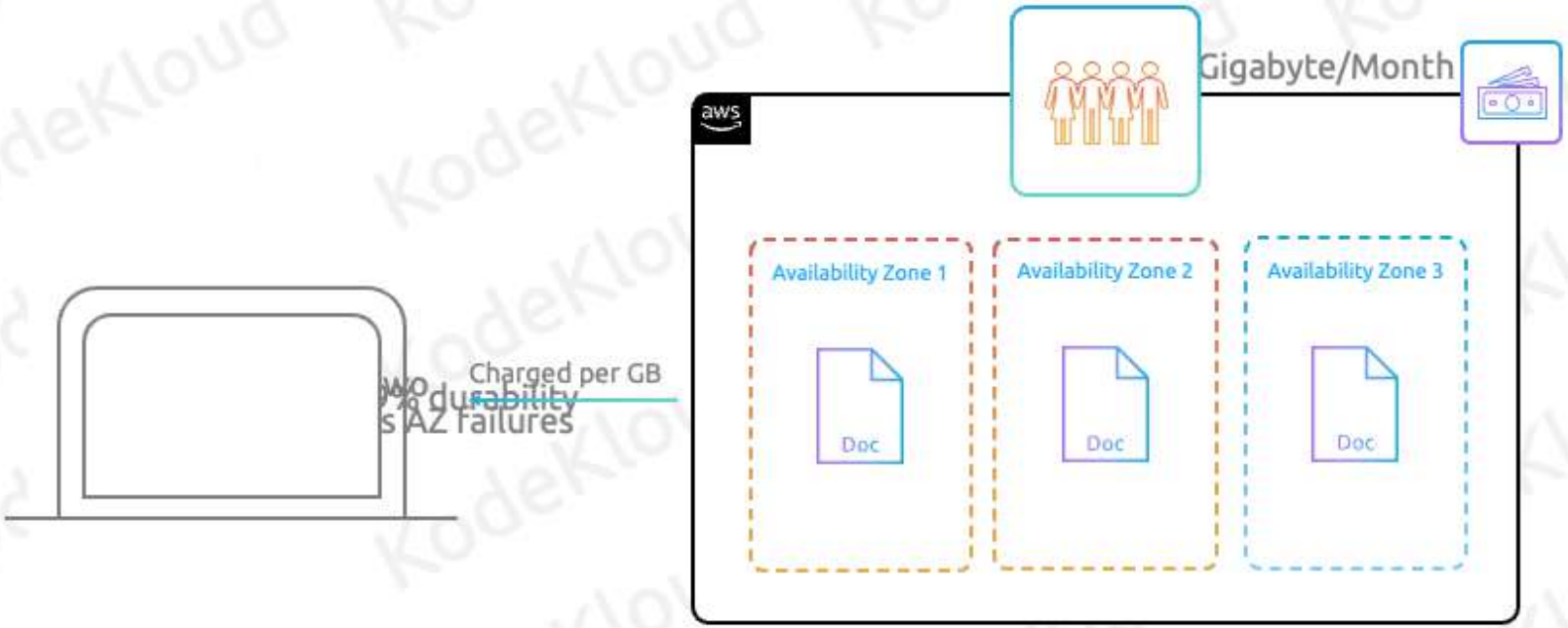
Storage Classes

Data Access

Resiliency

Cost

S3 Standard (default)



S3 Standard-IA

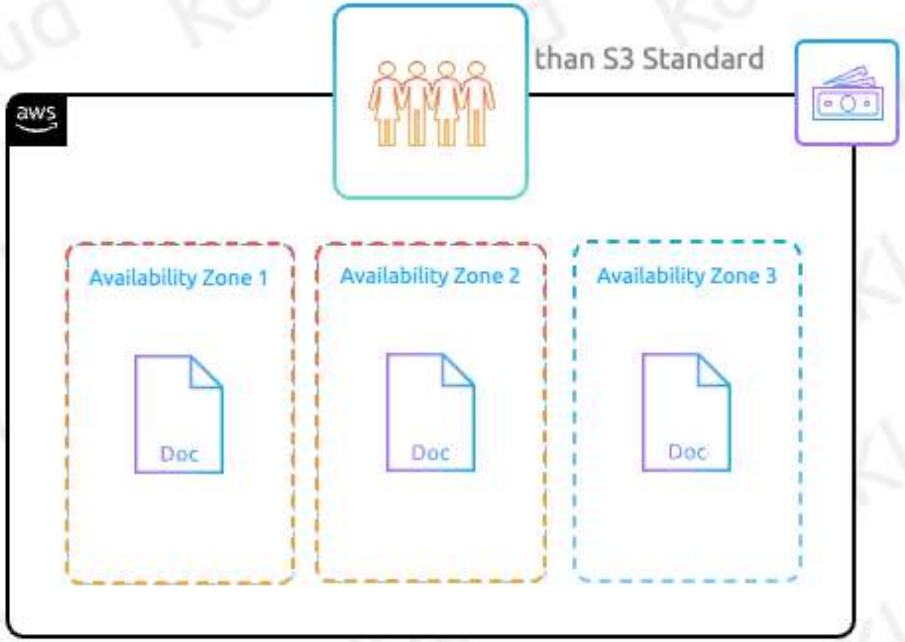
Has a retrieval fee

Minimum duration charge of 30 days

Charged per GB

two
9% durability
us AZ failures

Minimum size charge of 128 KB per object



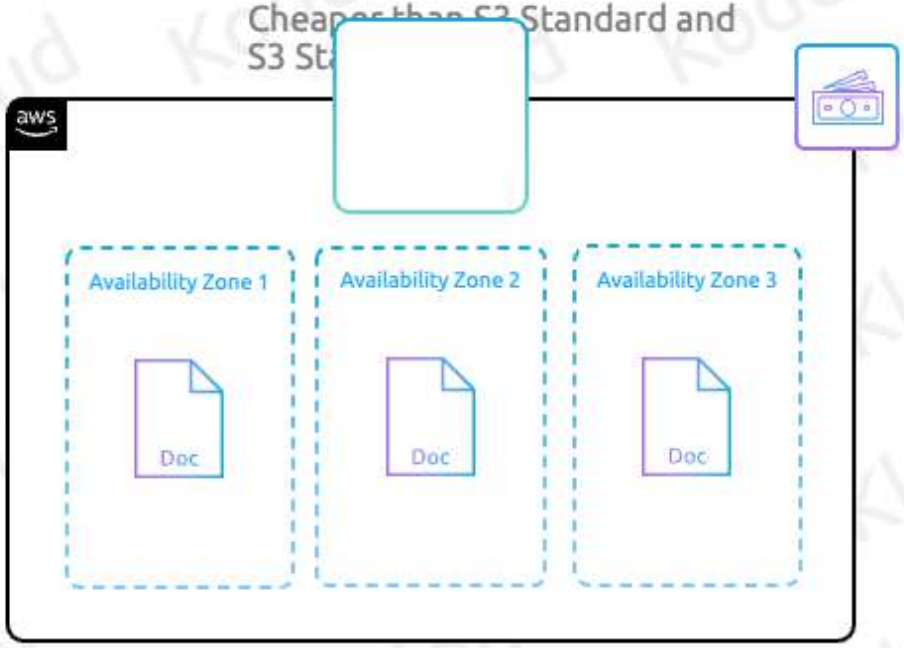
S3 One Zone-IA

Has a retrieval fee

Minimum duration charge of 30 days

Minimum size charge of 128 KB per object

Charged per GB



Note

Designed for IA data

Not required to handle AZ failure

Replication still occurs within AZ

S3 Glacier-Instant

Has a retrieval fee

Minimum duration charge of 90 days

Minimum size charge of 128 KB per object

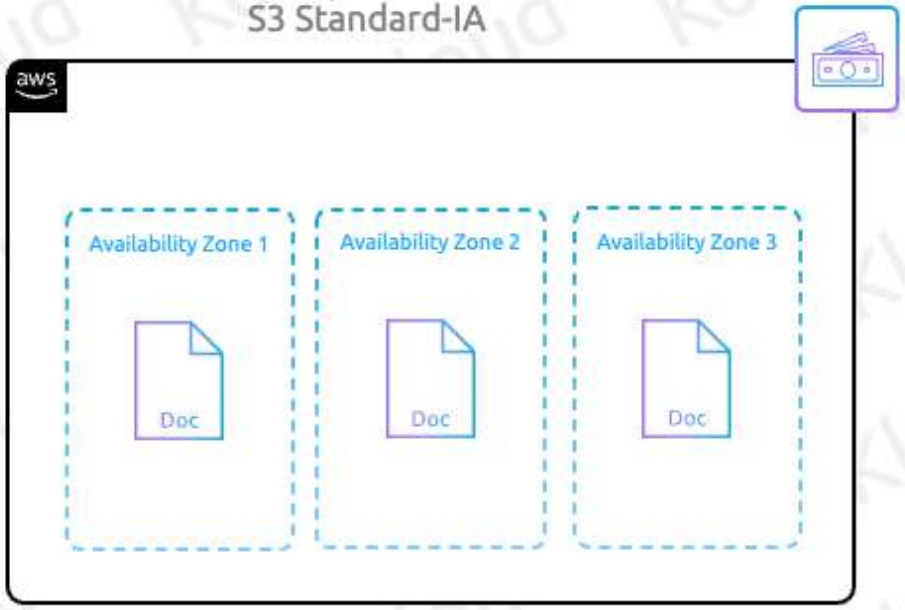
Charged per GB

Note

Low-cost option for rarely accessed data

Performance same as that of S3 standard

Cheaper than S3 Standard and S3 Standard-IA



Summary

Very cheap storage

Higher retrieval cost

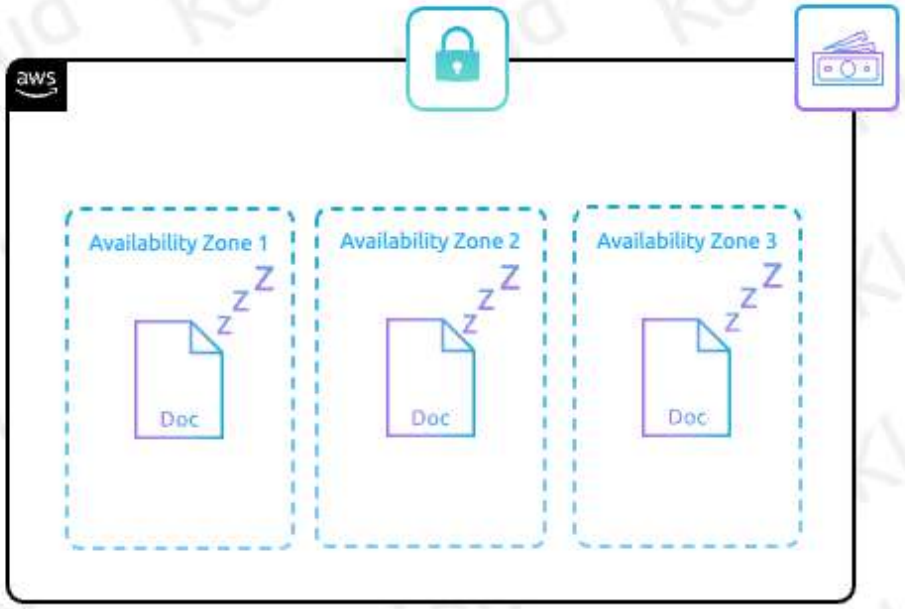
Longer minimum duration

S3 Glacier-Flexible

Cheaper than S3 Standard and so on...

Has a retrieval fee
Minimum duration charge of 90 days
Minimum size charge of 40 KB per object

Charged per GB



Options

Bulk	:	5–12 Hours
Expedited	:	1–5 Minutes
Standard	:	3–5 Hours

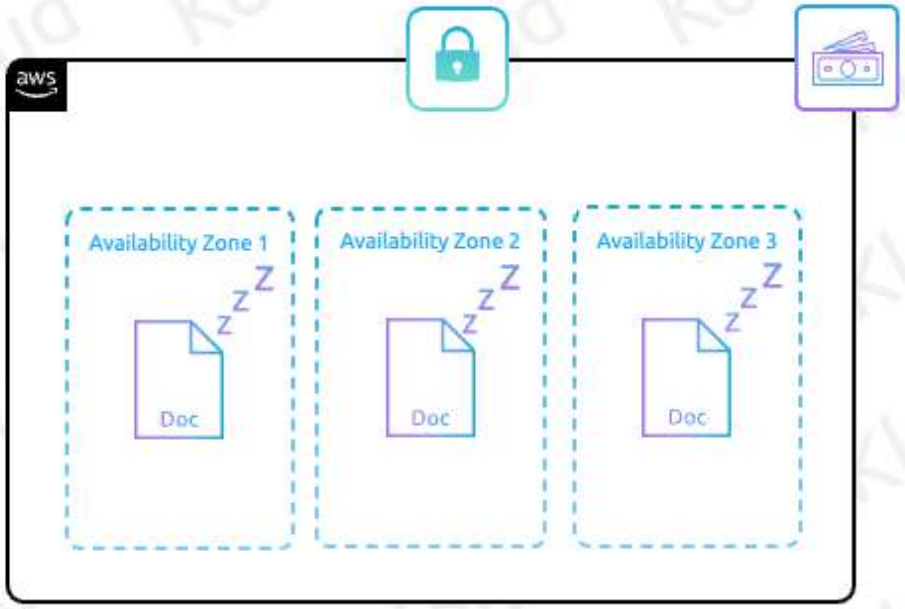
During retrieval, objects are stored in S3 Standard-IA class temporarily

S3 Glacier Deep Archive

The cheapest storage class in S3

Has a retrieval fee
Minimum duration charge of 180 days
Minimum size charge of 40 KB per object

Charged per GB



Options

Standard : 12 Hours
Bulk : 48 Hours

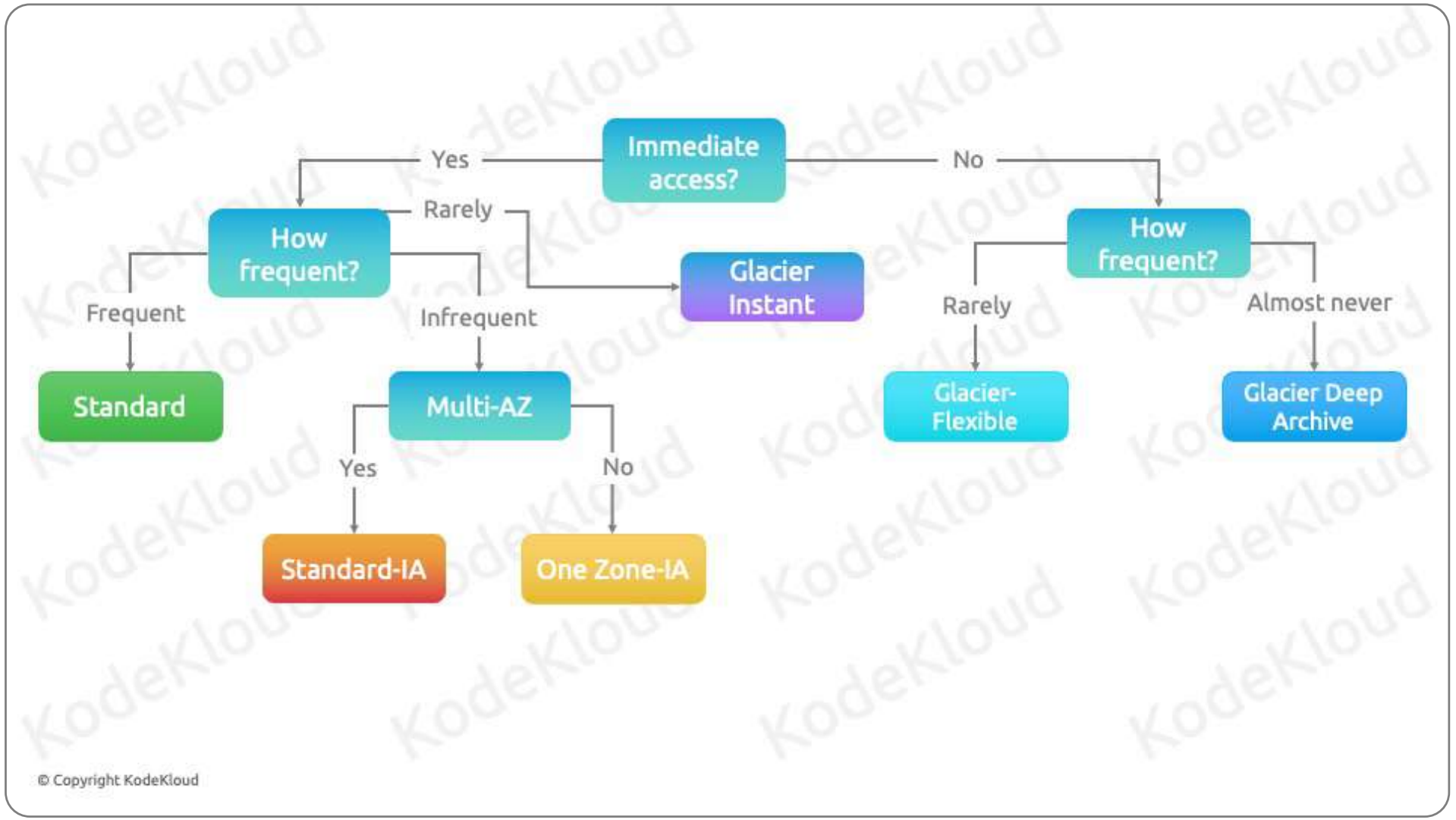
S3 Intelligent-Tiering

01

Automatically reduces storage costs by intelligently moving data to the most cost-effective access tier

02

Apart from the cost of a storage class an object gets assigned to, all objects will also incur a monitoring/automation cost per **1,000 objects**


Immediate access?
Yes
No
How frequent?
Rarely
Glacier Instant
Frequent
Infrequent
Standard
Multi-AZ
Yes
No
Standard-IA
One Zone-IA
How frequent?
Rarely
Almost never
Glacier-Flexible
Glacier Deep Archive
© Copyright KodeKloud

S3 Storage Class Comparison

Storage class	Designed for	Durability (designed for)	Availability (designed for)	Availability Zones	Min storage duration	Min billable object size	Other considerations
S3 Standard	Frequently accessed data (more than once a month) with millisecond access	99.999999999%	99.99%	>= 3	None	None	None
S3 Standard-IA	Long-lived, infrequently accessed data (once a month) with millisecond access	99.999999999%	99.9%	>= 3	30 days	128 KB	Per-GB retrieval fees apply.
S3 Intelligent-Tiering	Data with unknown, changing, or unpredictable access patterns	99.999999999%	99.9%	>= 3	None	None	Monitoring and automation fees per object apply. No retrieval fees.
S3 One Zone-IA	Recreatable, infrequently accessed data (once a month) with millisecond access	99.999999999%	99.5%	1	30 days	128 KB	Per-GB retrieval fees apply. Not resilient to the loss of the Availability Zone.
S3 Glacier Instant Retrieval	Long-lived, archive data accessed once a quarter with millisecond access	99.999999999%	99.9%	>= 3	90 days	128 KB	Per-GB retrieval fees apply.
S3 Glacier Flexible Retrieval	Long-lived archive data accessed once a year with retrieval times of minutes to hours	99.999999999%	99.99% (after you restore objects)	>= 3	90 days	NA*	Per-GB retrieval fees apply. You must first restore archived objects before you can access them. For information, see Restoring an archived object.
S3 Glacier Deep Archive	Long-lived archive data accessed less than once a year with retrieval times of hours	99.999999999%	99.99% (after you restore objects)	>= 3	180 days	NA**	Per-GB retrieval fees apply. You must first restore archived objects before you can access them. For information, see Restoring an archived object.
RRS (not recommended)	Noncritical, frequently accessed data with millisecond access	99.99%	99.99%	>= 3	None	None	None

When creating a new object, you can specify its storage class, and the way you do this is by adding the `x-amz-storage-class` request header to specify the desired storage class. If you don't add the header, then s3 will use the default storage class

Summary

01 Storage classes provide varying levels of data access, resiliency, and cost

02 Storage classes are set up on upload by setting the x-amz-storage-class request header but can be changed after upload as well

