AWS Training

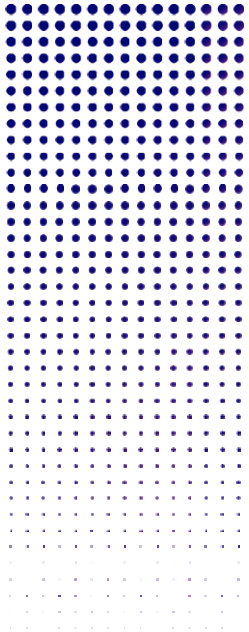
amazon web services | Partner Network

PREMIER CONSULTING PARTNER

Associate Architect and Development on AWS

Version: 1.0-201906/CCG/FSJ

Contact Point: tuanba5@fsoft.com.vn

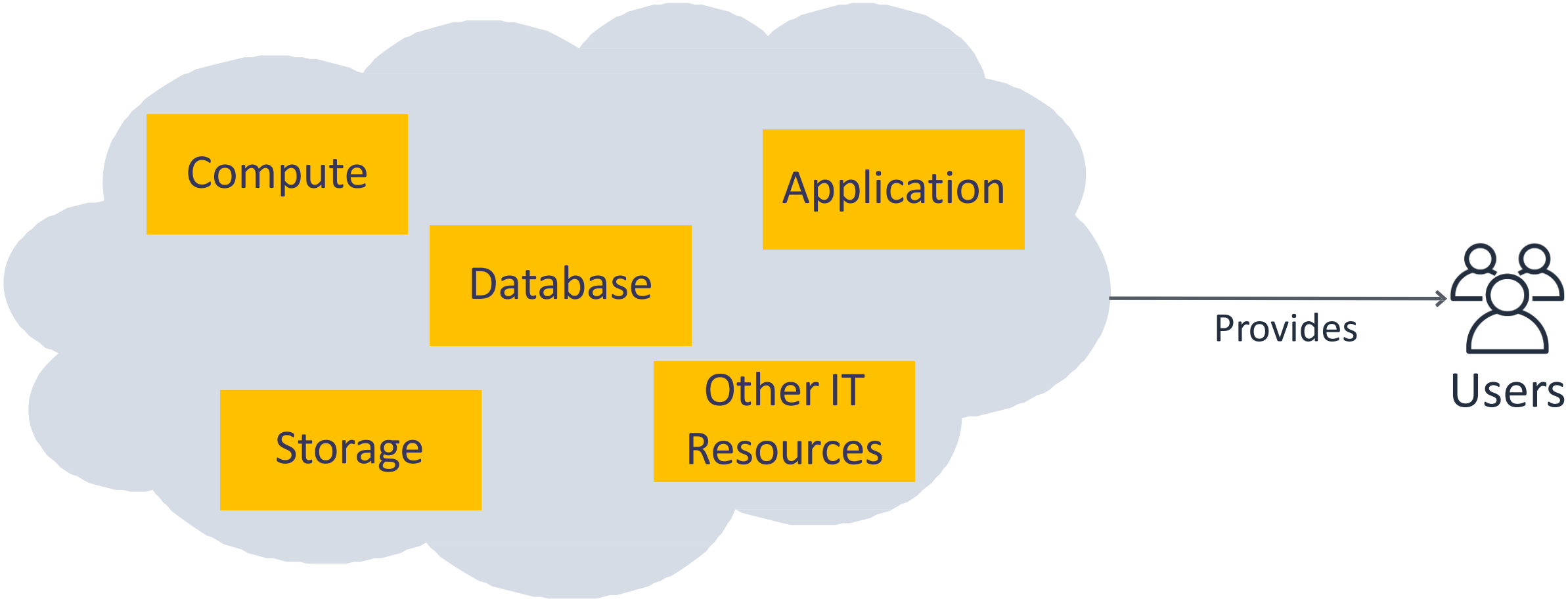


Day
1

AWS Cloud Platform
&
Cloud Computing

Cloud

- What is Cloud?