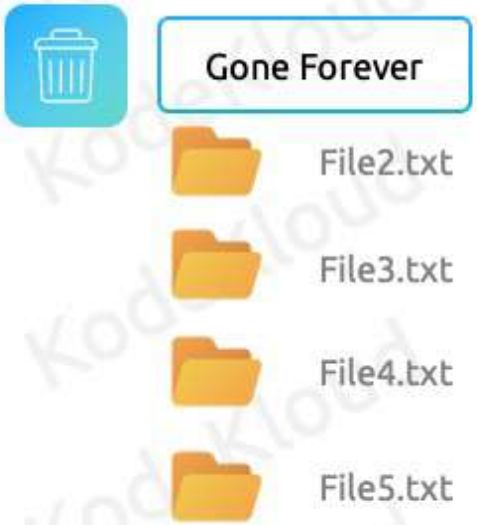
Service Section

The Power of Object Storage

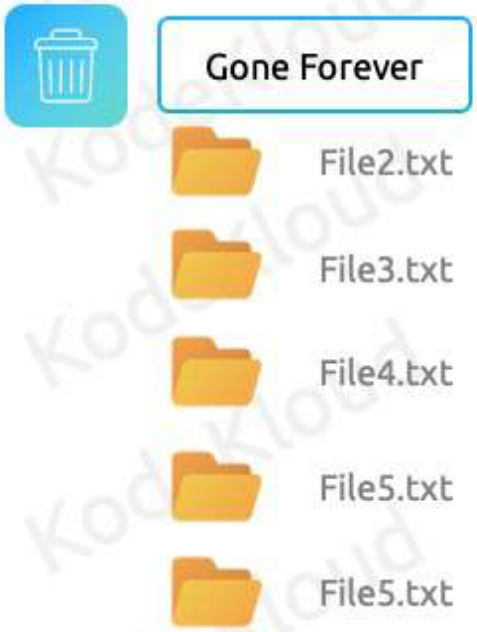
S3 Versioning

Versioning

Versioning



Versioning



Versioning

Three States

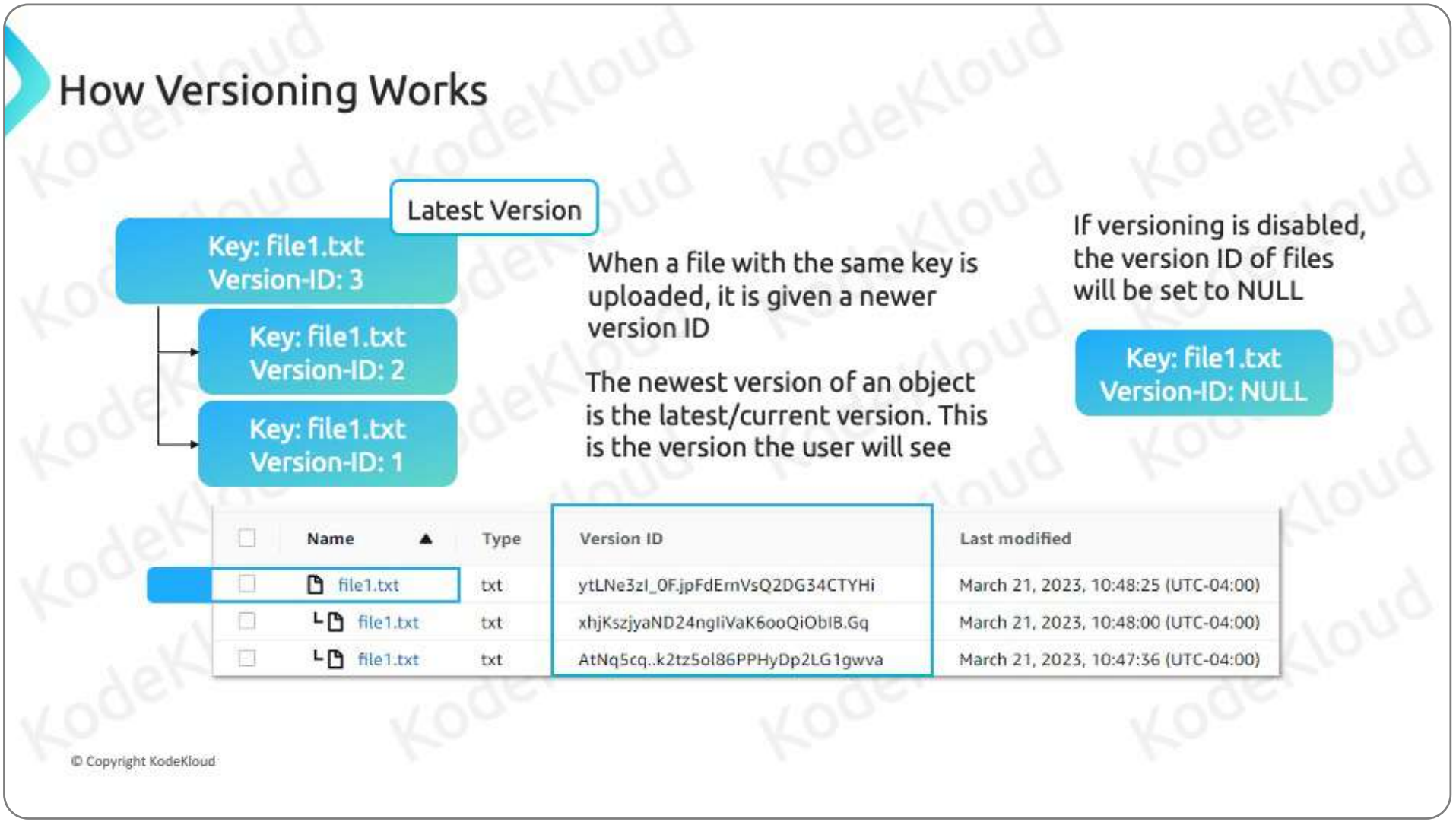
Unversioned

Versioning Enabled

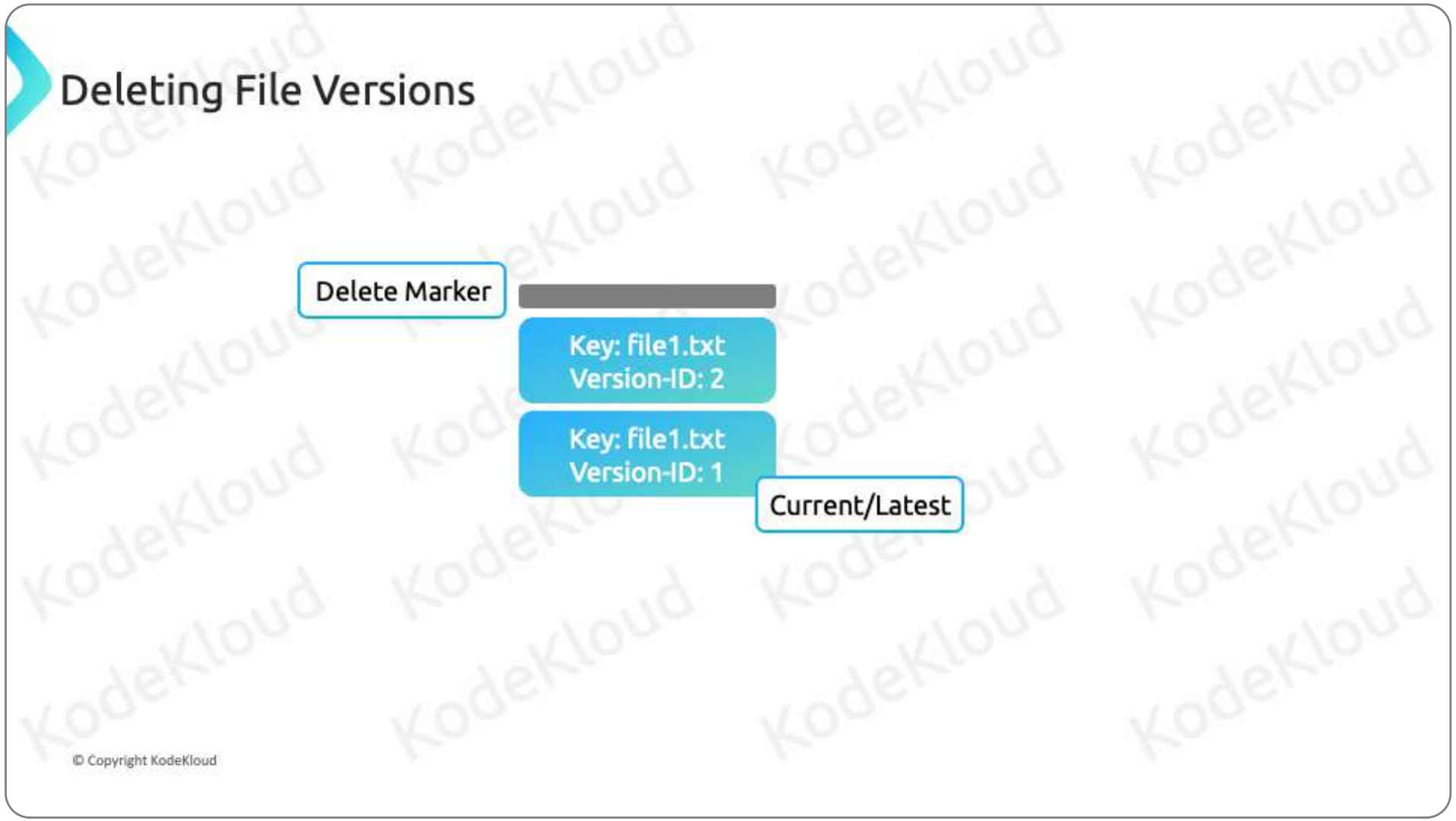
Versioning Suspended

How Versioning Works

When a file with the same key is uploaded, it is given a newer version ID



If you request a key without a specific version you will always get the latest version. But you can always request a specific version by providing the version id



When you delete an object without specifying a version id, a special version of that object called a delete marker will be added. The delete marker is technically a new version of that object, but it doesn't actually delete anything; it just makes it look like its deleted. The marker will hide all previous versions of the object.

If you want to undelete an object just delete the delete marker.

With s3 you can specify a specific version of an object you want to delete. When you specify a specific version, that will permanently delete that object.

If you are deleting the most recent version, then the next most recent version becomes the current version

Versioning Pricing

Version Suspending

Multi-Factor Authentication (MFA) Delete

Multi-Factor Authentication (MFA) Delete

Note

When this feature is enabled, MFA is required to change the versioning state of the bucket

MFA is required to delete versions and can only be enabled using CLI

Summary

01 Versioning is a feature that allows you to preserve, retrieve, and restore every version of an object stored in your bucket

02 Versioning is disabled on buckets by default and must be explicitly enabled

03 Versioning has to be enabled at the bucket level; you cannot enable versioning for individual objects

04 Buckets have 3 versioning states – unversioned, versioning enabled, and versioning suspended

Summary

05 Once versioning is enabled on a bucket, it cannot move back to an unversioned state and can only be in a suspended state

06 In a suspended state, previous versions remain, but no new versions will be created

07 Users are charged for each version of an object

08 MFA can be configured to secure the versioning state of a bucket

Service Section

The Power of Object Storage

S3 Bucket Policies

ACLs and Resource Policies

S3 Access

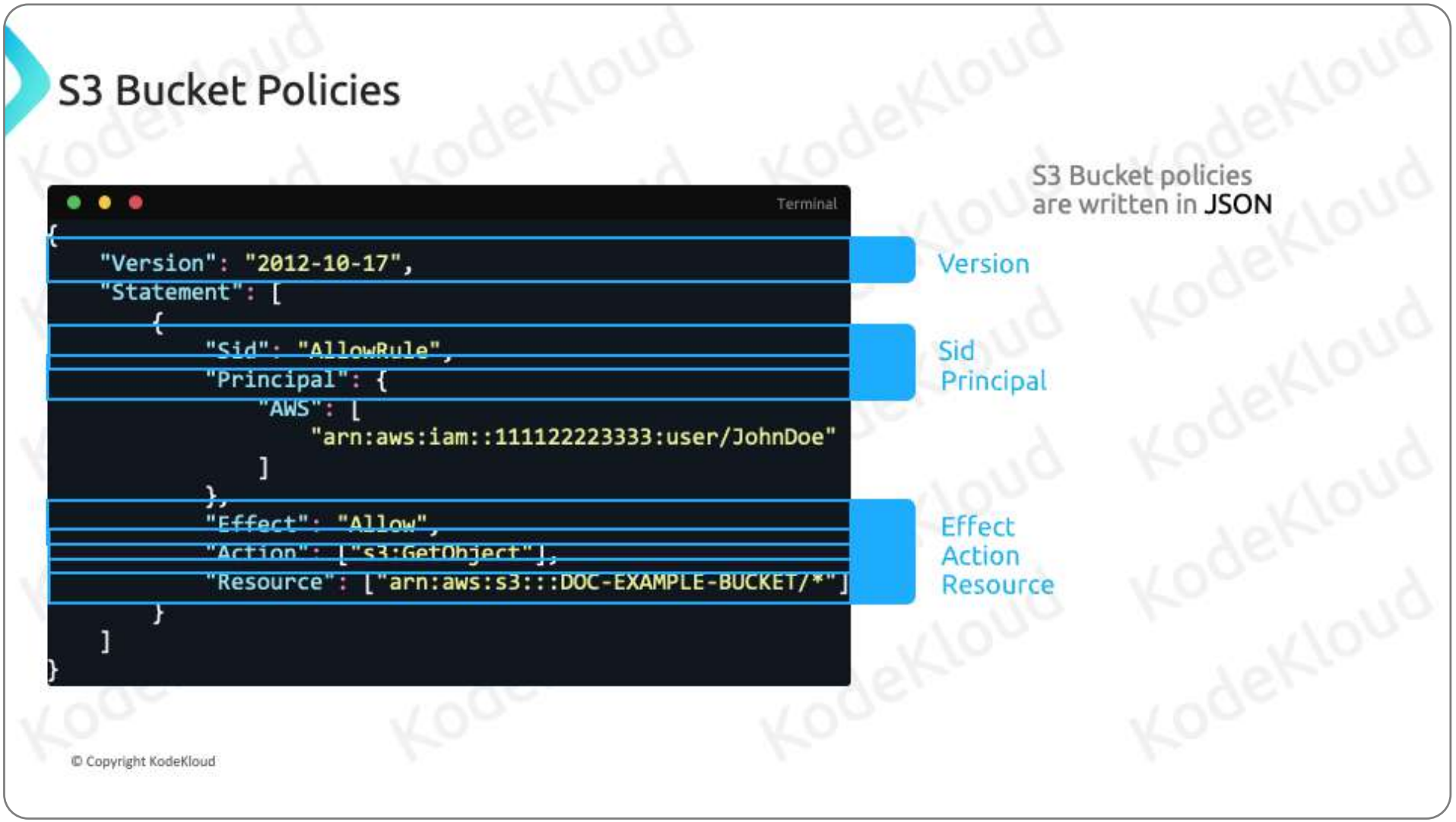
S3 Bucket Policies

Resource Policy

Determines who has access to an S3 resource

S3 Bucket Policy

Determines who can have access to the bucket and what operations they can perform



Version policy element specifies the language syntax rules that are to be used to process a policy. 2012-10-17 is the current version of the policy

This was an extra security feature that was added due to the fact that many aws customers were accidentally exposing their s3 buckets due to misconfigured bucket policies.

Terminal

```
{
    "Id": "PolicyId2",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAll",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET1/*"
            ],
        }
    ]
}
```

Block *all* public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  S3 will ignore all ACLs that grant public access to buckets and objects.
- **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

IAM Policies vs Resource Policies

IAM Policies vs Resource Policies

IAM Policies vs Resource Policies

IAM Policy

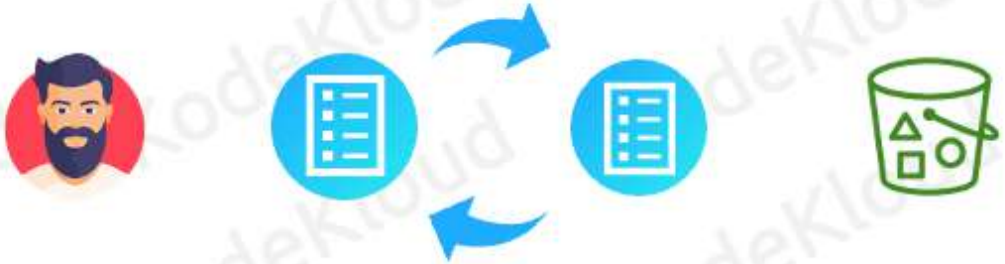
Can only be applied to authenticated AWS users

Cannot be applied to anonymous users

Resource Policy

Since the policy is applied to the resource, rules can be added for anonymous/public users

IAM Policies vs Resource Policies



IAM Policies vs Resource Policies

S3 ACLs

ACLs

Have a legacy access control mechanism that predates IAM

Note

Are inflexible and provide only a limited set of rules

Cannot be applied to a group of objects

Can be used but it is not recommended

ACL permissions

Permission	When granted on a bucket	When granted on an object
READ	Allows grantee to list the objects in the bucket.	Allows grantee to read the object data and its metadata
WRITE	Allows grantee to create new objects in the bucket. For the bucket and object owners of existing objects, also allows deletions and overwrites of those objects.	Not applicable
READ_ACP	Allows grantee to read the bucket ACL	Allows grantee to read the object ACL
WRITE_ACP	Allows grantee to write the ACL for the applicable bucket	Allows grantee to write the ACL for the applicable object
FULL_CONTROL	Allows grantee the READ, WRITE, READ_ACP, and WRITE_ACP permissions on the bucket	Allows grantee the READ, READ_ACP, and WRITE_ACP permissions on the object

Summary

01 Determines who can have access to the bucket and what operations they can perform

02 Within a Policy you have Principal, Resources, Effect, and Action

03 Principal determines who the policy should apply to

04 Resources determines what AWS resources the policy should apply to

Summary

05 Action determines what the principal is allowed to perform on the resources

06 Effect either allows or denies an action

07 Bucket policies work alongside IAM policies

08 To allow public users access to a bucket, bucket policies need to be used, as IAM policies only apply to AWS users

Summary

09 ACLs are a legacy access control method that predate IAM

Service Section

The Power of Object Storage

S3 Static Web Hosting

Static Hosting

Note

A website is just an HTML file

Static Hosting

Structure/ Content of a website

Adds colors and provides visual elements

Adds dynamic functionality

Images/Videos/Audio

Static Hosting

Static Hosting

Allows access to website files through HTTP

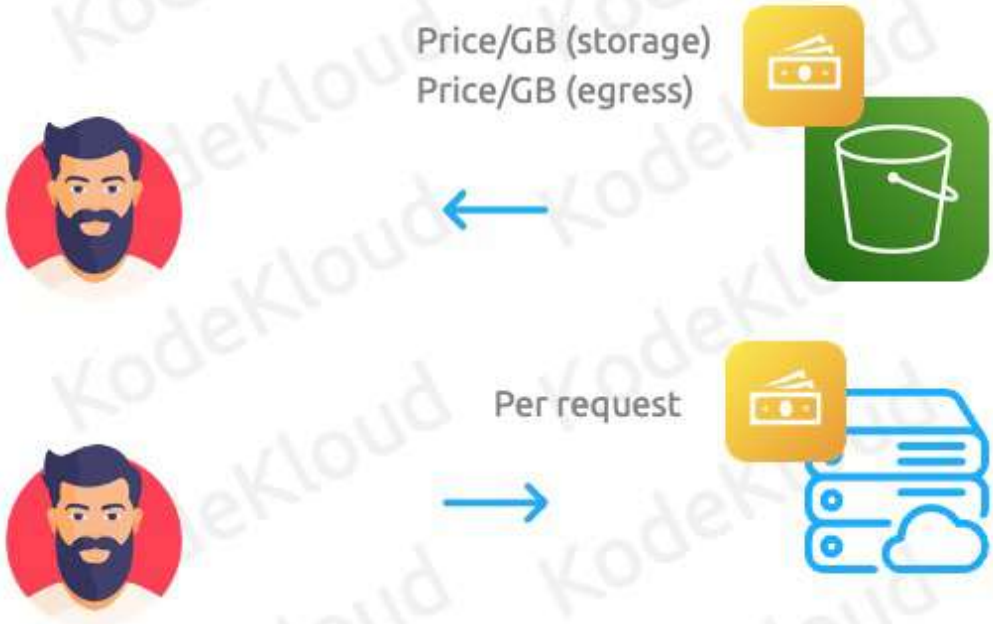
S3 gives the URL through which you can access the website

Note

It is used only for static websites

To customize a domain for your website, the bucket must follow a specific format

Pricing



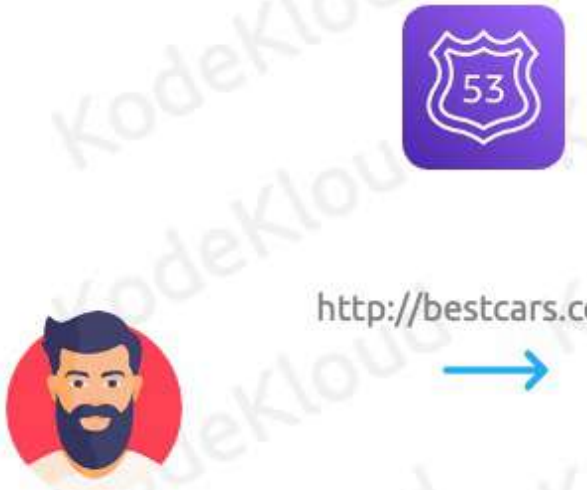
Pricing

	PUT, COPY, POST, LIST requests (per 1,000 requests)	GET, SELECT, and all other requests (per 1,000 requests)	Lifecycle Transition requests into (per 1,000 requests)	Data Retrieval requests (per 1,000 requests)	Data retrievals (per GB)
S3 Standard	$0.005	$0.0004	n/a	n/a	n/a
S3 Intelligent - Tiering *	$0.005	$0.0004	$0.01	n/a	n/a
Frequent Access	n/a	n/a	n/a	n/a	n/a
Infrequent Access	n/a	n/a	n/a	n/a	n/a
Archive Instant	n/a	n/a	n/a	n/a	n/a
Archive Access, Standard	n/a	n/a	n/a	n/a	n/a
Archive Access, Bulk	n/a	n/a	n/a	n/a	n/a
Archive Access, Expedited	n/a	n/a	n/a	$10.00	$0.03
Deep Archive Access, Standard	n/a	n/a	n/a	n/a	n/a
Deep Archive Access, Bulk	n/a	n/a	n/a	n/a	n/a

Custom Domain Name

http://bucketname.s3-website-<region-name>.amazonaws.com

Custom Domain Name



Summary

01 S3 can be used to host static websites; it does not work if website needs server-side logic

02 Charged for files in S3 with an additional fee per HTTP request

03 S3 provides a URL to access website

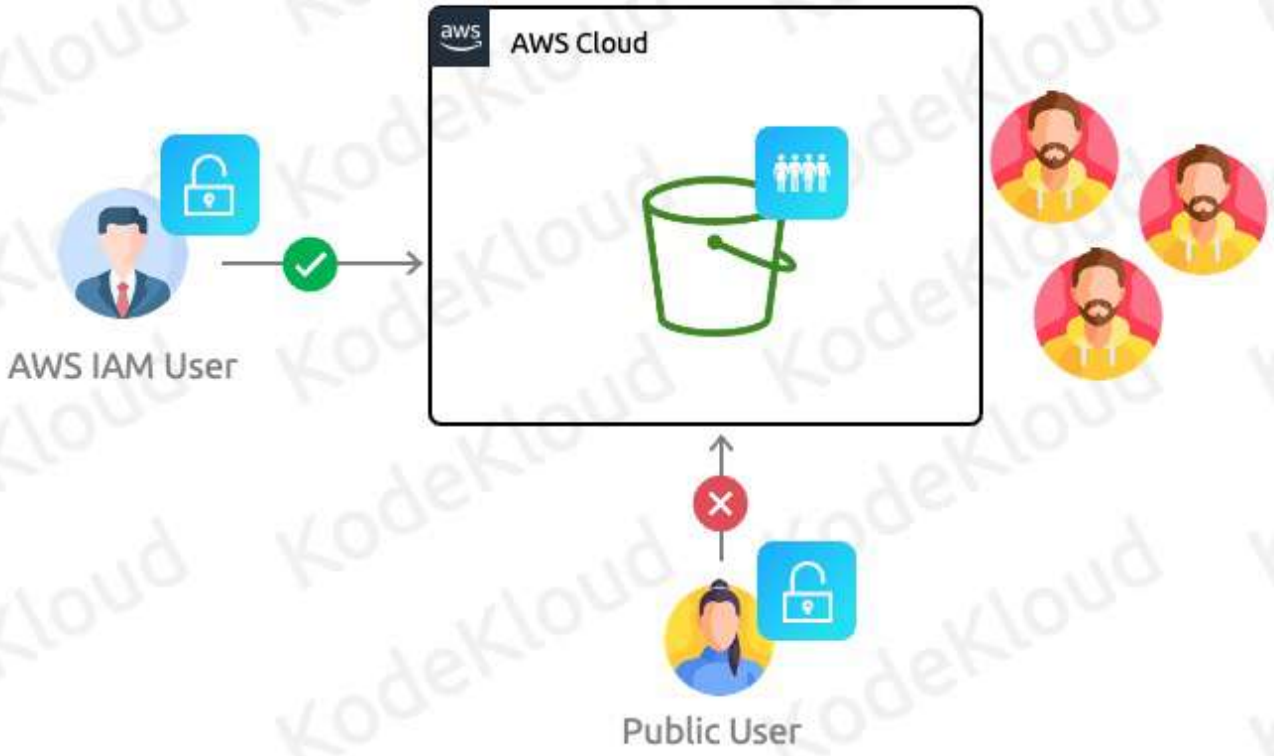
04 If a custom domain is used, then the bucket name needs to match the domain name (example.com)

Service Section

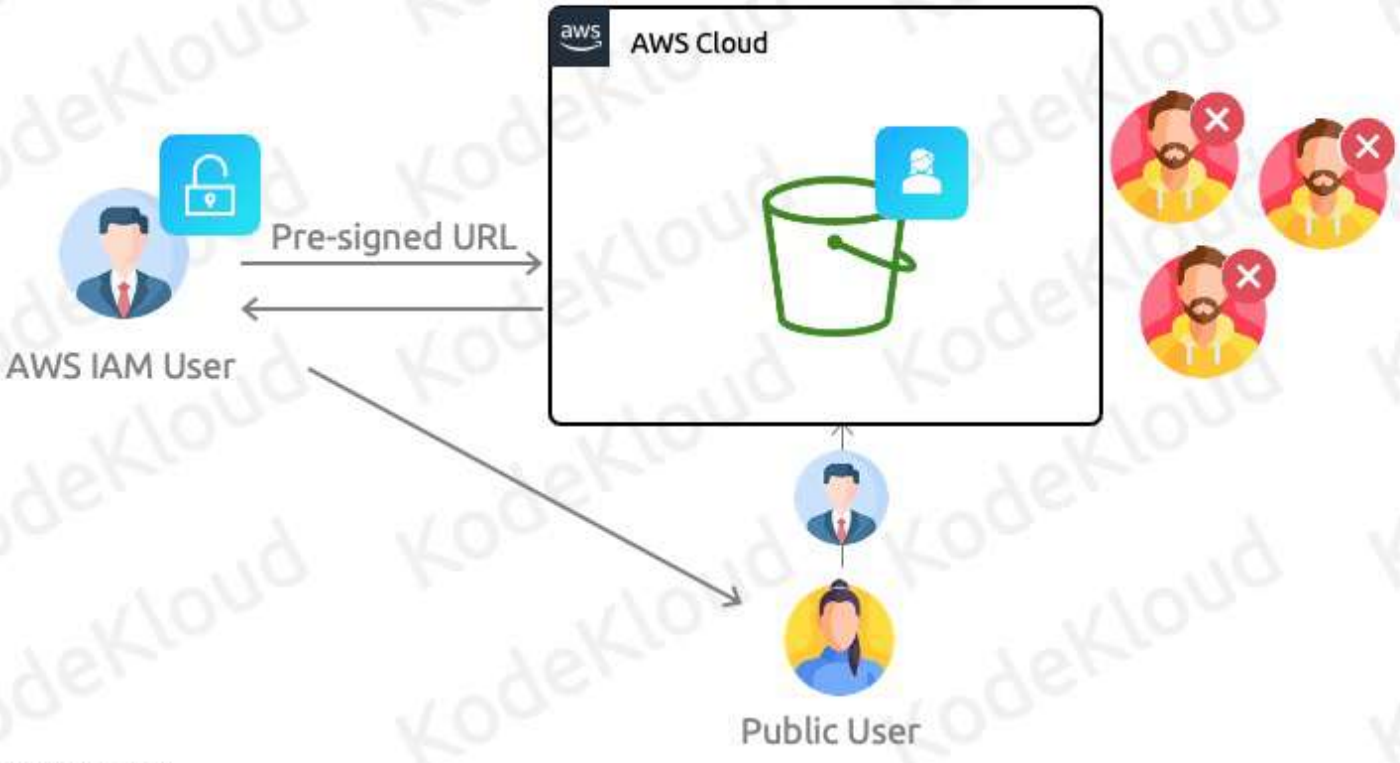
The Power of Object Storage

S3 Pre-Signed URLs

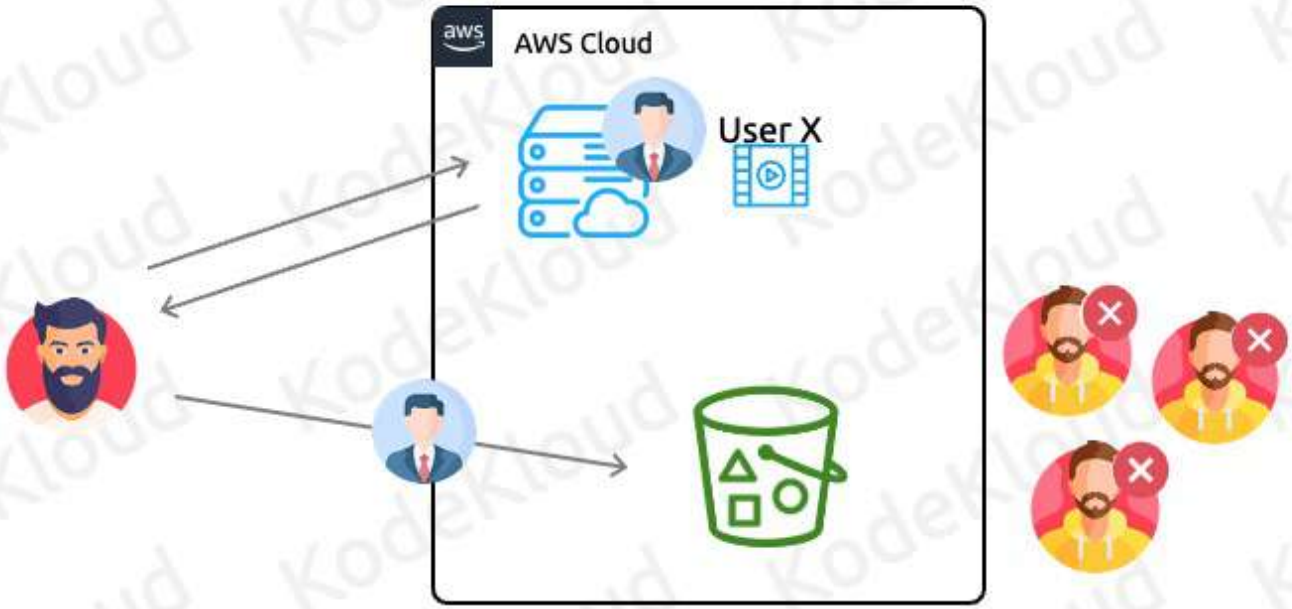
Pre-Signed URLs



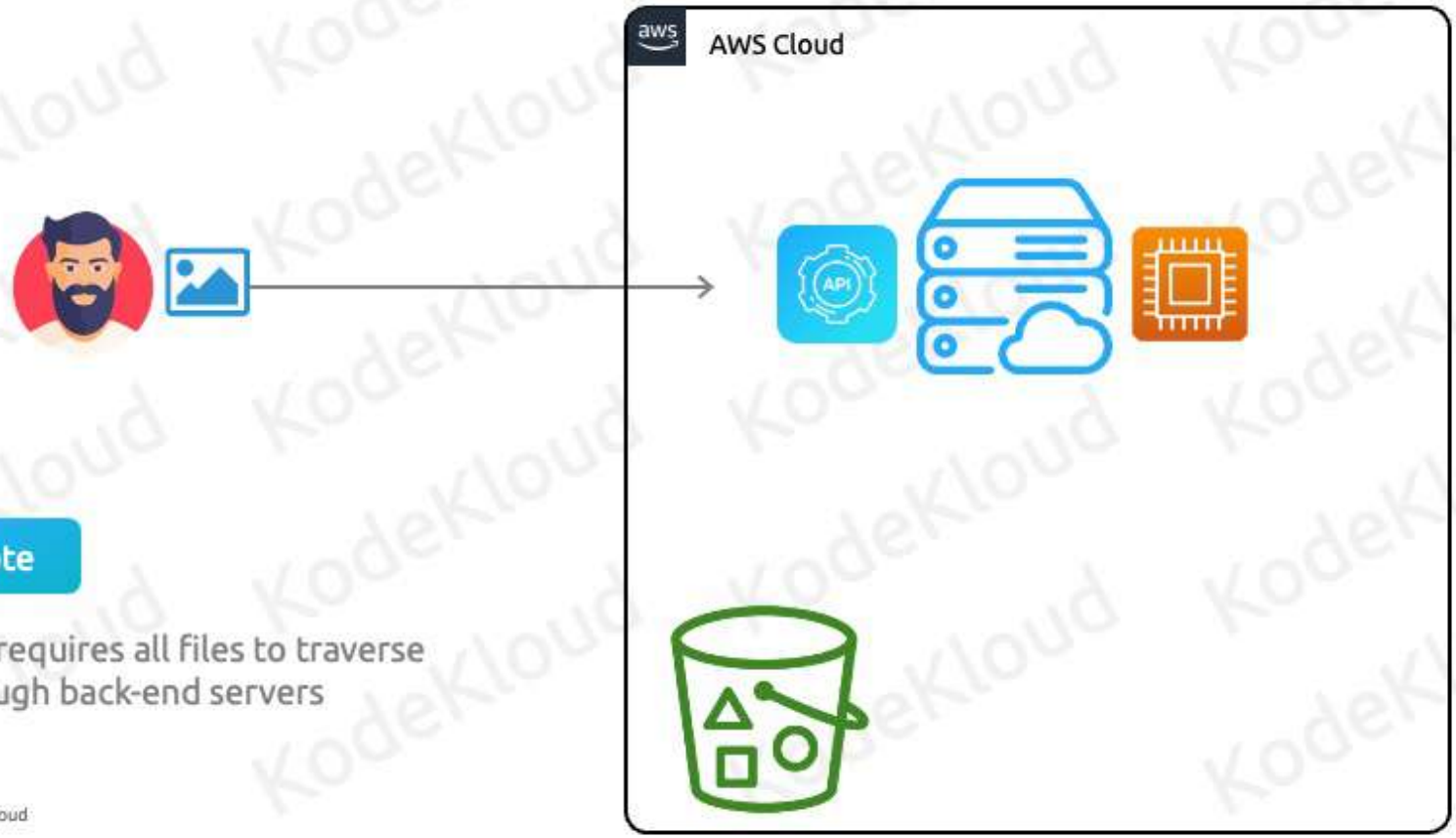
Pre-Signed URLs



Pre-Signed URLs – Use Case



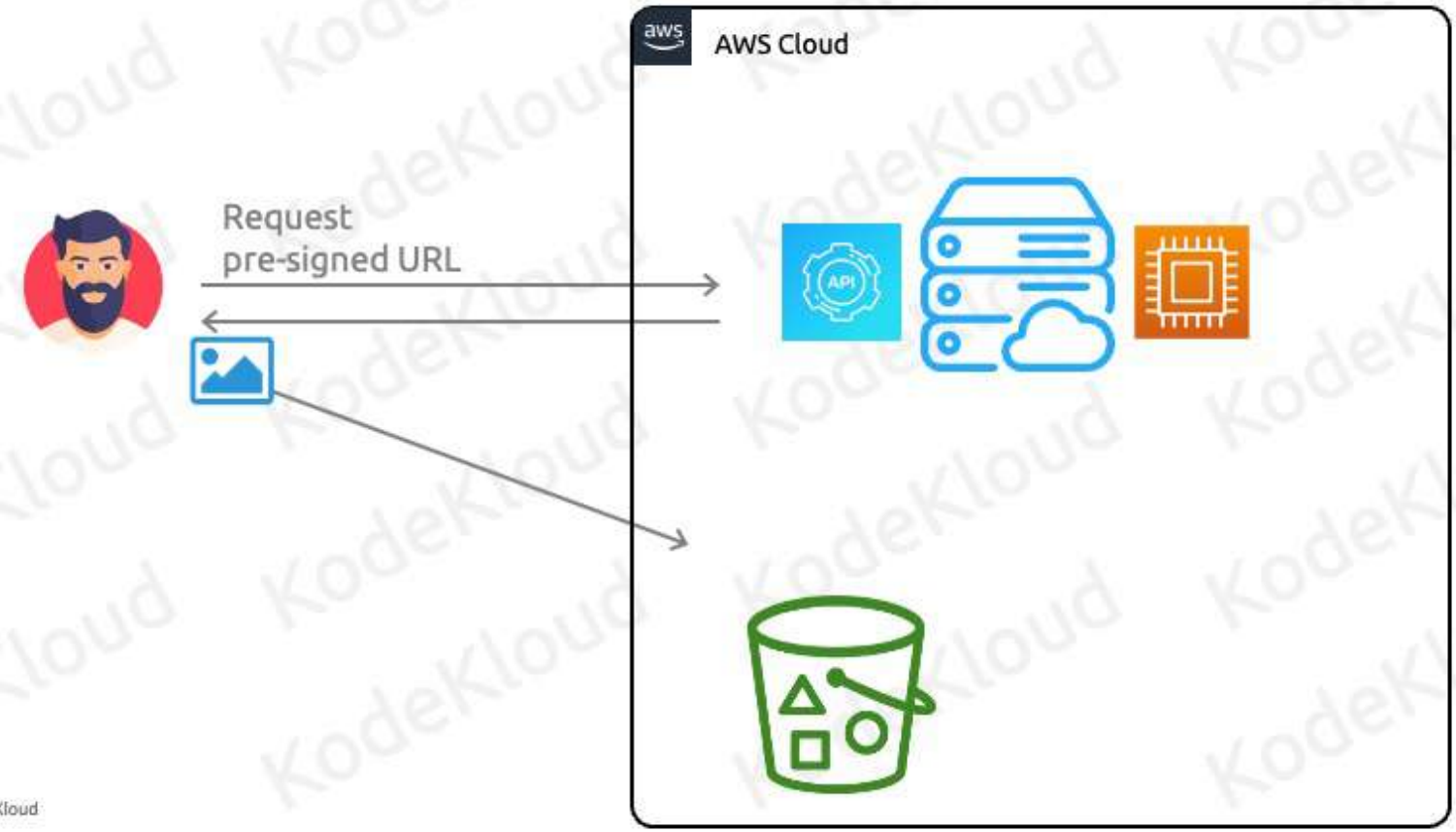
Pre-Signed URLs



Note

This requires all files to traverse through back-end servers

Pre-Signed URLs

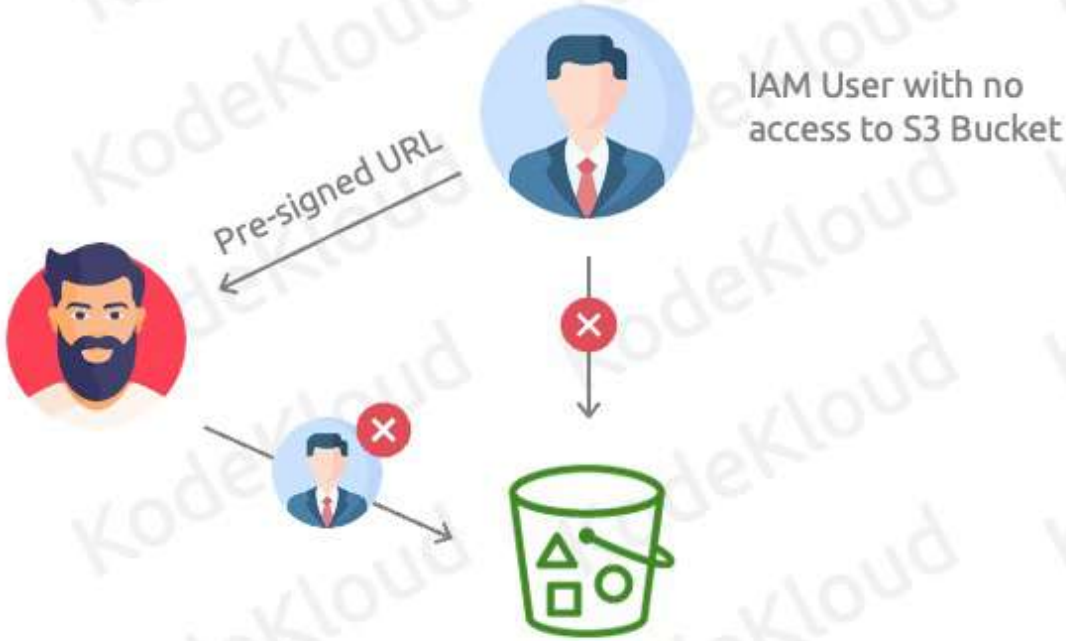


Pre-Signed URLs

Note

- When creating pre-signed URLs, an expiration date must be provided
- Expiration duration of maximum 7 days using an IAM user is provided
- If an IAM user does not have access to an S3 bucket, a pre-signed URL can still be generated using that account
- The pre-signed URL does not give you access to a bucket; however, it allows you to send a request to S3 as the user that generated the URL