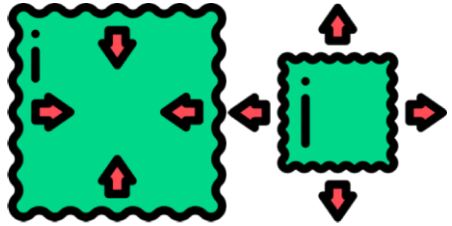
Cloud vs On-premise

- Benefit of Cloud

Cost Effective

Agility

Mobility



Elasticity

Backup & Restore

Cloud

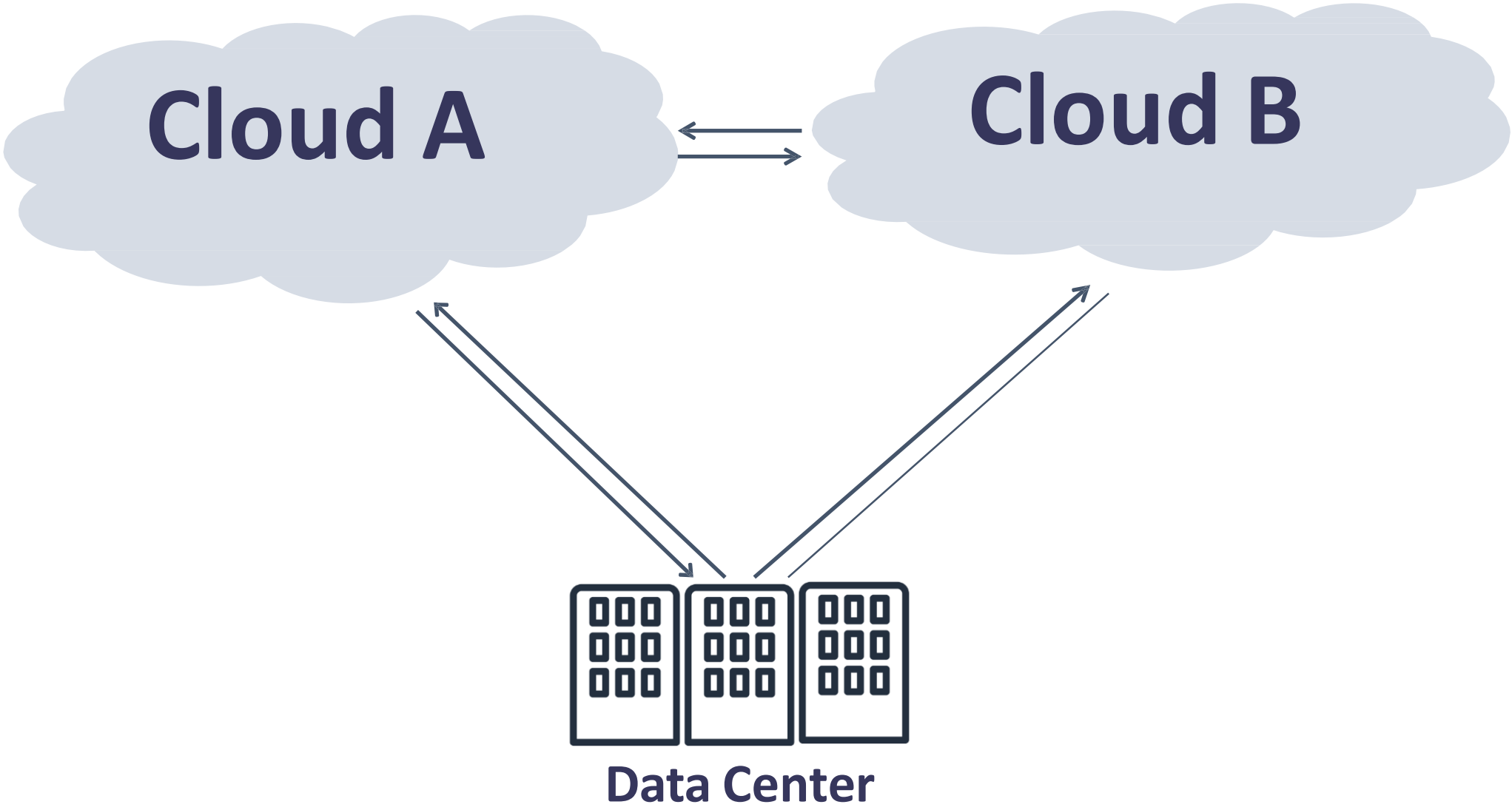
Cloud Service

- No Up-front cost
- Pay-as-you-go
- Workload: Elasticity
- Locale: Mobility
- Flexible Security
- DR (Disaster Recovery) ability
- Well Backup & Restore
- More Options