Pre-Signed URLs



Summary

01 Pre-signed URLs use security credentials to grant time-limited permissions to download objects

02 When a user clicks on a pre-signed URL, they are performing a request to AWS API with the identity of the user that created the pre-signed URL

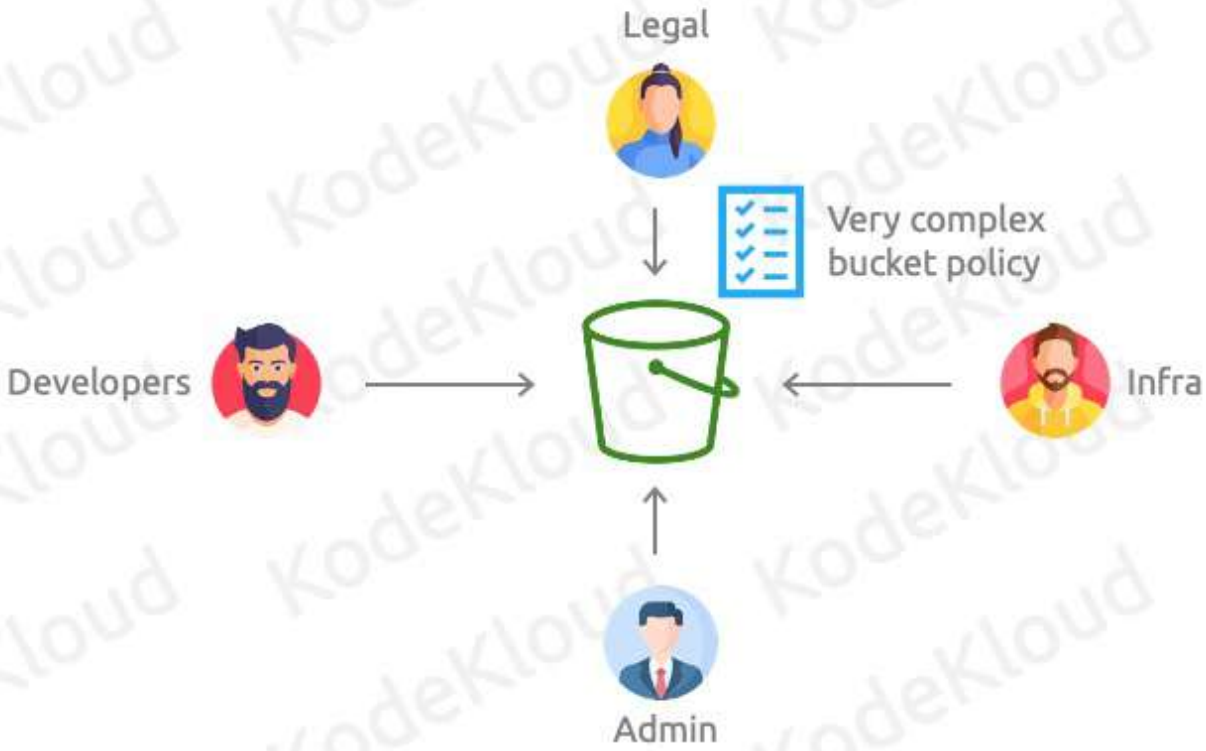
03 If the user that creates a pre-signed URL cannot access an object, then the user that clicks the URL will also not be able to access the object

Service Section

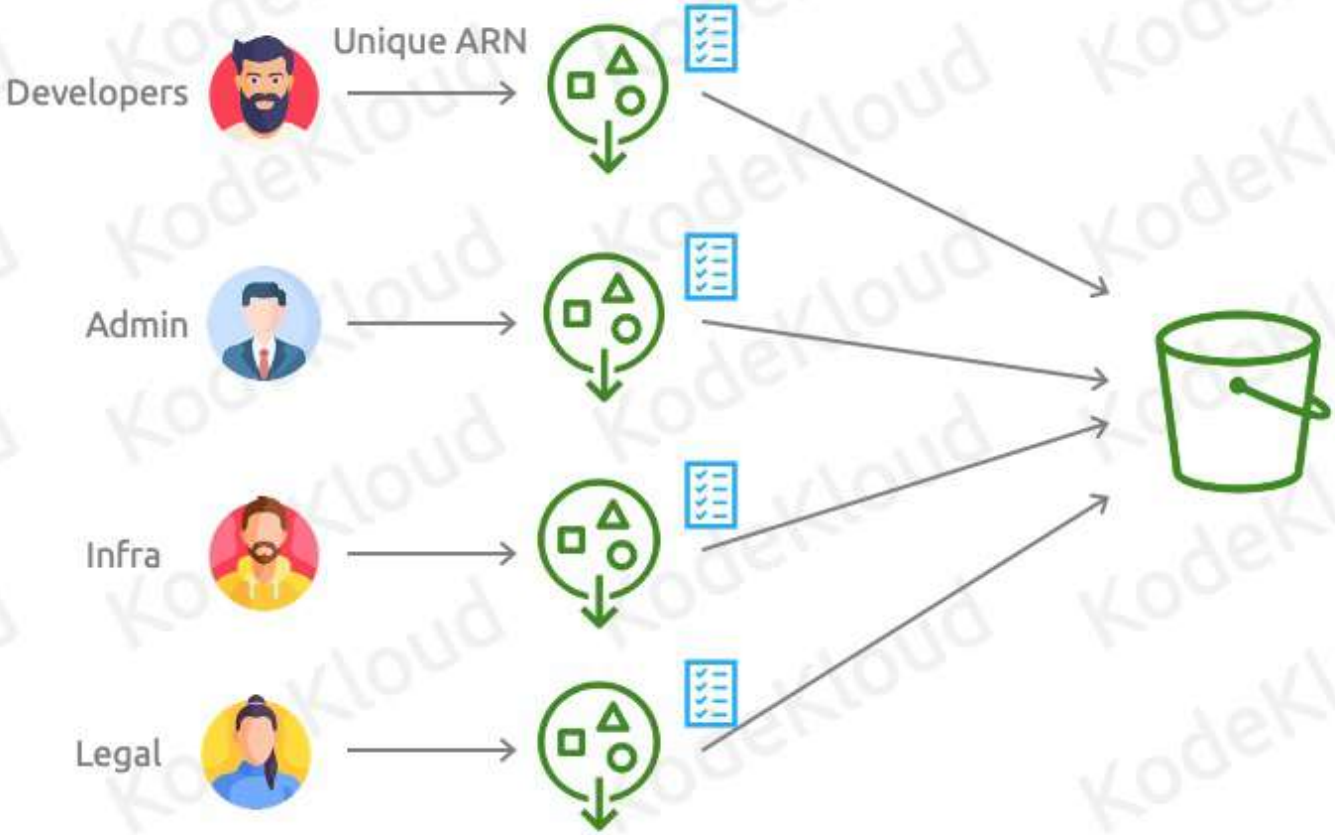
The Power of Object Storage

Access Points

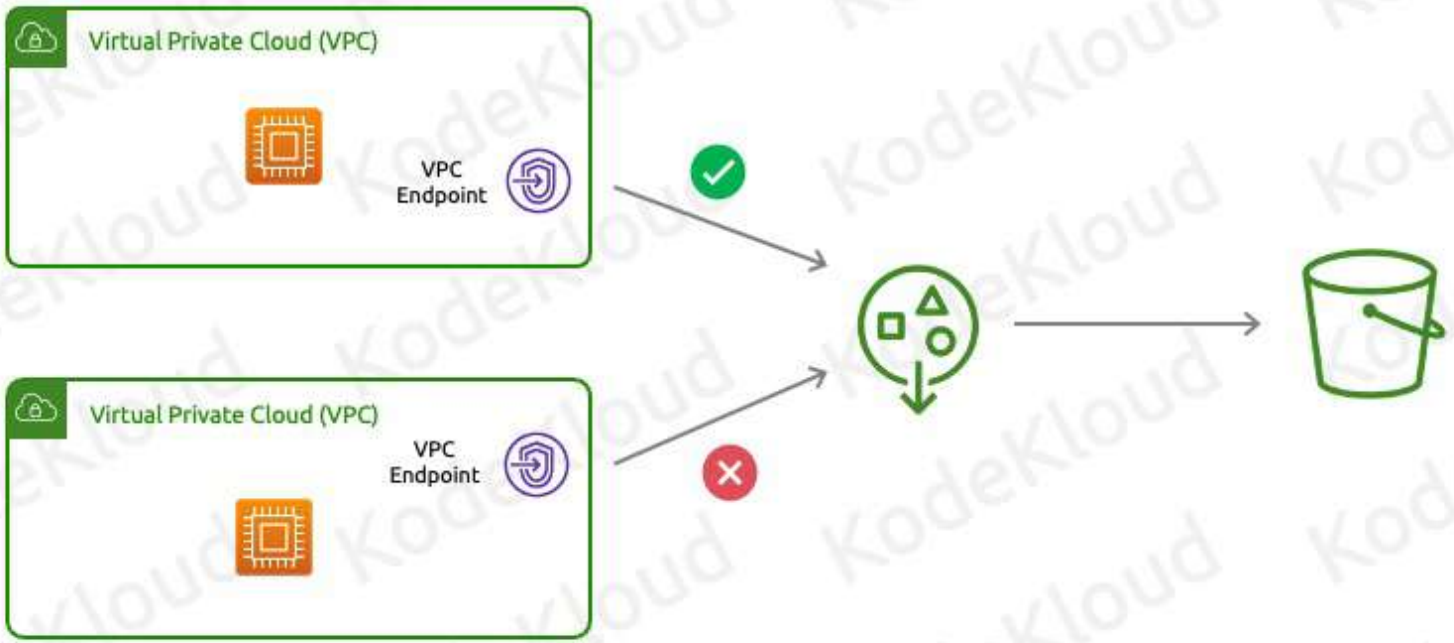
Access Points



Access Points



Access Point Restricting VPCs



Access Point Policy

Same policies need to be copied to the bucket policy

Delegate policies to the access point

Summary

01 Help simplify how you manage access to your S3 buckets

02 Every group/user can be given their own access point which acts as their own view/tunnel into S3 bucket

03 Every access point gets its own ARN, and users instead of going to bucket URL will utilize the Access Point URL

04 Instead of applying policies on buckets, we can move the policies down to the access points, which makes the policies more manageable

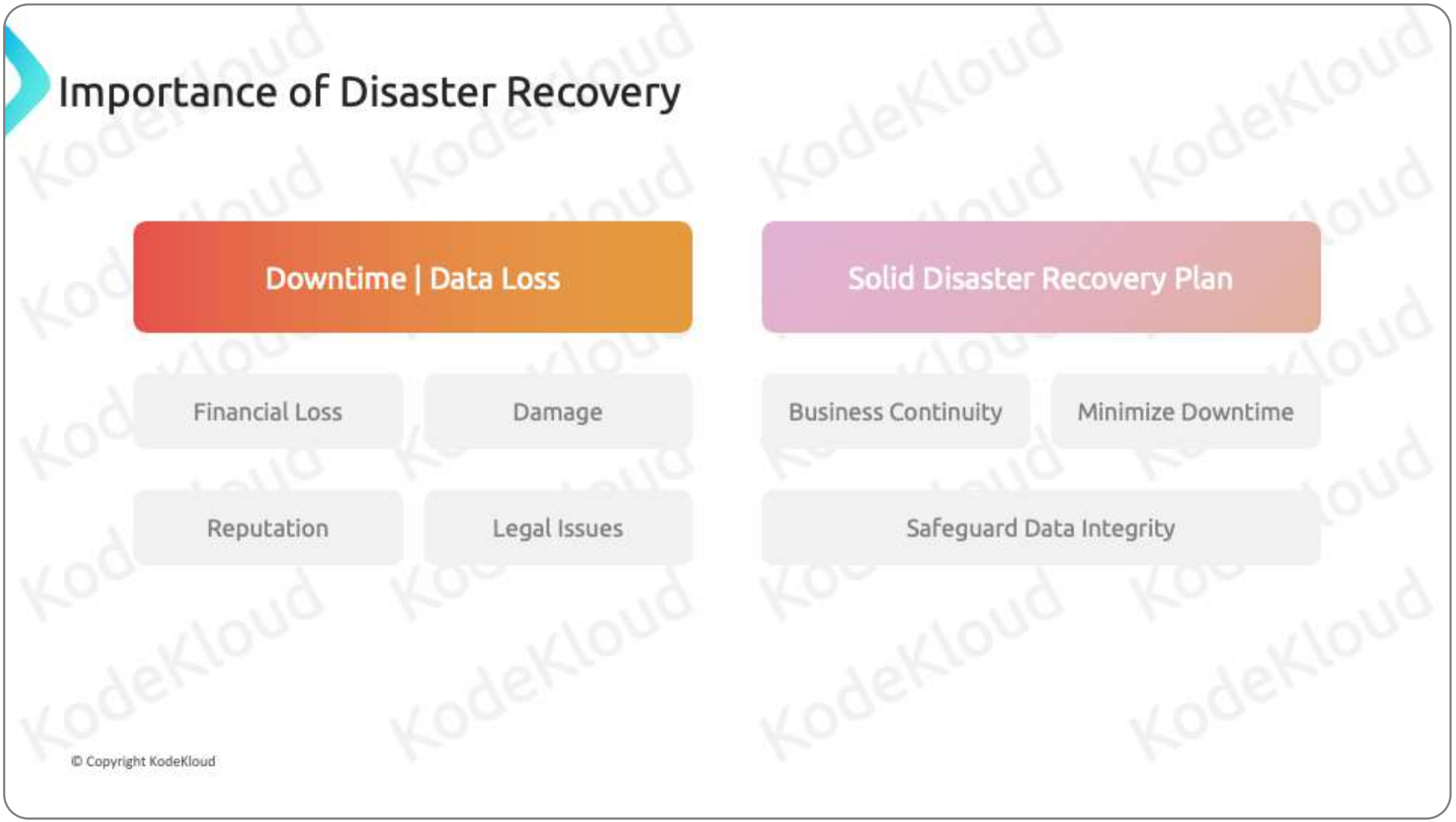
05 Restrict access to bucket to devices in specific VPCs

Service Section

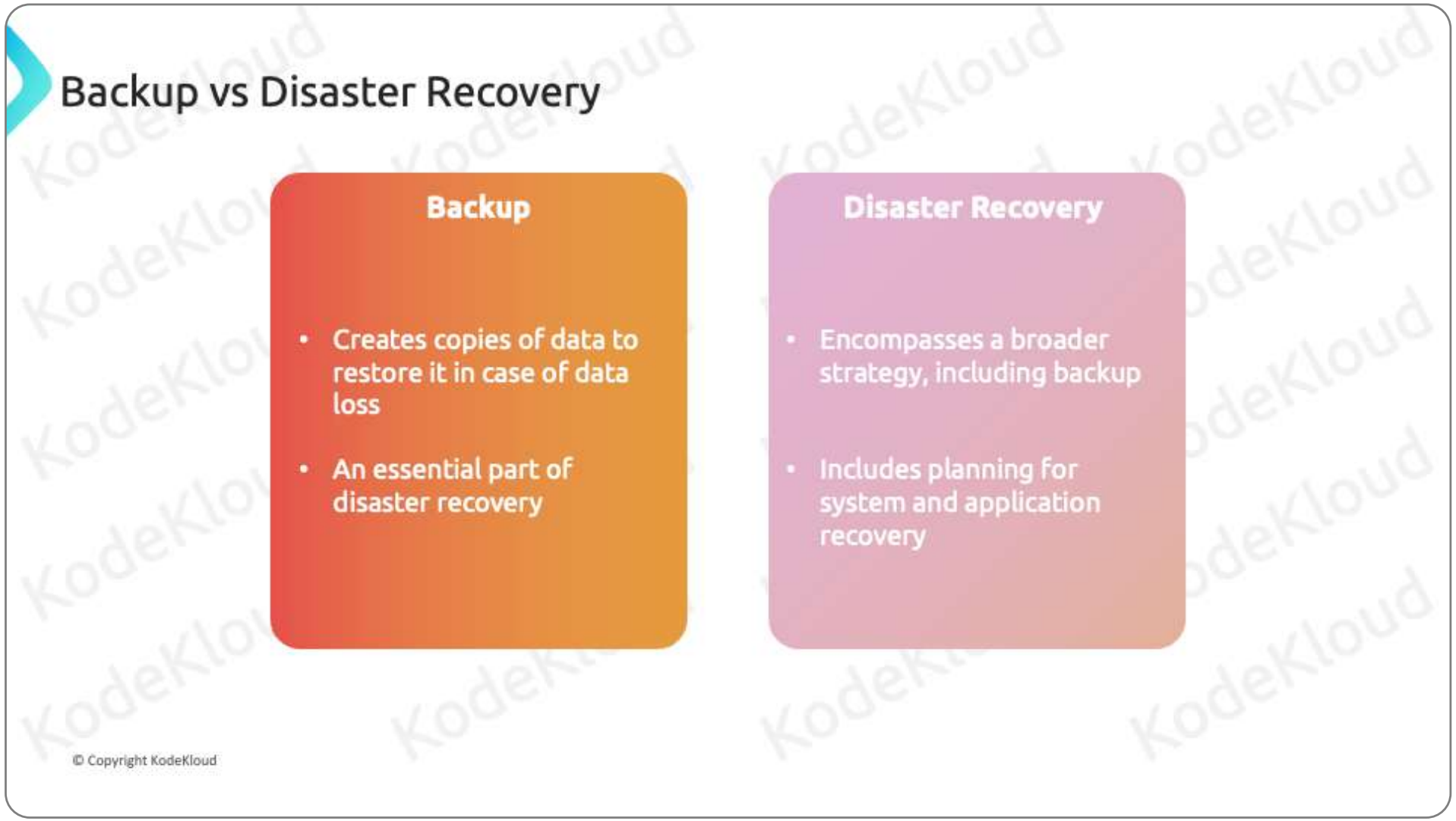
The Power of Storage Manipulation

Backup and Disaster Recovery

- Let's start with the basics. What exactly is disaster recovery (DR)?
- Disaster recovery refers to the process of planning for and responding to events that could cause data loss or system downtime.
- These events, or disasters, can be natural (like earthquakes or floods) or man-made (such as hardware failures or cyberattacks).

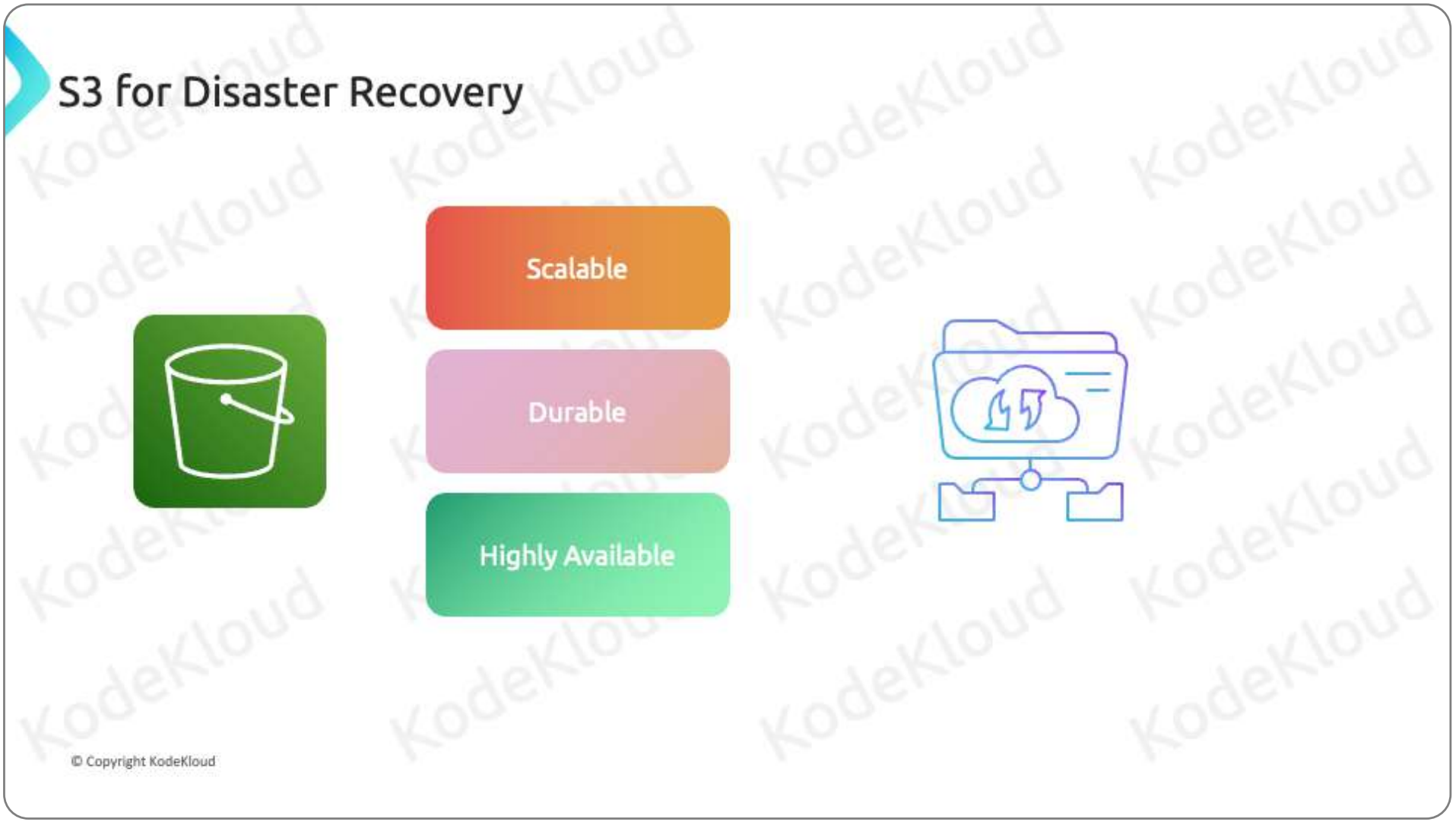


- Why is disaster recovery important?
- Downtime and data loss can have severe consequences for businesses, including financial loss, damage to reputation, and legal issues.
- A solid disaster recovery plan ensures business continuity, minimizes downtime, and safeguards data integrity.

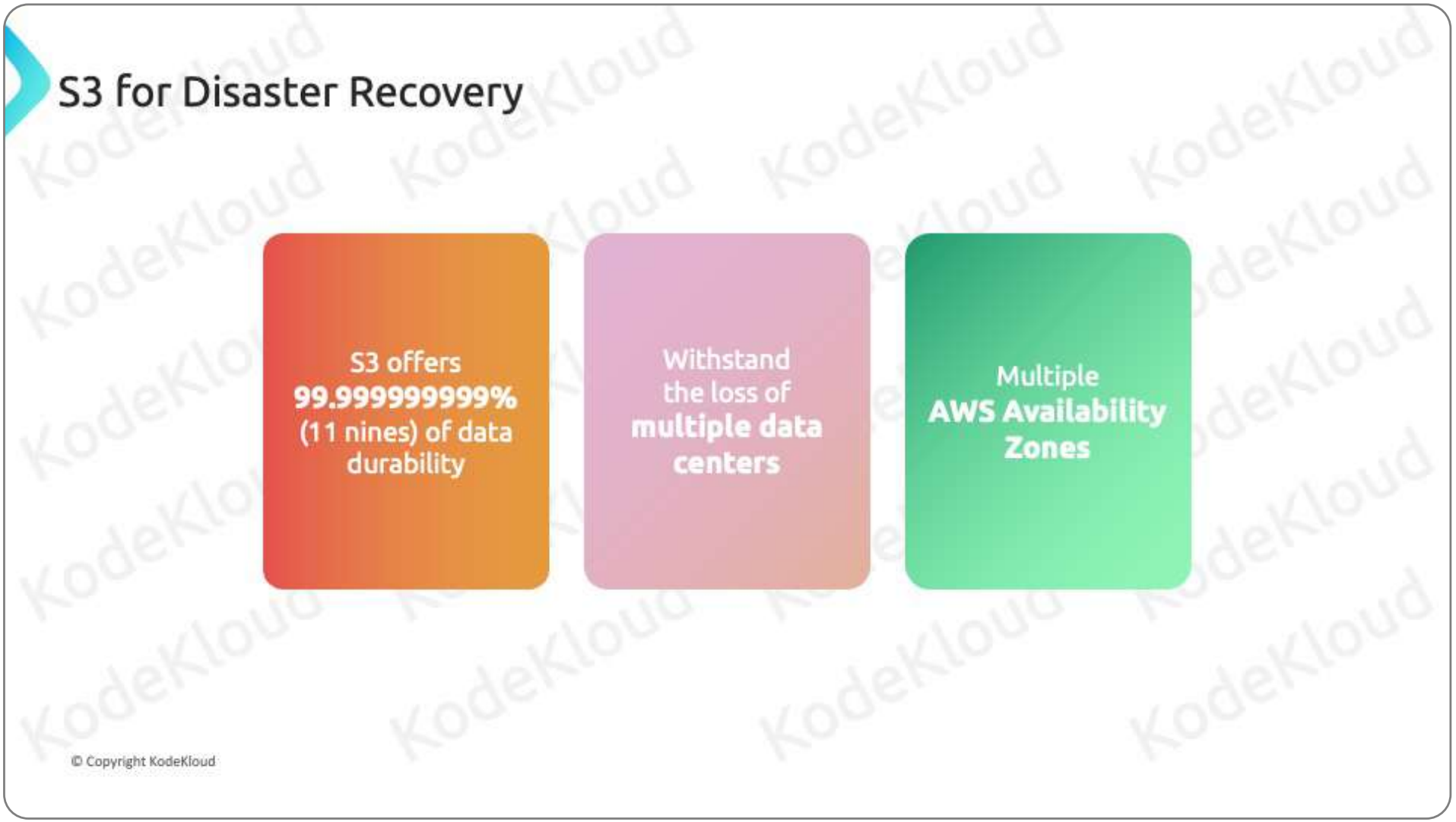


- It's essential to distinguish between backup and disaster recovery.
- Backups involve making copies of data to restore it in case of data loss. It's an essential part of disaster recovery.
- Disaster recovery encompasses a broader strategy, including backup, but also includes planning for system and application recovery.

- How does AWS fit into the picture?
- AWS provides a range of services and tools to help you implement robust disaster recovery strategies.
- These services are designed to be flexible, scalable, and cost-effective.



- One of the foundational services for disaster recovery is Amazon S3, Simple Storage Service.
- S3 provides scalable, durable, and highly available object storage.
- It's ideal for storing backup data, ensuring that your critical data is safe and easily accessible during recovery.



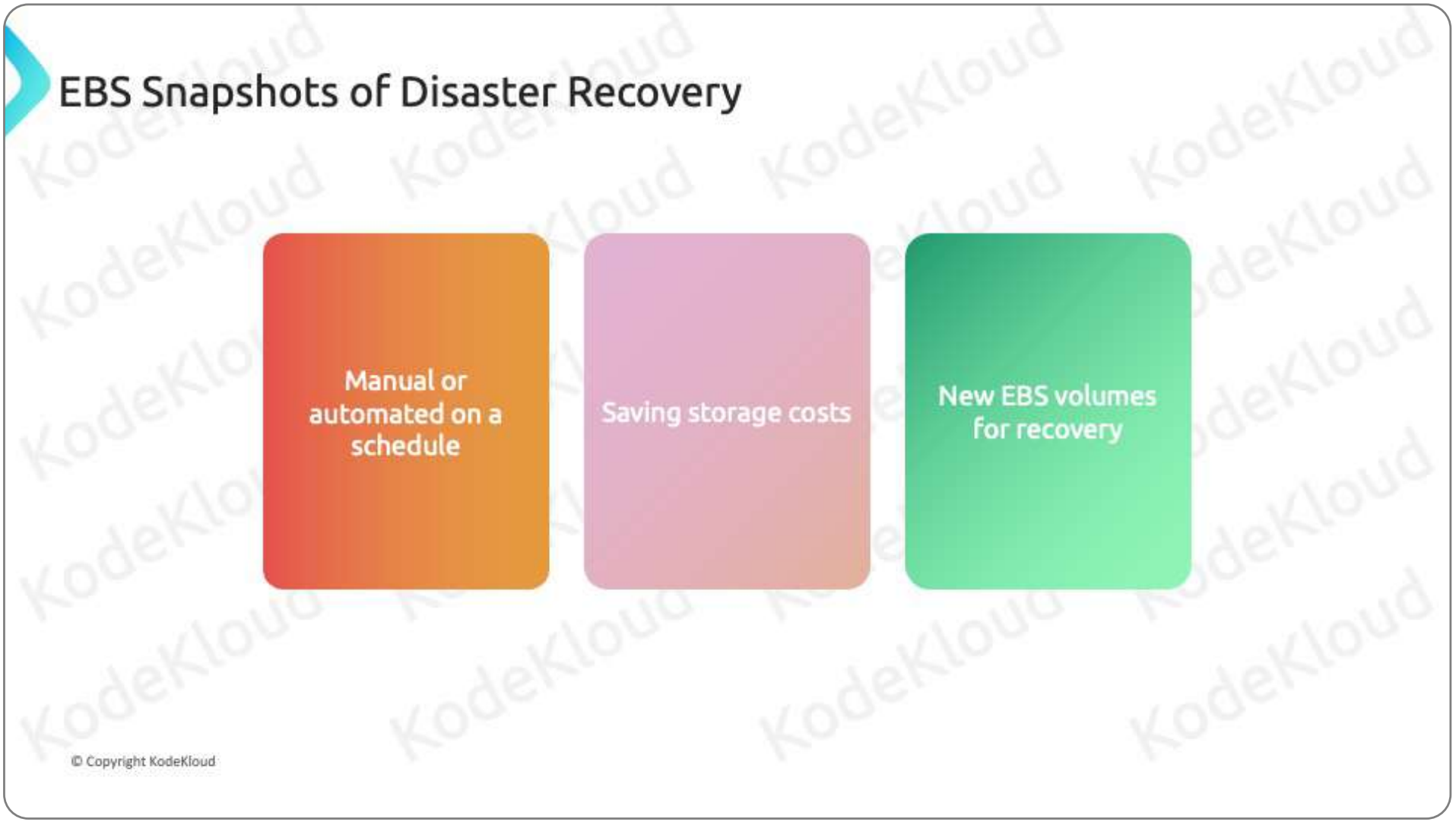
- Let's dive deeper into why Amazon S3 is a strong choice for backup:S3 offers 99.999999999% (11 nines) of data durability.
- It's designed to withstand the loss of multiple data centers, making it highly reliable.
- You can replicate data across multiple AWS Availability Zones for high availability.

EBS Snapshots for Disaster Recovery

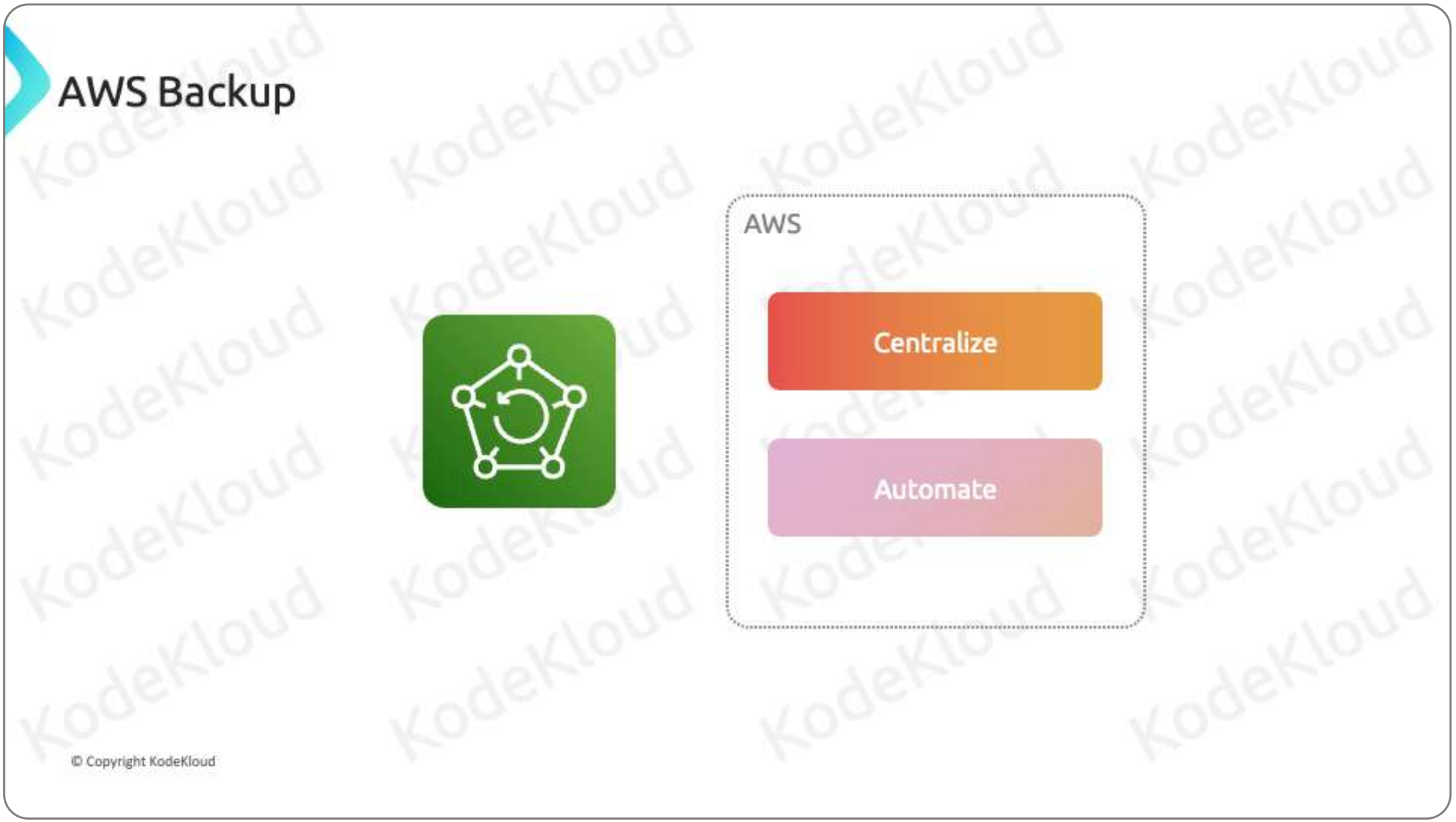
Point-in-Time Copies

EC2 Instance and Data Protection

- Another critical component of disaster recovery is Amazon Elastic Block Store (EBS) snapshots.
- EBS snapshots are point-in-time copies of your EBS volumes.
- They are a fundamental part of protecting your EC2 instances and their data.



- Let's explore how EBS snapshots work:You can create snapshots manually or set up automated snapshots on a schedule.
- Snapshots are incremental, meaning they only store changes since the last snapshot, saving storage costs.
- You can use snapshots to create new EBS volumes for recovery.



So what is AWS backup?

AWS Backup is a fully managed backup service that makes it easy to centralize and automate the backup of your data across various AWS services and resources.

- It provides a single, unified console for managing backups across aWS services
- AWS Backup automates backup scheduling and retention policies, minimizing manual work
- Can backup resources to different regions and even different accounts

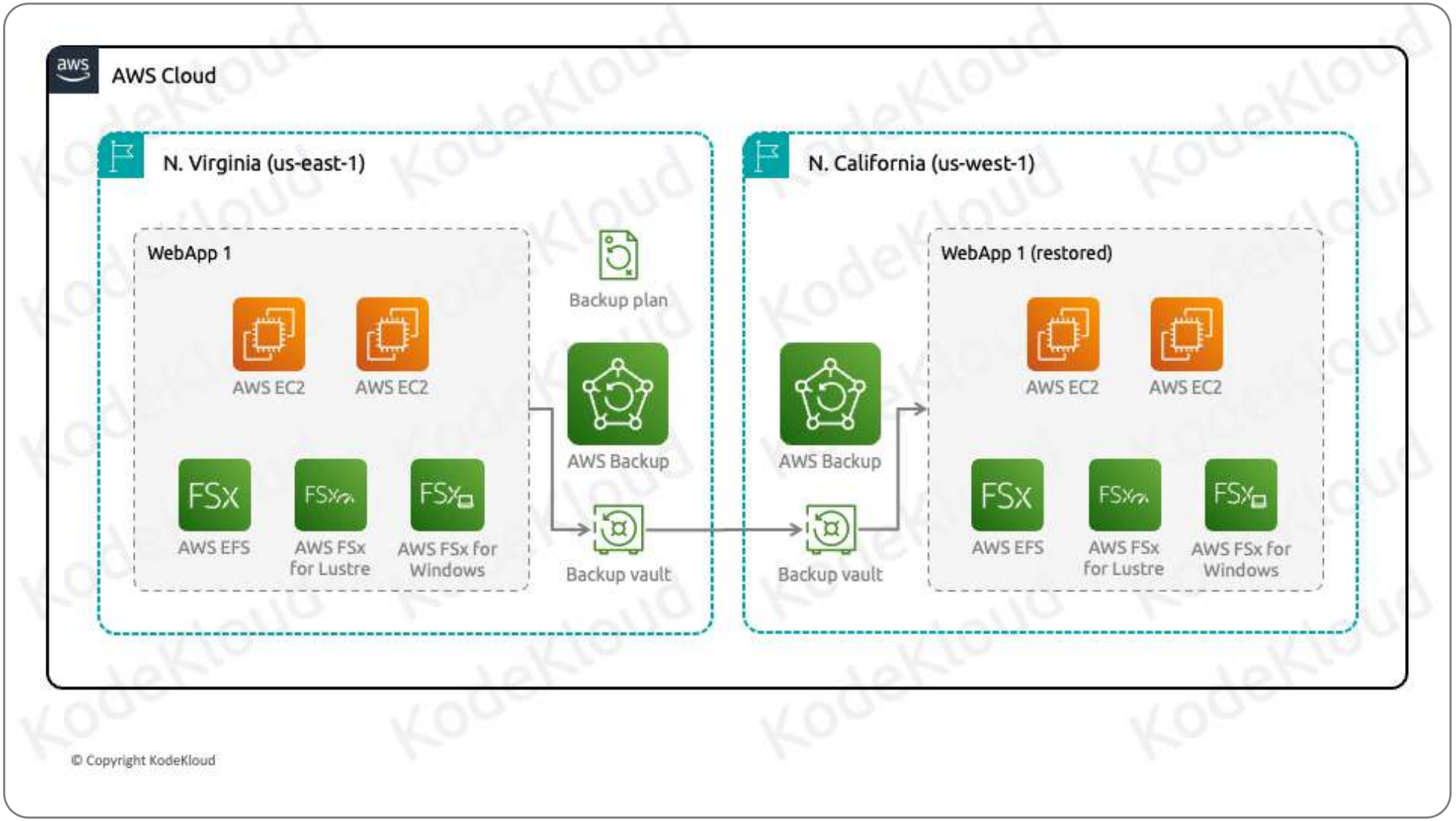
Basic Components and Features

There are 3 main concepts in AWS Backup:

1. Backup Vault: Think of a vault as a container that stores all of your backups. You can create multiple vaults across different regions and accounts to organize your backups
    - Can have vaults in different regions and different accounts
    - It's up to you to decide how you want to manage your vaults. For example you could have a separate vault per

application

2. Backup Plan: Defines the backup schedule, retention policies, selects the backup vault for your resources
3. Recovery Point – Point in time to which data can be restored.