Traditional Computing

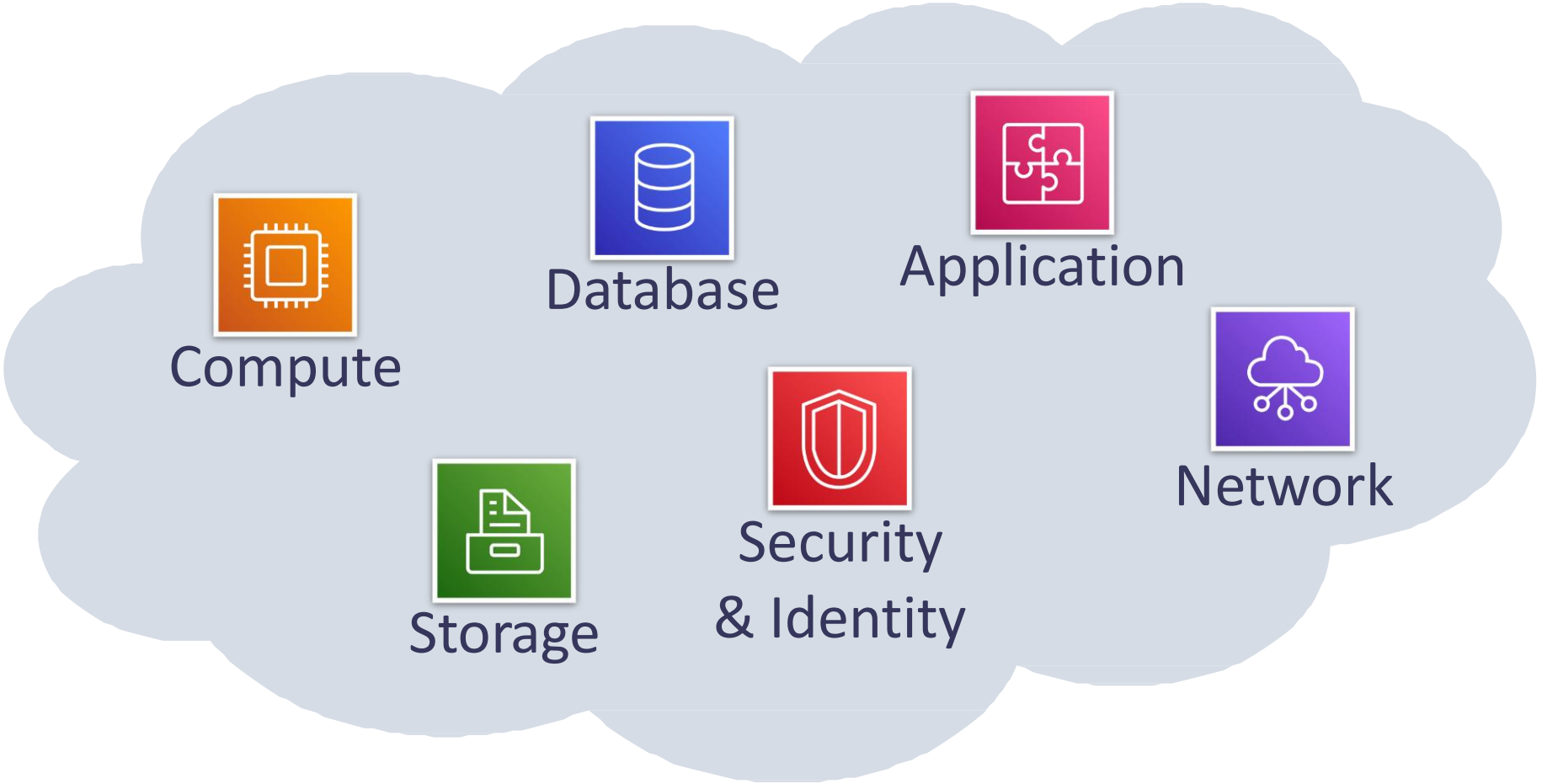
- To much on Up-front cost
- Pay all the time even idle time
- Workload: Fixed
- Locale: Fixed
- Hard Security
- SOF (Single Point Of Failure)
- Limited Backup & Restore
- Fixed Options