So let’s say in the us-east-1 region we have “app1” that’s deployed that has a couple of ec2 instances and efs volume, and ebs volume, and an RDS instance.

To use AWS backup we will first create a Vault to store our backups. We can decide where this vault will be stored, so we’ll add a vault in the same region.

We will then create a backup plan where we will tell AWS backup that we want to backup all resources in “app1” and send the backups to our newly created vault. We’ll also specify the scheduling of backups so how frequently should it run as well as how long we should retain the backups.

We can also configure another vault in another region or account. And we can create a copy job to copy the backups from us-east-1 region to us-west-1.

So now that we have backups in both regions, we can perform a recovery of any resource in “app1” to either region

[design team] heres a final image of kind of what I’m going for. So we have resources in app1 and then animate in a vault, then animte backup plan, and then animate the second vault in us-west-1

Basic Integrations With Other Services

AWS Backup – Supported Resource Types

Supported resource	Supported resource type
Amazon Elastic Compute Cloud (Amazon EC2)	Amazon EC2 instances (excluding store-backed AMIs)
Amazon Simple Storage Service (Amazon S3)	Amazon S3 data
Amazon Elastic Block Store (Amazon EBS)	Amazon EBS volumes
Amazon DynamoDB	Amazon DynamoDB tables
Amazon Relational Database Service (Amazon RDS)	Amazon RDS database instances (including all database engines); Multi-Availability Zone clusters
Amazon Aurora	Aurora clusters
Amazon Elastic File System (Amazon EFS)	Amazon EFS file systems
FSx for Lustre	FSx for Lustre file systems
FSx for Windows File Server	FSx for Windows File Server file systems
Amazon FSx for NetApp ONTAP	FSx for ONTAP file systems
Amazon FSx for OpenZFS	FSx for OpenZFS file systems
AWS Storage Gateway (Volume Gateway)	AWS Storage Gateway volumes
Amazon DocumentDB	Amazon DocumentDB clusters
Amazon Neptune	Amazon Neptune clusters
Amazon Redshift	Amazon Redshift clusters
Amazon Timestream	Amazon Timestream clusters

•AWS Organizations manage and monitor backup, restore, and copy jobs across multiple AWS accounts.
•Amazon EventBridge to view and monitor AWS Backup events.
•AWS CloudWatch to track metrics, create alarms, and view dashboards.
•AWS CloudTrail to monitor AWS Backup API calls.
•Amazon SNS to subscribe and notify you of AWS Backup events.

Summary

01 Disaster recovery refers to the process of planning for and responding to events that could cause data loss or system downtime

02 A solid disaster recovery plan ensures business continuity, minimizes downtime, and safeguards data integrity

03 Disaster recovery encompasses a broader strategy, including backup, and also includes planning for system and application recovery

04 AWS provides several services that can be utilized to assist in Backups and Disaster Recovery (S3, EBS snapshots, AWS Backup)

Summary

05 AWS Backup is a fully managed backup service that makes it easy to centralize and automate the backup of your data across various AWS services and resources

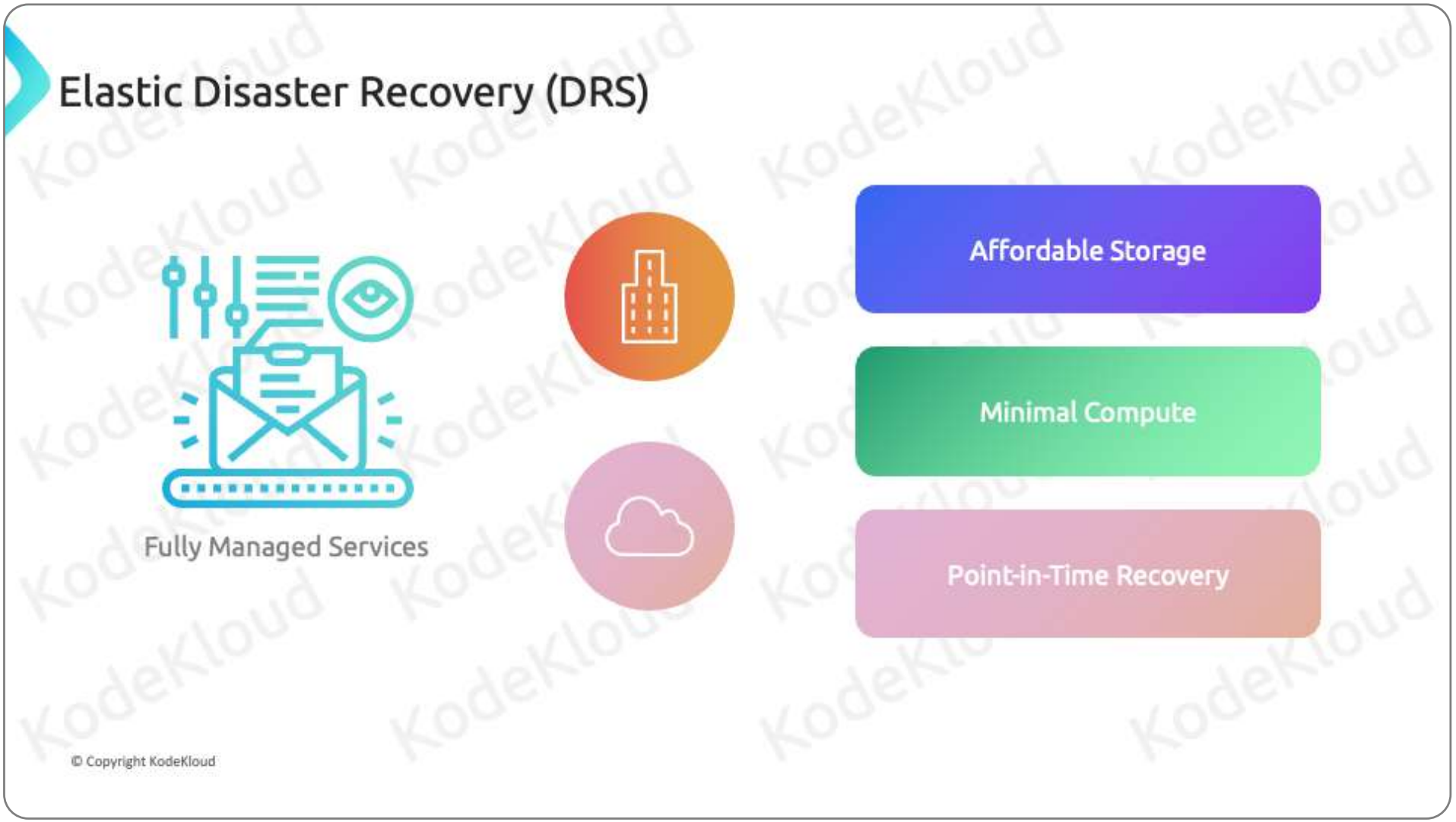
06 AWS Backup has 3 main concepts – Backup Vault, Backup Plan, and Recovery Point

07 AWS Backup can perform backups across a wide variety of services including EC2, EBS, EFS, and RDS

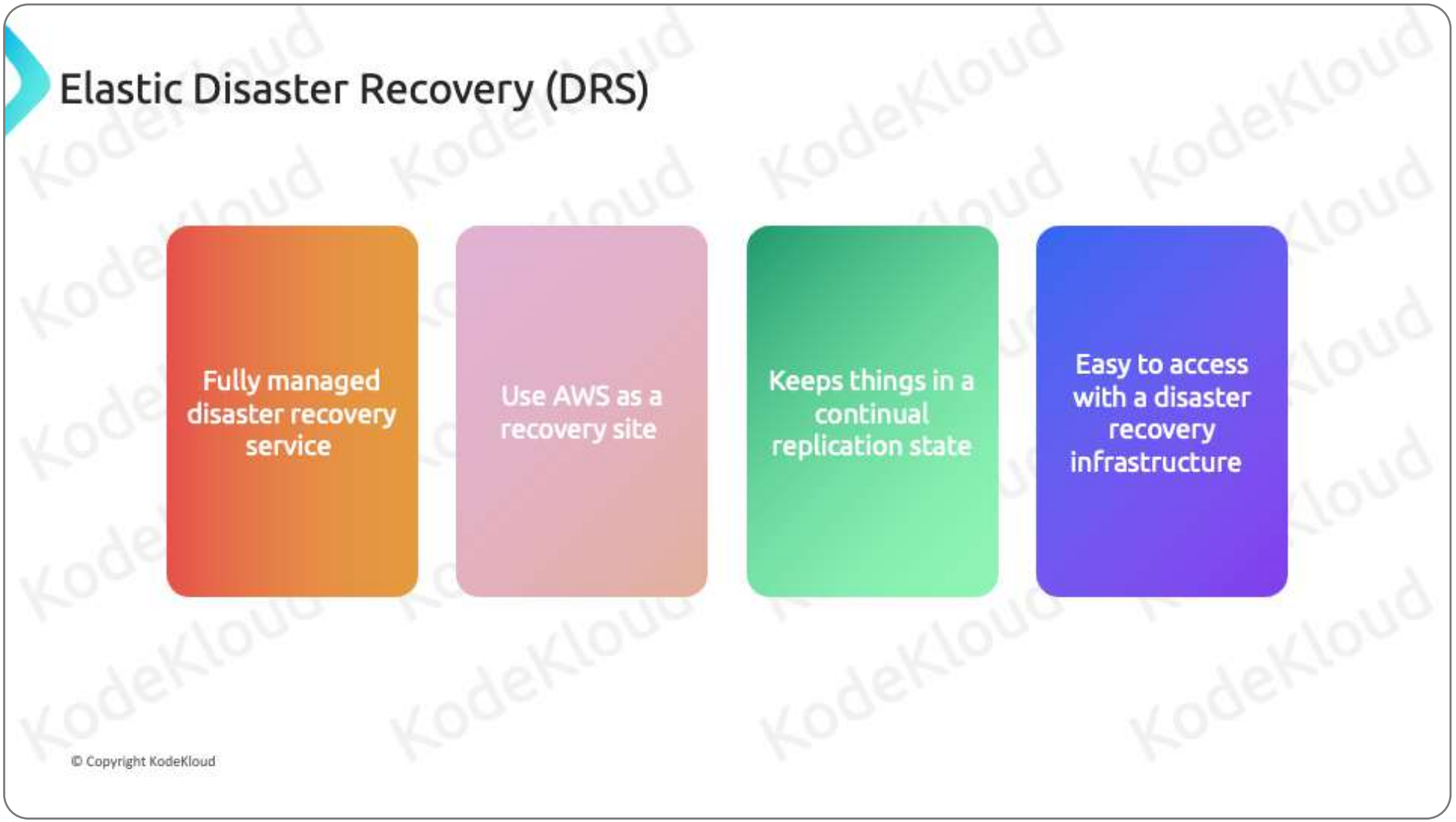
Service Section

The Power of Storage Manipulation

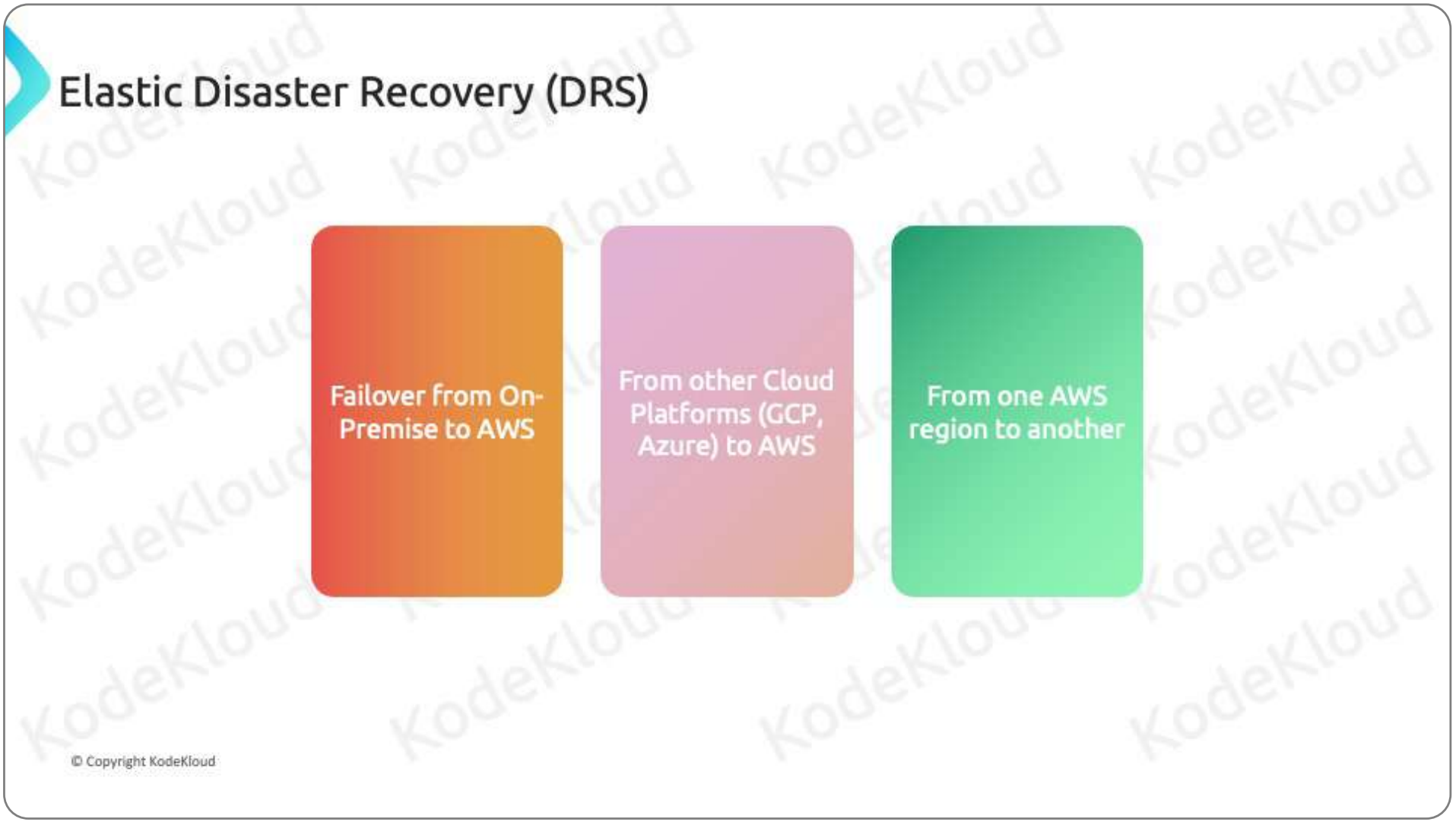
Elastic Disaster Recovery



- fully managed service that provides fast, reliable recovery of on-premises and cloud-based applications using affordable storage, minimal compute, and point-in-time recovery.

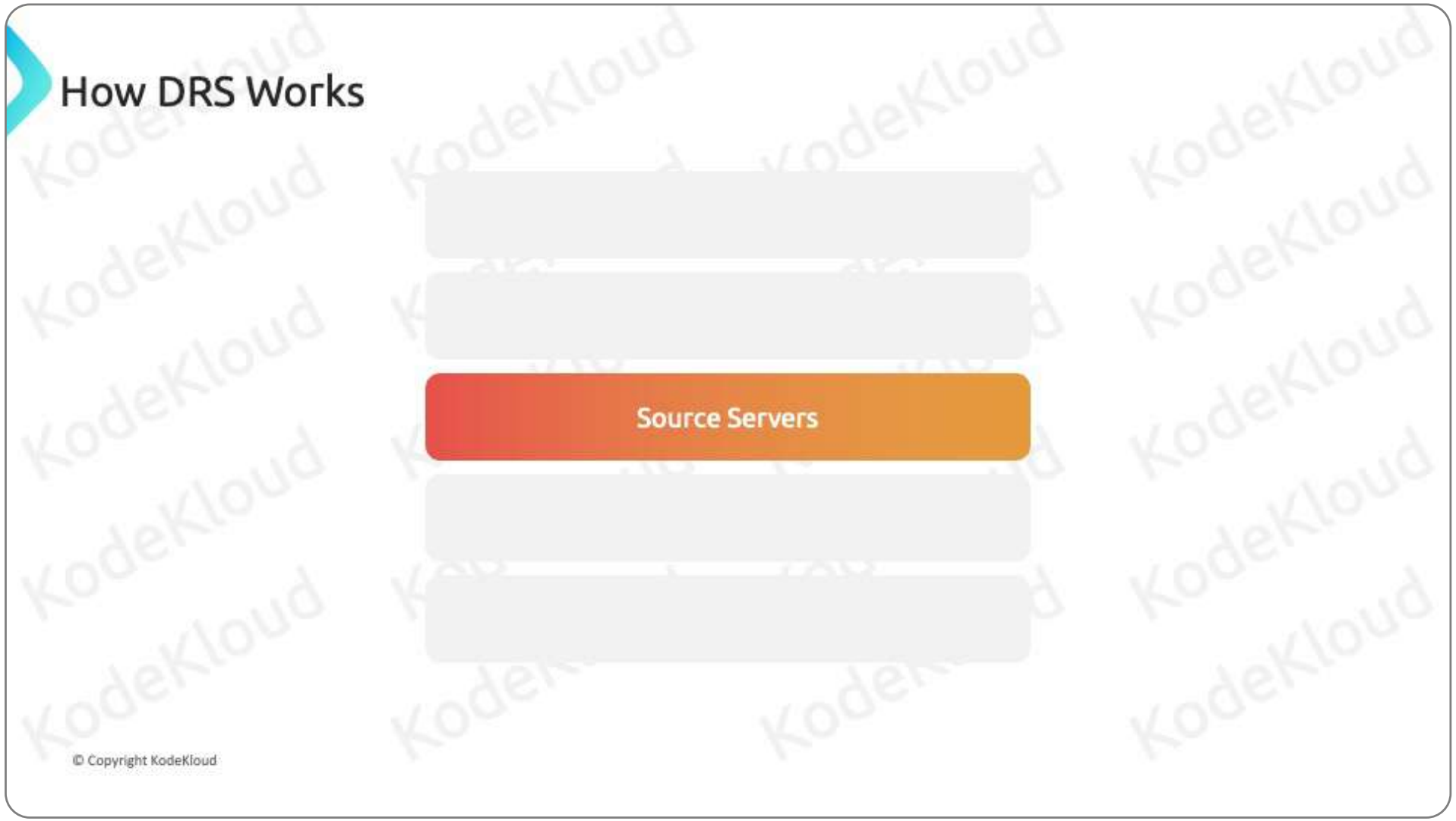


- Fully managed disaster recovery service for physical, virtual and cloud based servers.
- Customers can use AWS as recovery site instead of investing in on-premises disaster recover infrastructure.
- Elastic Disaster Recovery keeps customers operating system, application, and databases in a continual replication state.
- So all the data is backed up and stored on AWS and and an entire disaster recovery infrastructure can be spun up with a click of a button using all the replicated data

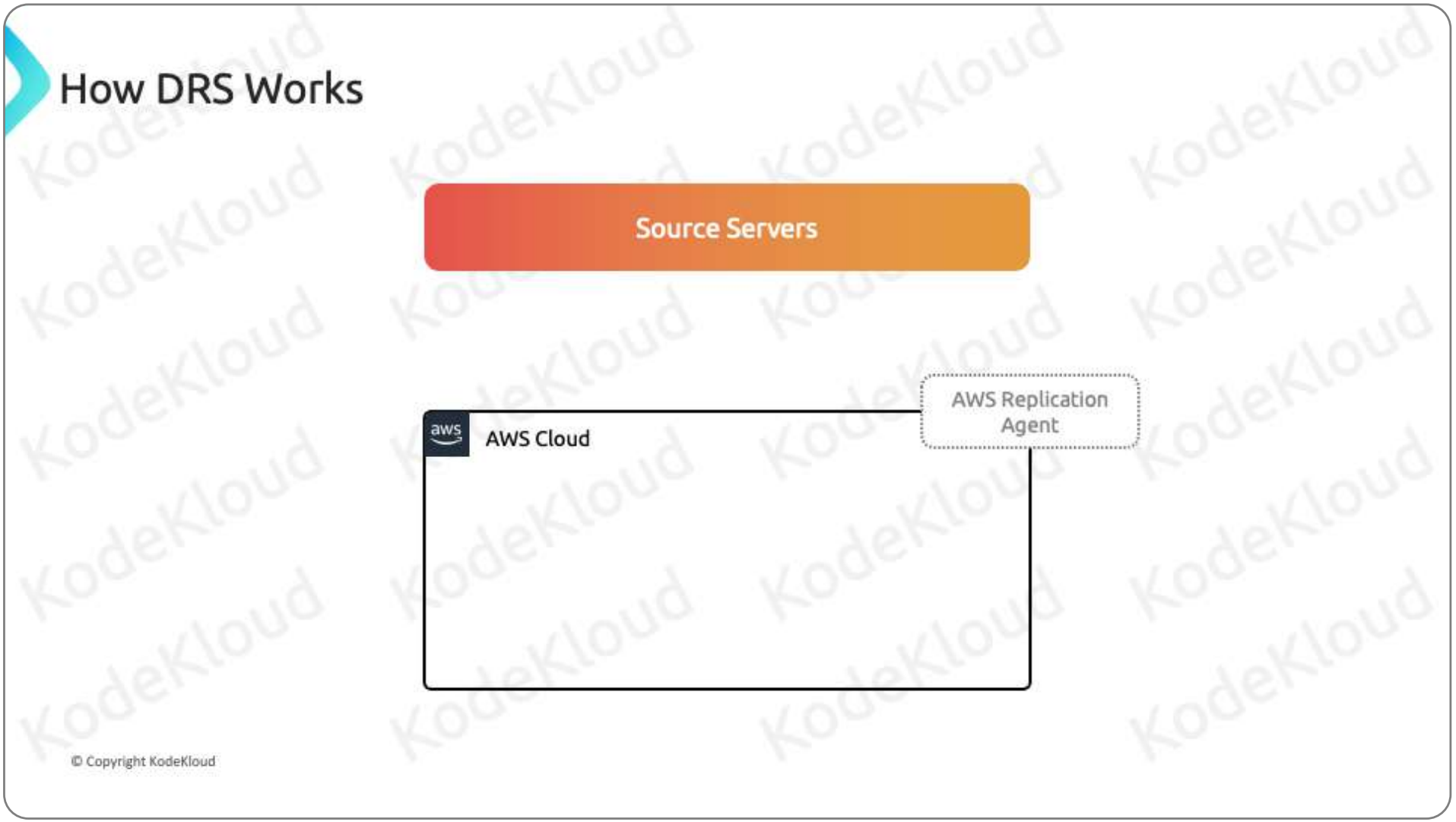


We can perform disaster recovery across a variety of different platform.

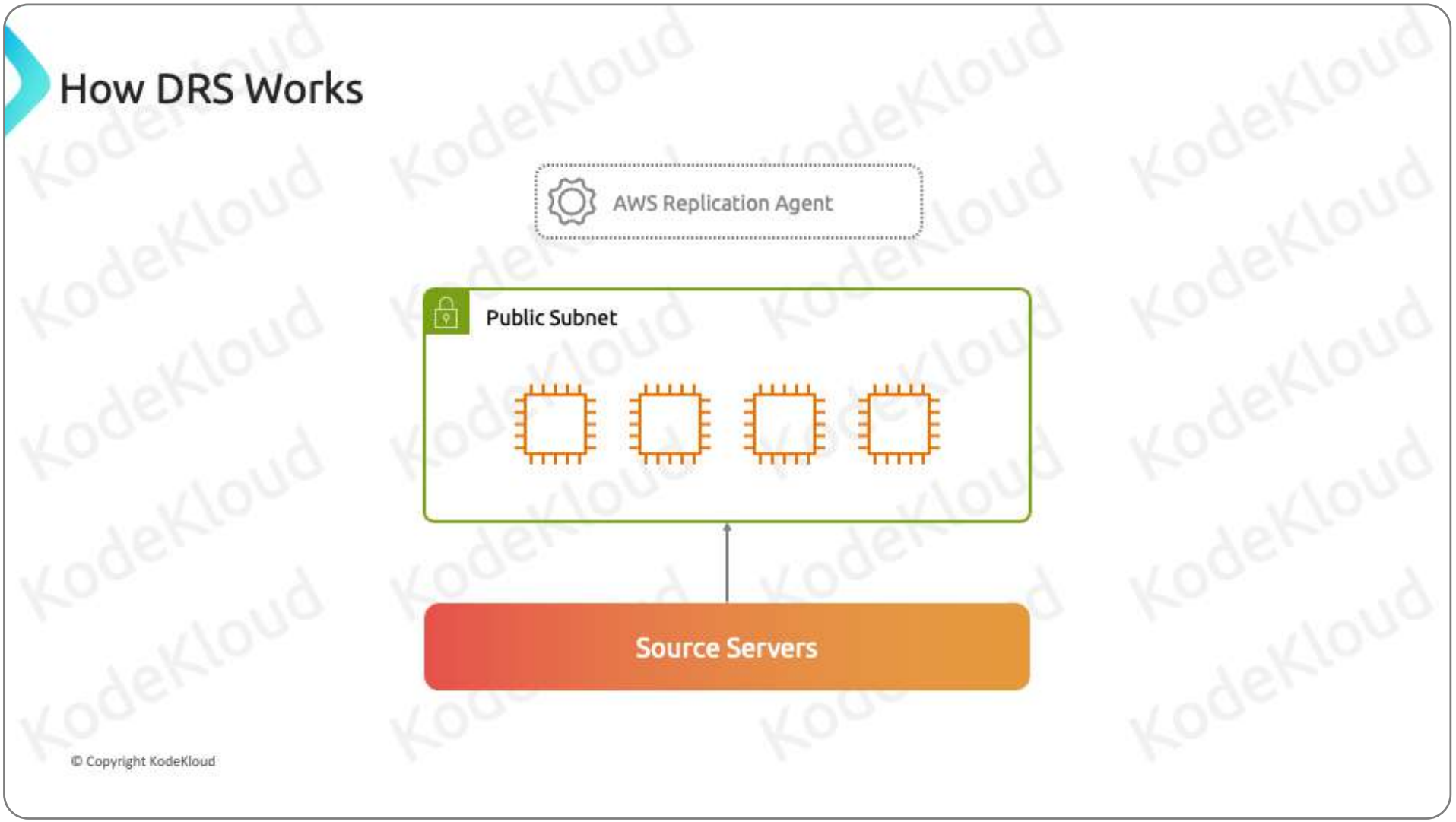
Failover from on-premise to AWS
From other cloud platforms(GCP, Azure) to AWS
From one AWS region to another



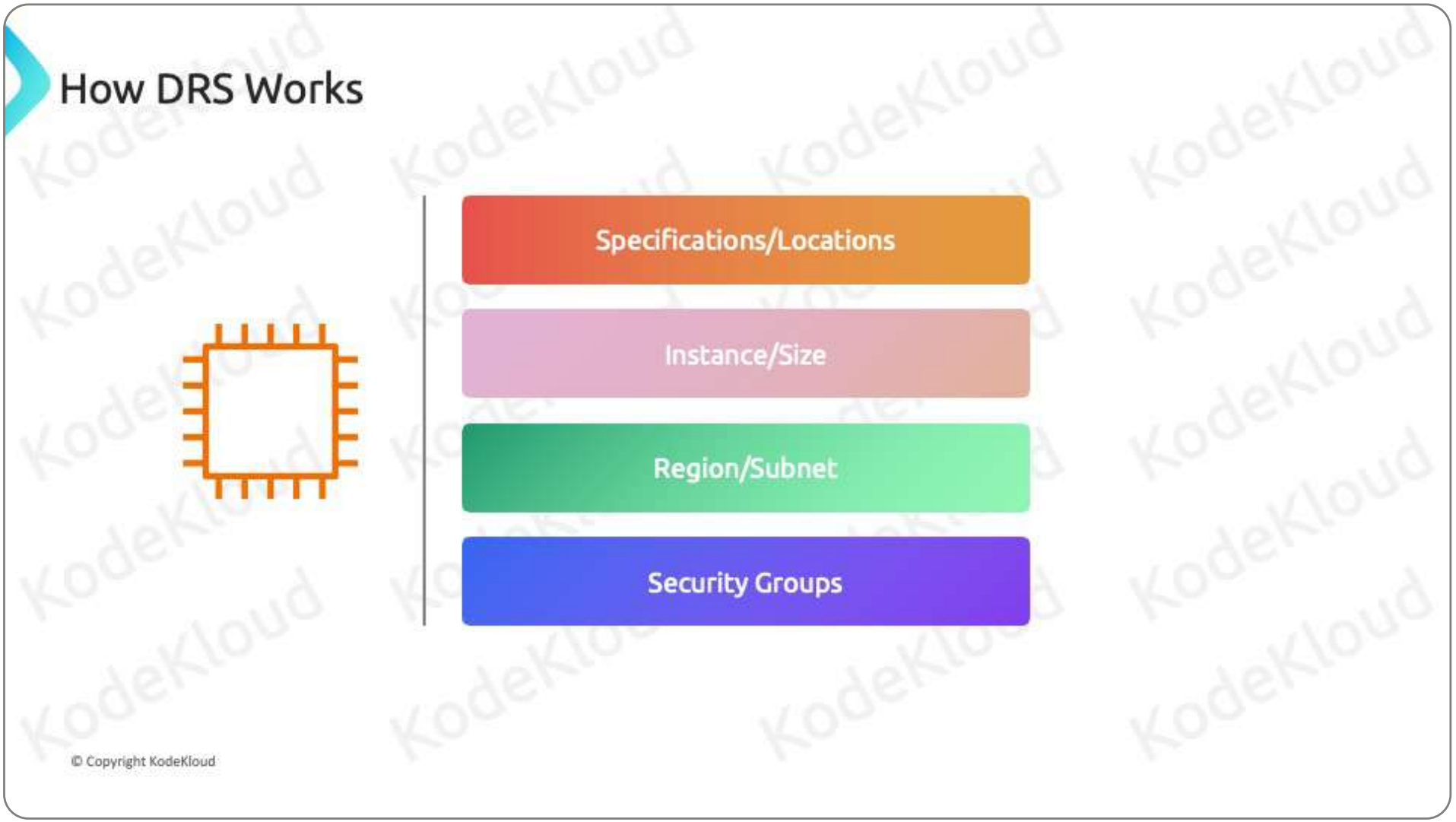
First we have to decide what devices/data should be monitored for failover. These devices are referred to as source servers.



For source servers to be able to replicate data to AWS they'll need to have the AWS Replication Agent installed on them.



Then we have to define our replication settings. Here we will define where our staging area is going to be. So this is going to be the subnet and the respective ec2 instances that will receive the replicated data from our source server.



Finally we have to define launch settings which is the ec2 settings for your recovery servers(these are the servers you will failover to during a disaster)
You'll specify the ec2 specs/locations

- Ec2 instance/size
- Region/subnet

- Security groups

So when we experience a disaster, we can go to the console and perform a recovery. This will create the recovery instances in the specified region/subnet and make sure it has access to all of the replicated data.

From there once we determine that the original issue is gone, we can perform a fallback to the original source servers

Summary

01 A fully managed disaster recovery service for physical, virtual, and cloud-based servers

02 Customers can use AWS as a recovery site instead of investing in on-premises disaster recovery infrastructure

03 Source servers represent the servers/data that we want to replicate

04 The staging area is the location where AWS will receive the replicated data

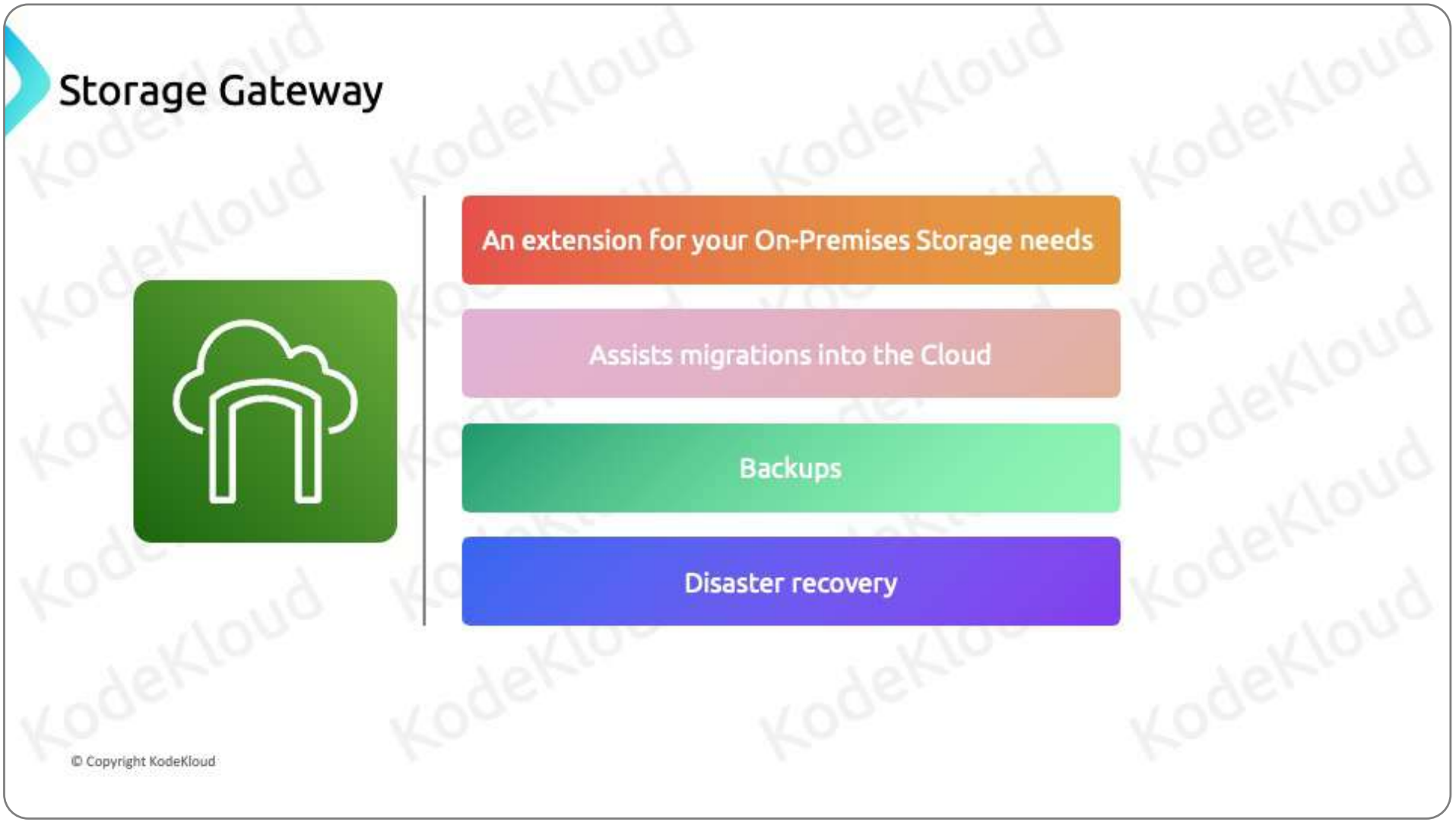
05 A launch template is used to configure the specifications of the recovery servers (size, region/subnet, security group)

Service Section

The Power of Storage Manipulation

Storage Gateway

AWS Storage Gateway is a hybrid cloud storage service provided by Amazon Web Services (AWS). It acts as a bridge between your on-premises environment and cloud-based storage, allowing you to seamlessly integrate on-premises applications with cloud storage resources.



AWS Storage Gateway is a hybrid cloud storage service provided by Amazon Web Services (AWS). It acts as a bridge between your on-premises environment and cloud-based storage, allowing you to seamlessly integrate on-premises applications with cloud storage resources.

So storage gateway scan be used for the following purposes:

- Make AWS storage services act as an extension for your onprem storage needs – so if you run low on storage in ur onprem

site, you can utilize AWS to increase ur storage needs
- Assist migrations into the cloud
- Backups
- Disaster Recovery

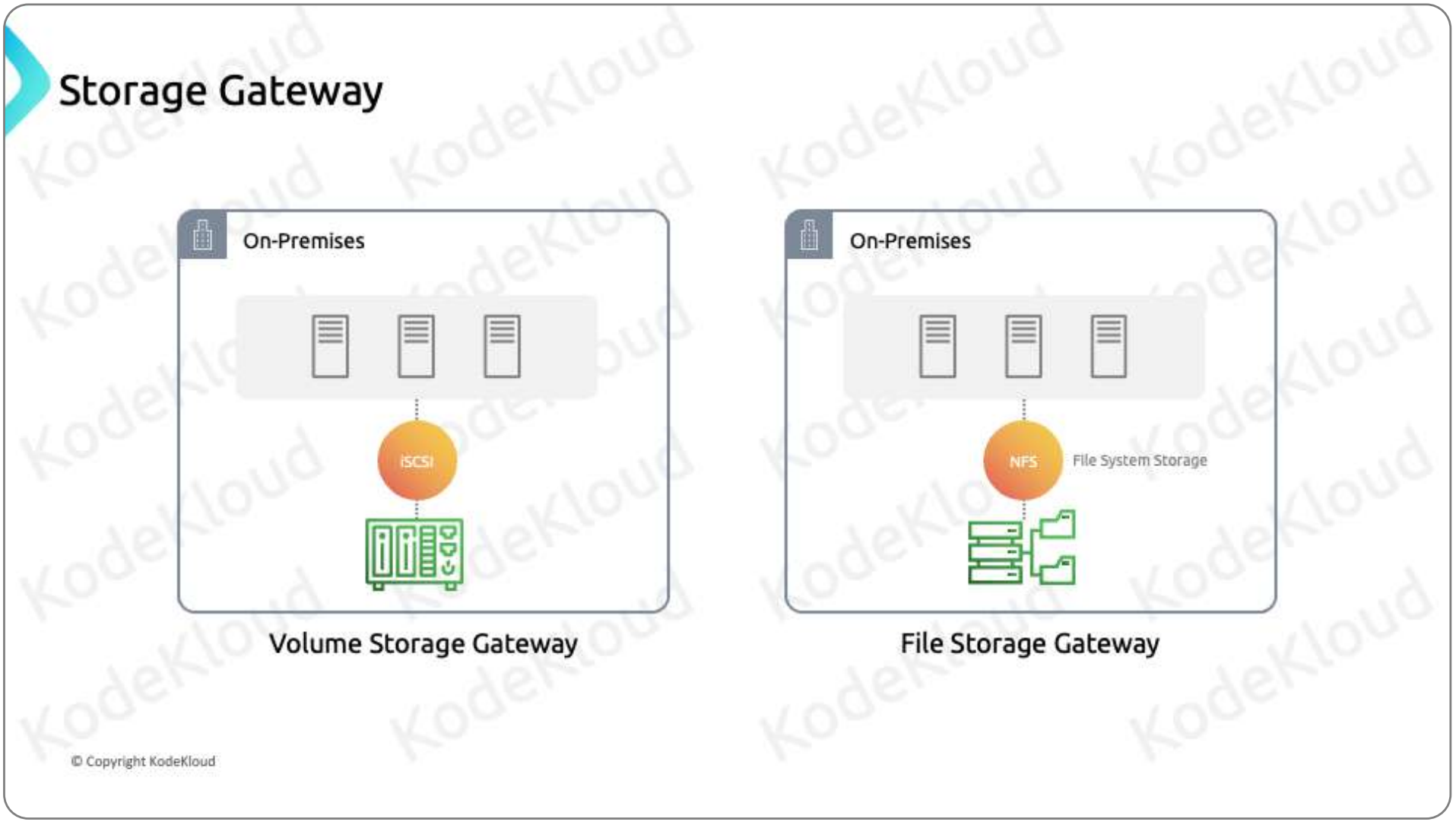
There’s a variety of different options and configurations with storage gateway, but there’s usually 2 common goals that you want to achieve with storage gateway

The storage gateway is either a virtual machine or a physical unit that you can deploy in your on-prem datacenter. It is the device that acts as a bridge between ur on-prem environment and aWS.