Hybrid Cloud



AWS Cloud Platform

- **AWS** stands for **A**mazon **W**eb **S**ervices

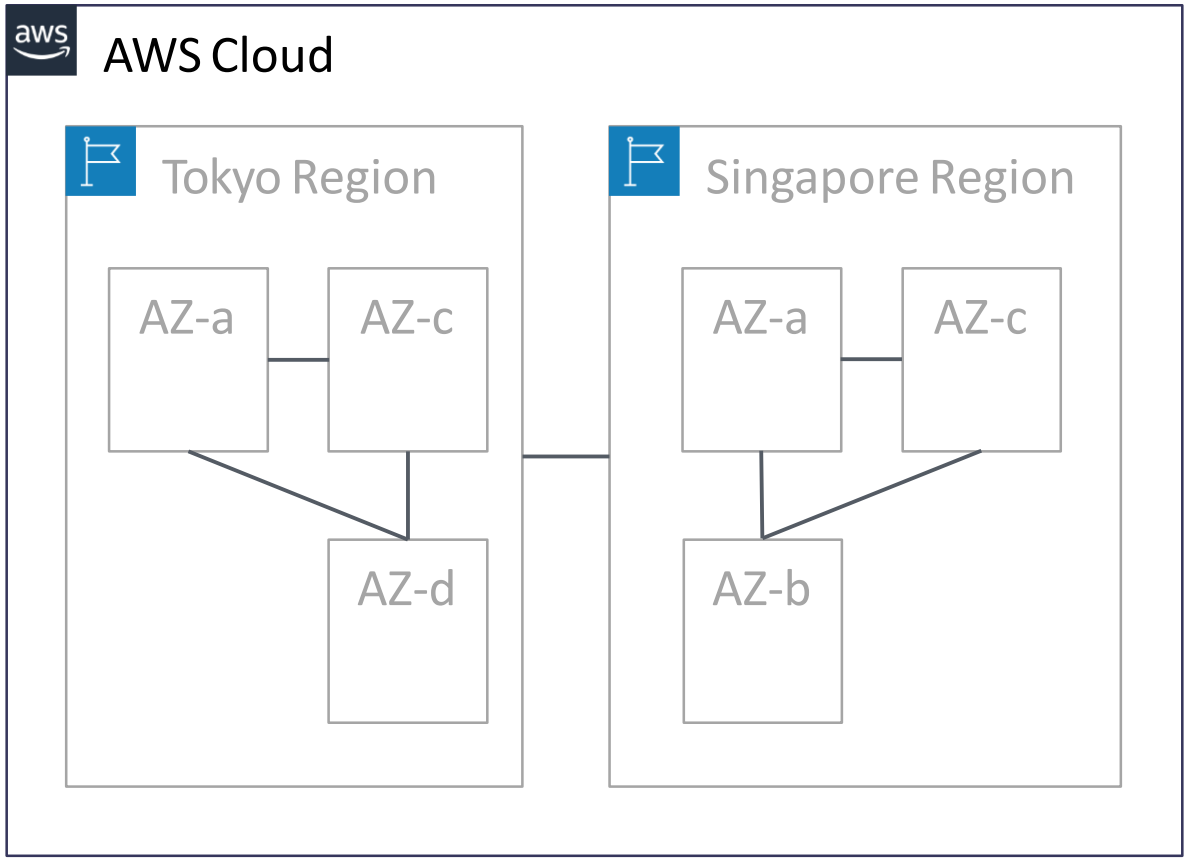


AWS Global Infrastructure Map

- 21 Regions
- 66 AZs (*)
- Edge Location
- Coming Soon
  - 12 AZs
  - 4 Regions

* AZ: Availability Zone

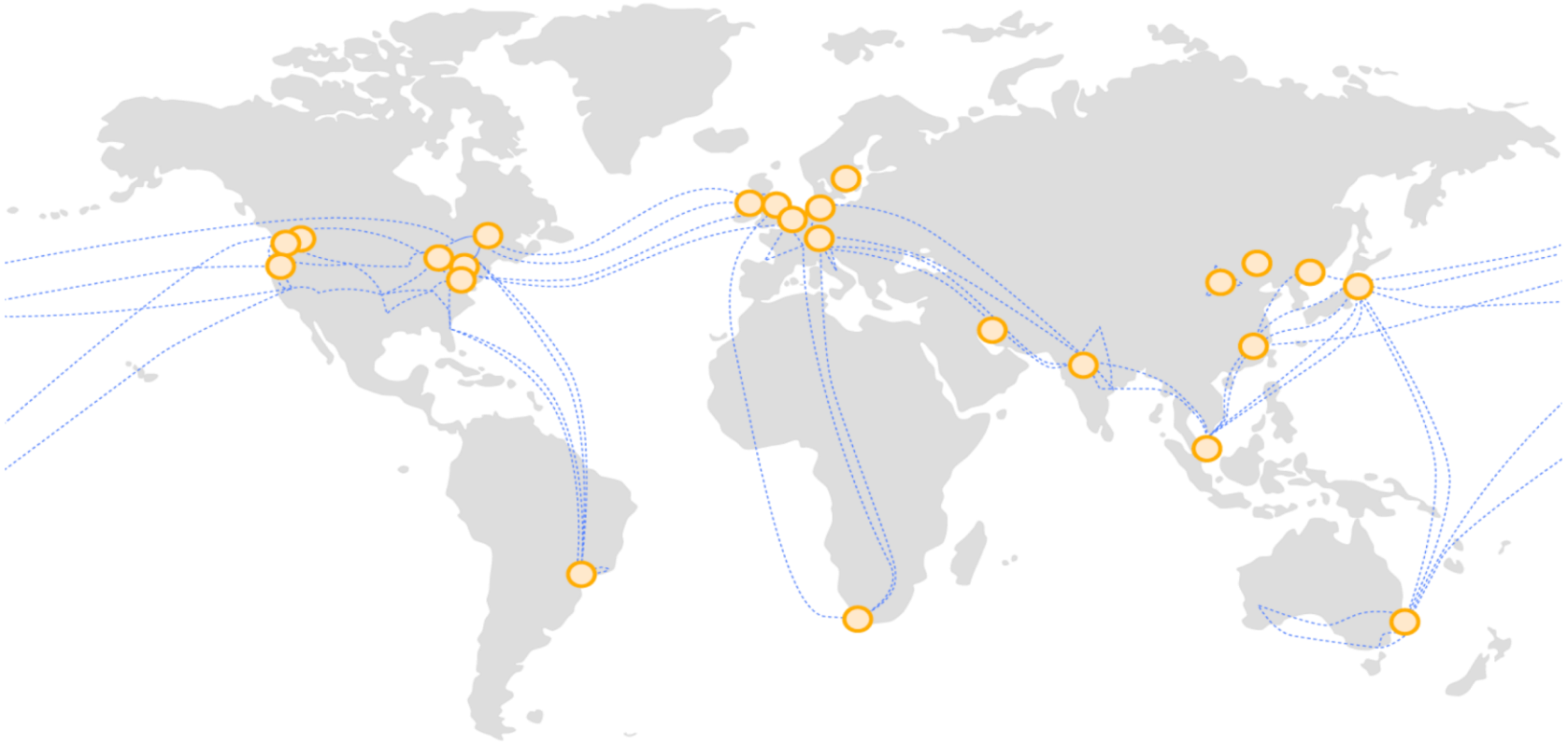
AWS Region and Availability Zone



Code	Name
us-east-1	**US East (N. Virginia) (default)**
us-east-2	US East (Ohio)
us-west-1	US West (N. California)
us-west-2	US West (Oregon)
ca-central-1	Canada (Central)
eu-central-1	EU (Frankfurt)
eu-west-1	EU (Ireland)
eu-west-2	EU (London)
eu-west-3	EU (Paris)
eu-north-1	EU (Stockholm)
ap-east-1	Asia Pacific (Hong Kong)
ap-northeast-1	**Asia Pacific (Tokyo)**
ap-northeast-2	Asia Pacific (Seoul)
ap-northeast-3	***Asia Pacific (Osaka-Local)***
ap-southeast-1	**Asia Pacific (Singapore)**
ap-southeast-2	Asia Pacific (Sydney)
ap-south-1	Asia Pacific (Mumbai)
sa-east-1	South America (São Paulo)

AWS Global Infra & Global Network

- High Availability & Disaster Recovery
- Flexible
- Security
- Performance
- Lower Cost



AWS 1st Practice

- Sign-up/Sign-in AWS Console

<For FJPer, use provided AWS Account>

- First look with AWS Console: common features

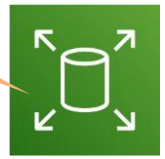
- Enable MFA, create IAM user, create budget

<Just follow the instruction of mentors>

AWS Storage Services

<Block File Storage>

Day 1



Elastic Block Store (EBS)

Elastic File System (EFS)

<Object File Storage>

Day 3

Simple Storage Service (S3)