Storage gateways come in 3 different flavors

- Volume
- File
- Tap

The type of storage gateway you use depends on the type of storage you are using in your onprem



Let’s say you have an onprem environment, and you have a bunch of servers whoyou connect to a Network Attached Storage (NAS) using iSCSI. Since this is using block storage then we’ll choose Volume storage gateway. If the servers are using filesystem storage and using NFS to communicate then we’ll select File Storage gateway and if we are creating tape backups, then we’ll select tape

A volume Gateway operates in 2 different modes:

Cached mode
Stored Mode

These 2 different modes offer different features and have their own advantages

In Stored mode, the storage gateway presents volumes over iSCSI to the onprem servers. These volumes will look exactly the same as the volumes presented by the NAS or SAN. Servers can then create filesystems ontop of these volumes. In Stored mode, these volumes on the storage gateway are stored on-prem. So the storage gateway has local storage, and all the volumes will exist on physical disks attached the storage gateway

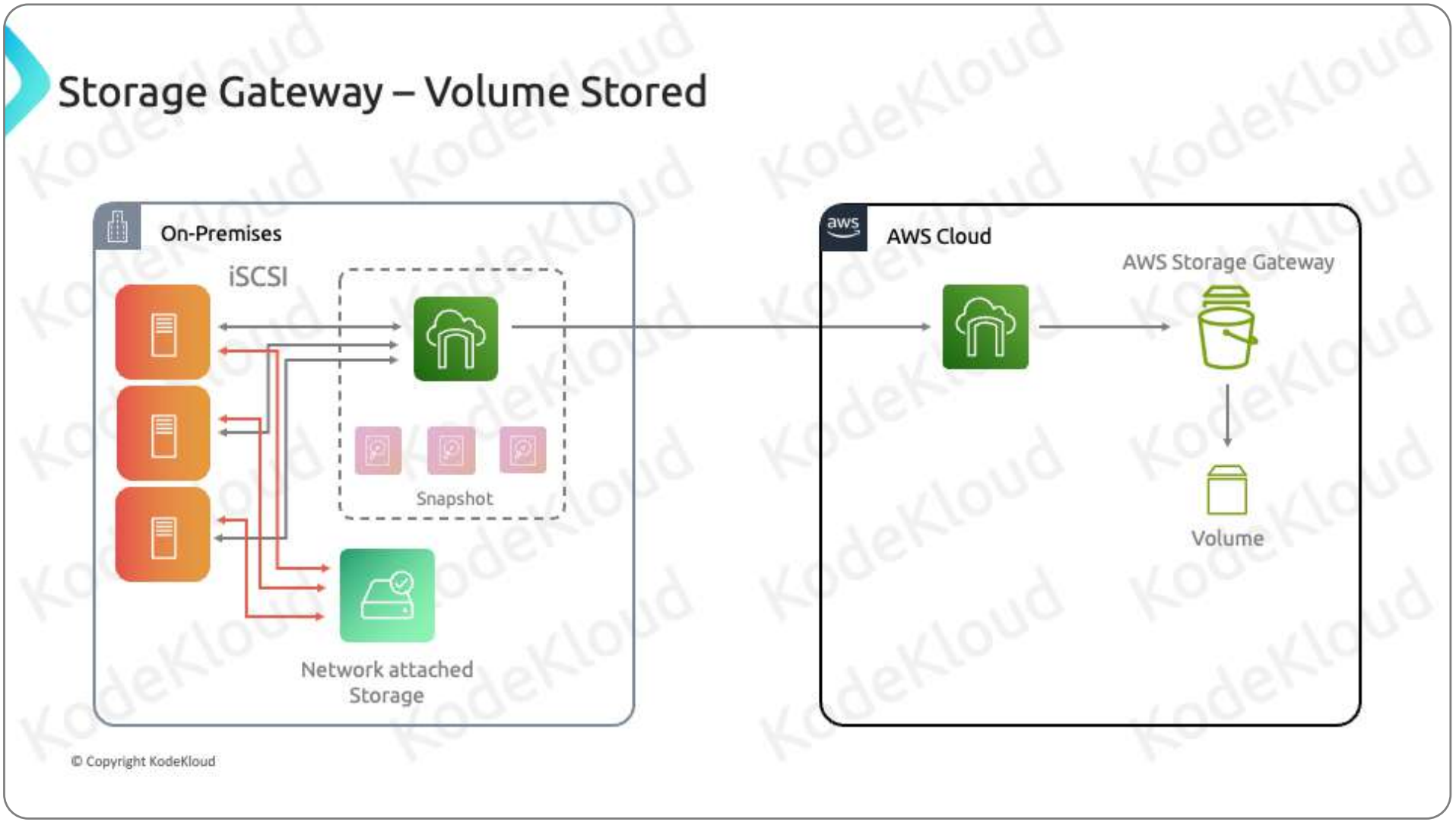
All data is stored locally

The storage gateway also has an upload buffer, so any data stored in the local storage gets copied to the upload buffer, which then gets uploaded to AWS via a storage Gateway endpoint. This is a public endpoint so this can go over internet or through a direct connect.

The data gets uploaded to AWS S3 and are stored as EBS Snapshots

Volume stored mode is great for doing backups and doing disaster recovery as the ebs snapshots can be quickly restored.
With Volume stored mode, its important to understand that this does not extend your data capacity needs into the cloud. All data is stored on local storage on-prem, so if you need ot add more storage space you have to add more local storage. Aws is only used for backups

[Design team] I added a picture just so you could have a general idea of what I'm trying to go for. We have onprem on left with storage gateway, and then the cloud on the right



So let’s see how storage gateways work.

So let’s say we have our on-prem data-center. And in this data center we have a bunch of servers. These servers connect to a Network attached storage which presents raw block storage to the servers over the networ using a protocol like iSCSI. They can then use it like any other storage device and create a filesystem on top of them.

Businesses would ideally like to have some system for backups for all their data as well as disaster recovery. But implementing these can become both architecturally challenging as well as quite expensive.

This is where aws storage gateway comes into play. Now the AWS storage gateway is deployed as an appliance at the on-prem datacenter.

Now in stored mode the virtual appliance presents storage volumes over iSCSI to the physical servers. In the same exact was the Network attached storage did. Servers can create filesystem ontop of these volumes.

In stored mode these volumes consume capacity on prom. So the storage gateway has local storage in the on-prem datacenter, and this is where all the data is stored.

So this is an important thing to remember for the exam. IN stored mode, data is stored locally on-prem

After data is written to disk, the storage gateway will then copy the data to AWS and this is via the storage gateway endpoint(public endpoint, so it can go over internet connection)

The data is then copied to s3 as ebs snapshots.

Storage Gateway – Volume Stored

Data is stored locally on-prem

Data is replicated asynchronously to AWS S3

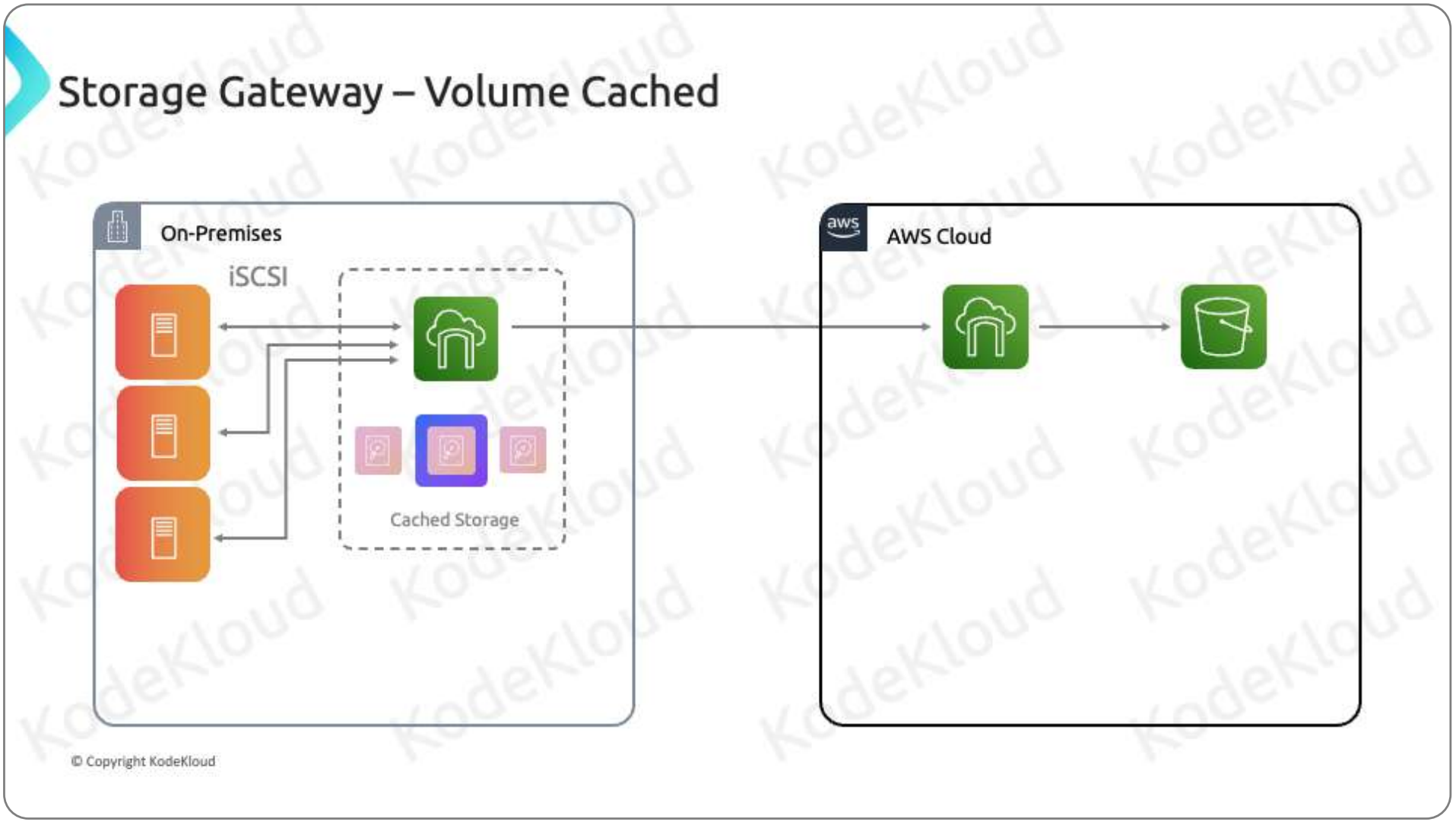
Provides convenient backup of data

Assists with disaster recovery

Create EBS volumes from snapshots

Doesn't increase datacenter storage capacity

All data is still stored on-prem, only backups are stored in AWS



Now with volume cached mode, once again we still have our storage gateway which is deployed at the physical datacenter. And it still prevents block storage to the servers over iscsi. But the main difference is that the storage gateway doesn't store data locally instead. All data is actually stored in AWS

More specifically the data is stored in s3. SO this is the most important distinction you need to understand for the exam. In Volume cached mode data is stored on AWS so AWS actually helps you extend the overall size of your storage solution and

infinitely scale up. Whereas with stored mode, the data still needs to be stored on disk at the onprem site.

In cached mode the only thing stored locally is frequently accessed data which is kept in the cache storage

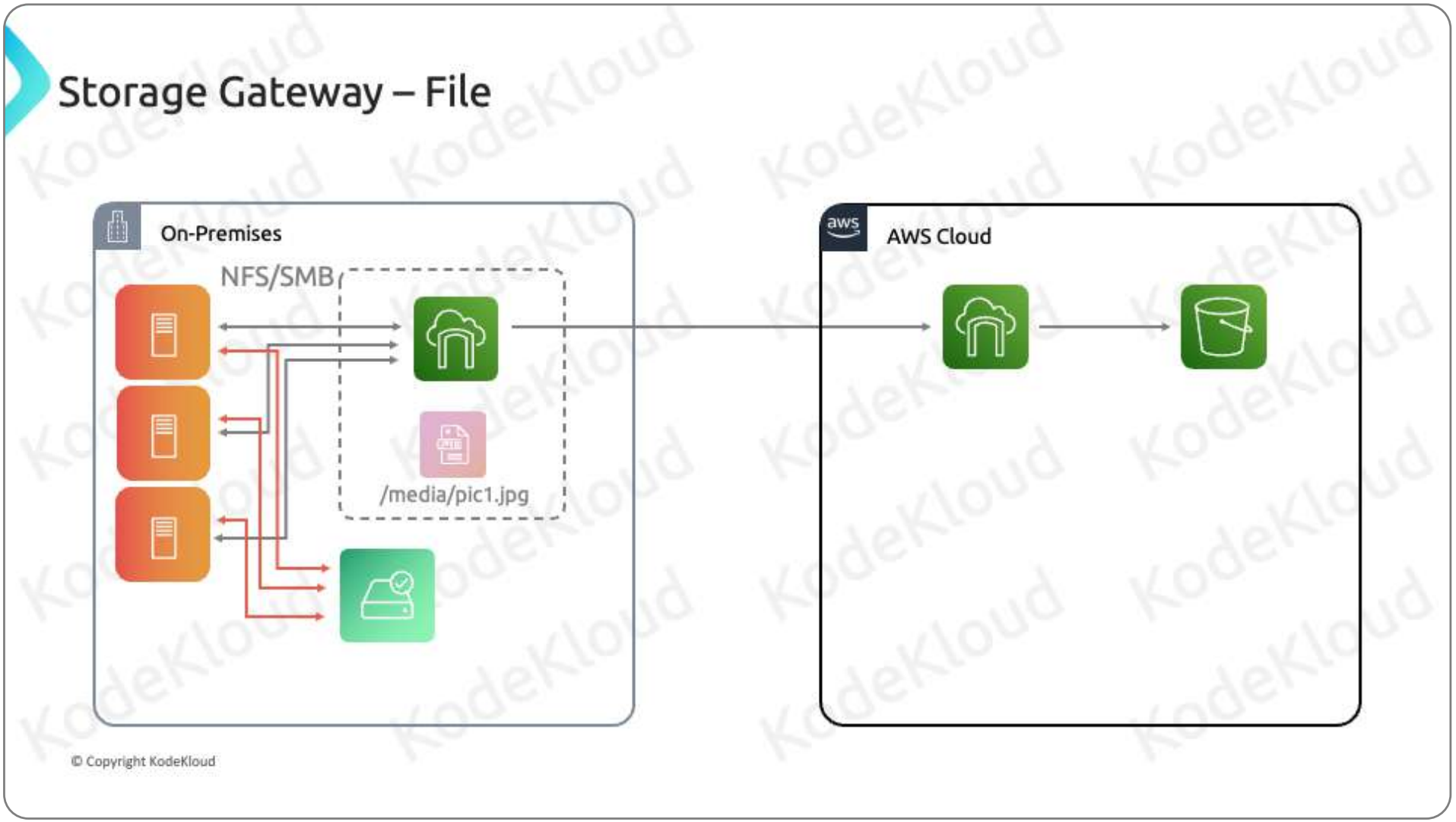
This is an example of datacenter extension. S

Storage Gateway – Volume Cached

Data is stored on S3

Only data on-prem is cached data, for frequently accessed data

Cached mode acts as a datacenter extension - increases customers storage capacity



Filegateway follows the same architecture as volume gateway. But its for instances where you have filesystem based storage onPrem instead of block storage. So instead of using iscsi your onprem servers use NFS.

So it works the same way. The storage gateway resides onprem and presents a filesystem storage to onprem servers. Servers can then connect ot it using nfs/SMB like any other NFS server and read/write to it. Files will be stored on an s3

bucket. And the file structure will be represented by the Keys on the fileand so you can build out a filestructure on Se3

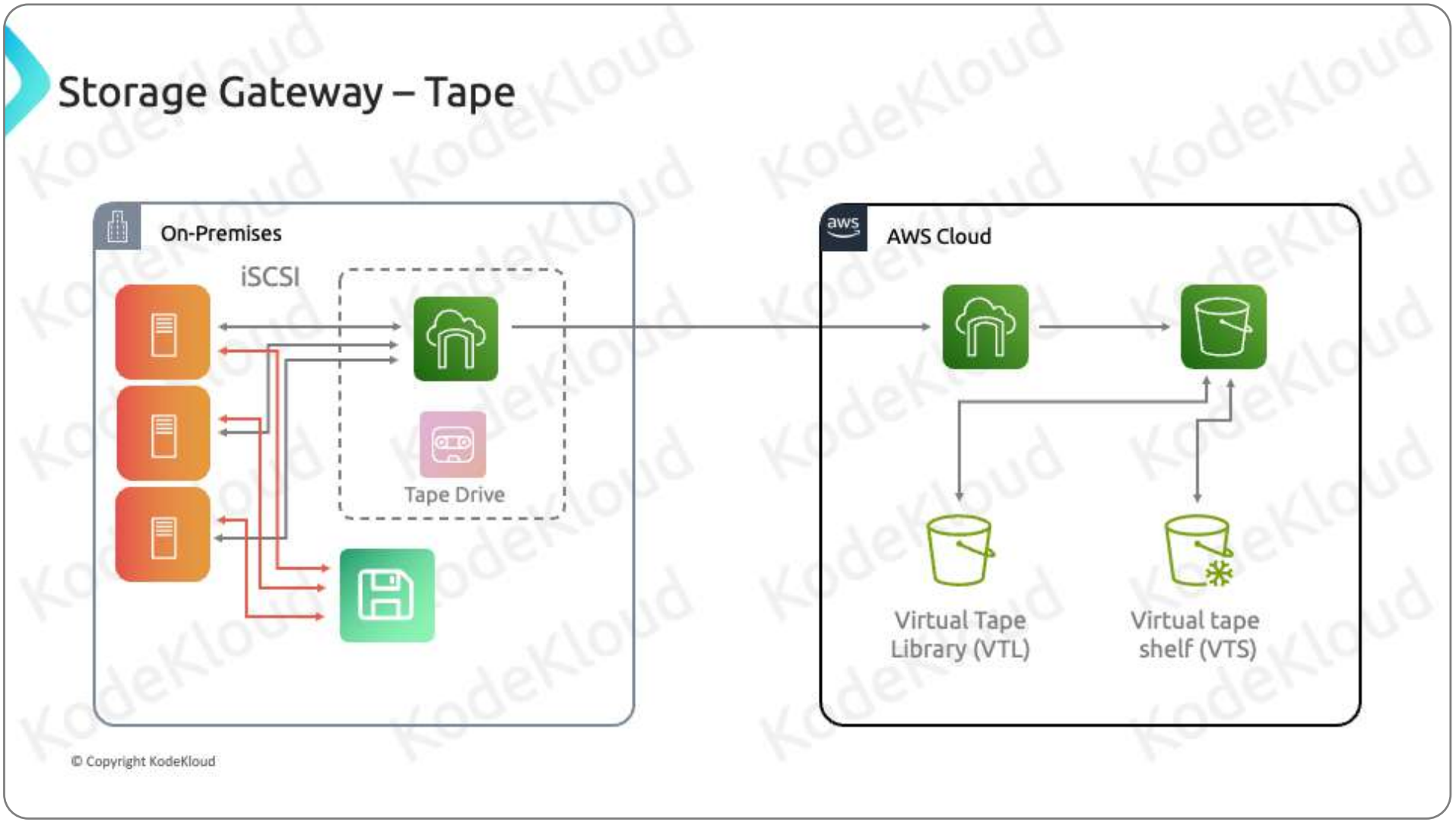
Storage Gateway – File

01

Stored in S3

02

Only data on-prem is cached data, for frequently accessed data



When it comes to enterprise backups, one form of backups is using Tapes.

With traditional tape backup, the backup servers connect to the tape library using iSCSI, to secure backups of data in a TAPE.

And this obviously has a certain level of maintenance/complexity and cost associated with it. You have to pay to store all the tapes and move it between locations. Amazon introduced Storage gateway Tape (VTL) to simplify this process.

The backup servers will connect to the storage gateway which will present itself as a tape library

Storage Gateway – Tape

01

Emulates a tape library

02

Data is stored in AWS

- Virtual Tape Library (VTL) – S3
- Tape Shelf (VTS) – Glacier

03

Virtual Tape 100GB – 5TB

As I mentioned Storage Gateway –file and tape follow the same model the only difference is instead of present raw block storage over iSCSI to the onprem servers:
For file they offer file storage over NFS and for tape they offer tape backups

Summary

01 The shared responsibility model delineates the customer's responsibilities and AWS's responsibilities

02 Unmanaged services need to be secured by users

03 Managed services offload some of the security responsibility onto AWS

KodeKloud