Glacier

<Peta/Exabyte-scale Data Migration Storage>

Snowmobile

Snowball

<Hybrid Storage>

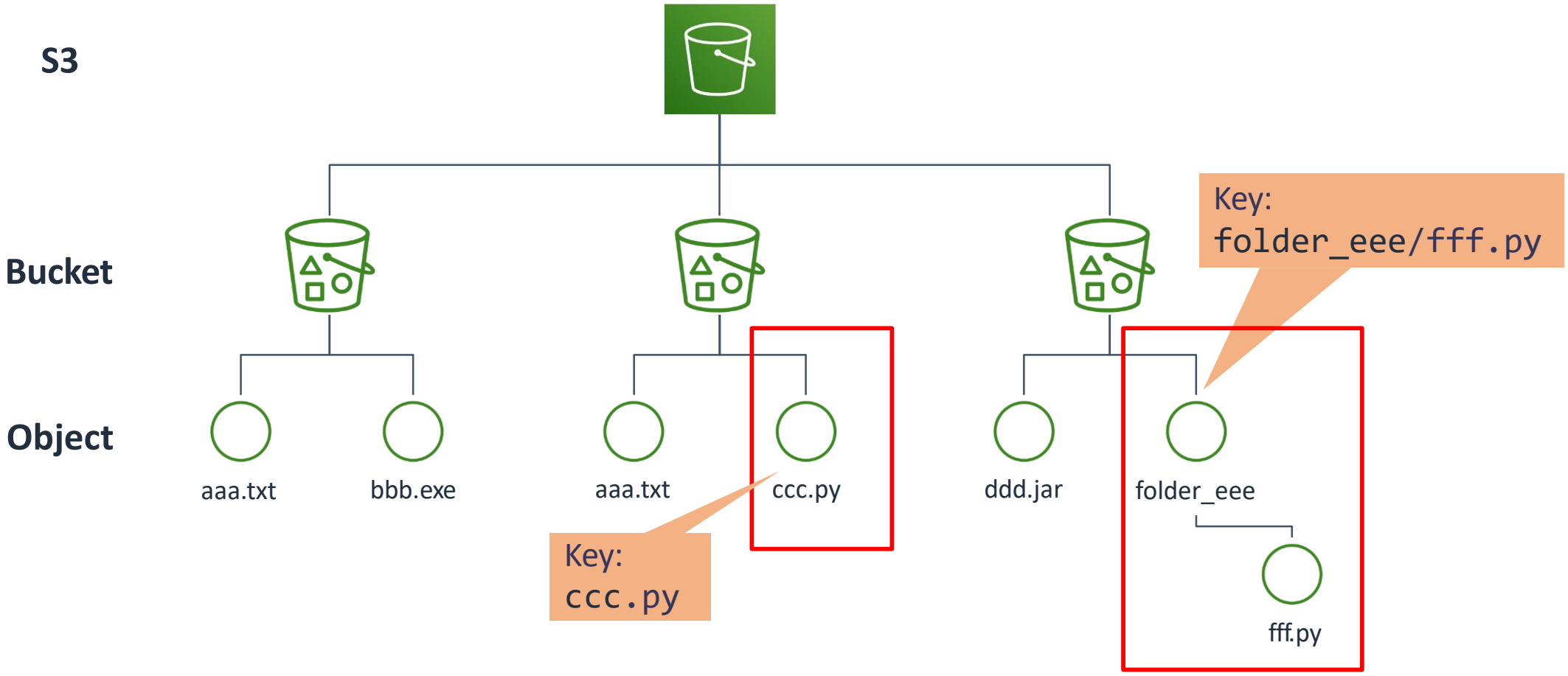
Storage Gateway

AWS S3 – Simple Storage Service

AWS S3 – Simple Storage Service

- Global Unique Identifier

`{bucket_name}/{object_name(key)}`

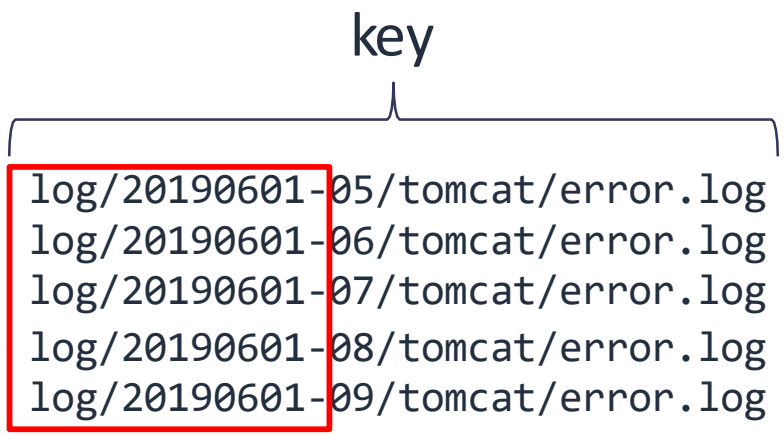


AWS S3 – Practice Time

- Try to create S3 bucket
  - Name: aws01-2019 => ???
  - Region: Any
- Create S3 bucket
  - Name: aws01-2019-FS_Acc
  - Region: Tokyo

AWS S3 – Simple Storage Service

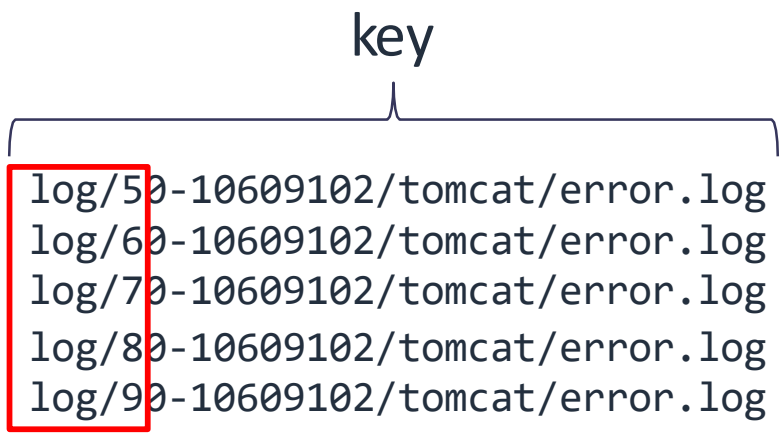
- S3 partitions based upon **key prefix**
- Object Name Best practice
  - Throughput Optimization
  - Increase number of partition
  - Avoid increment key
  - Revert increment key



- **1 Partition:**
- *Log/20190601-*

AWS S3 – Simple Storage Service

- S3 partitions based upon **key prefix**
- Object Name Best practice
  - Throughput Optimization
  - Increase number of partition
  - Avoid increment key
  - Revert increment key

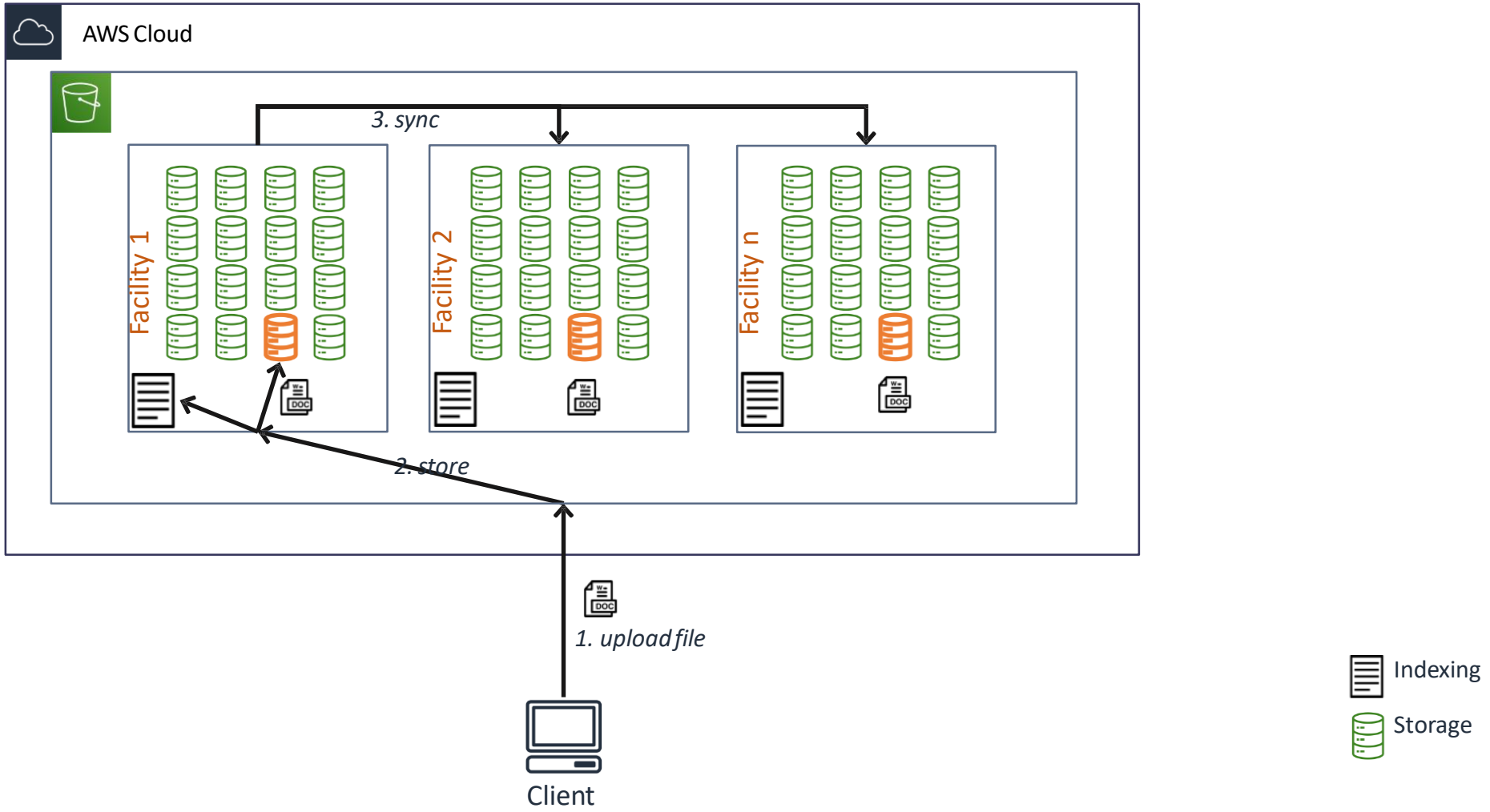


5 Partitions:

log/5
log/6
log/7
log/8
log/9

AWS S3 – Simple Storage Service

- **Write once, Read many**

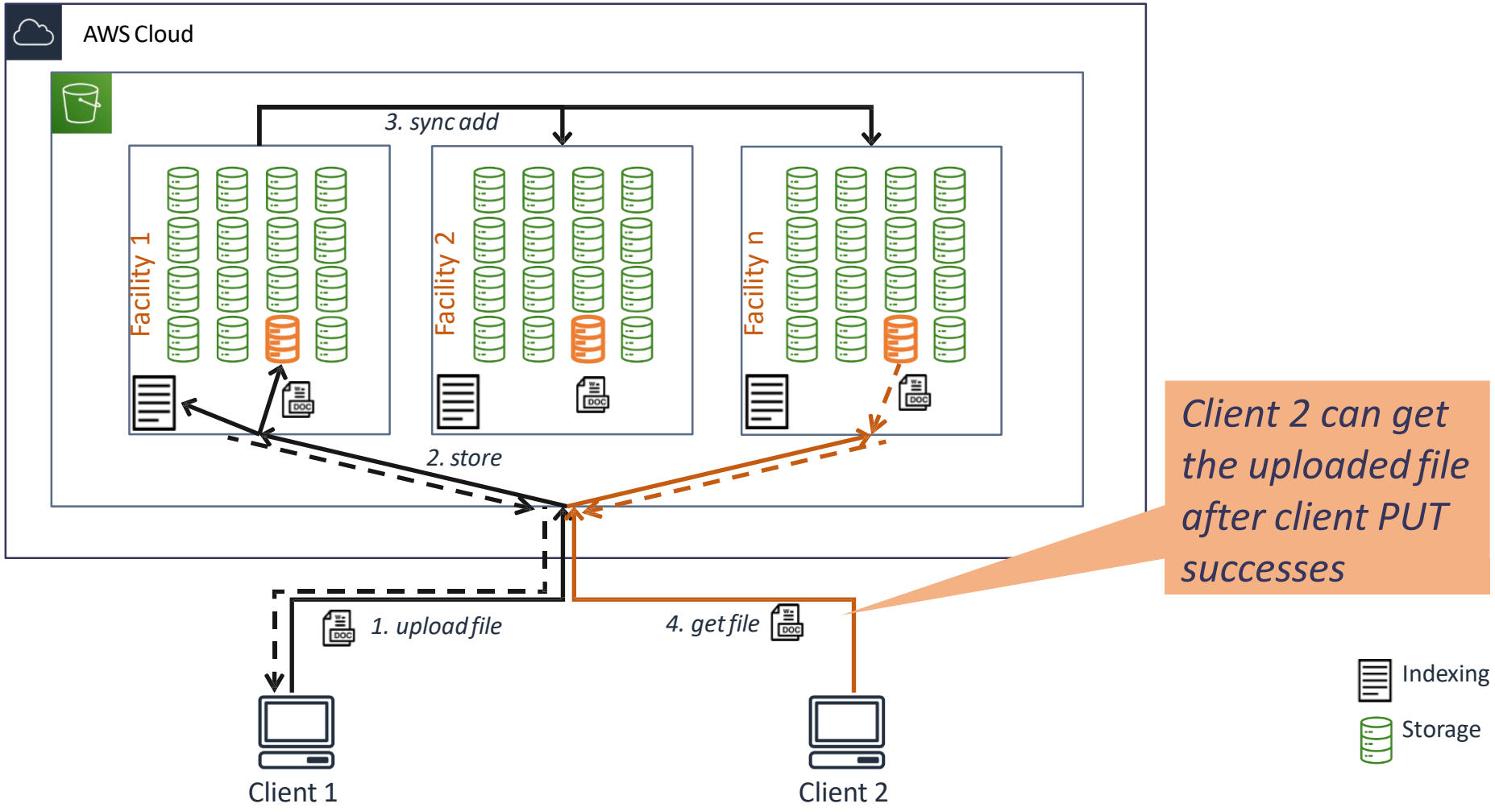


AWS S3 – Simple Storage Service

- Eventually Consistent
  - New Object (PUT)
    - Read-after-write consistency
  - Update
    - Write-then-read
    - Overwrite-then-read
  - Delete
    - Delete-then-read

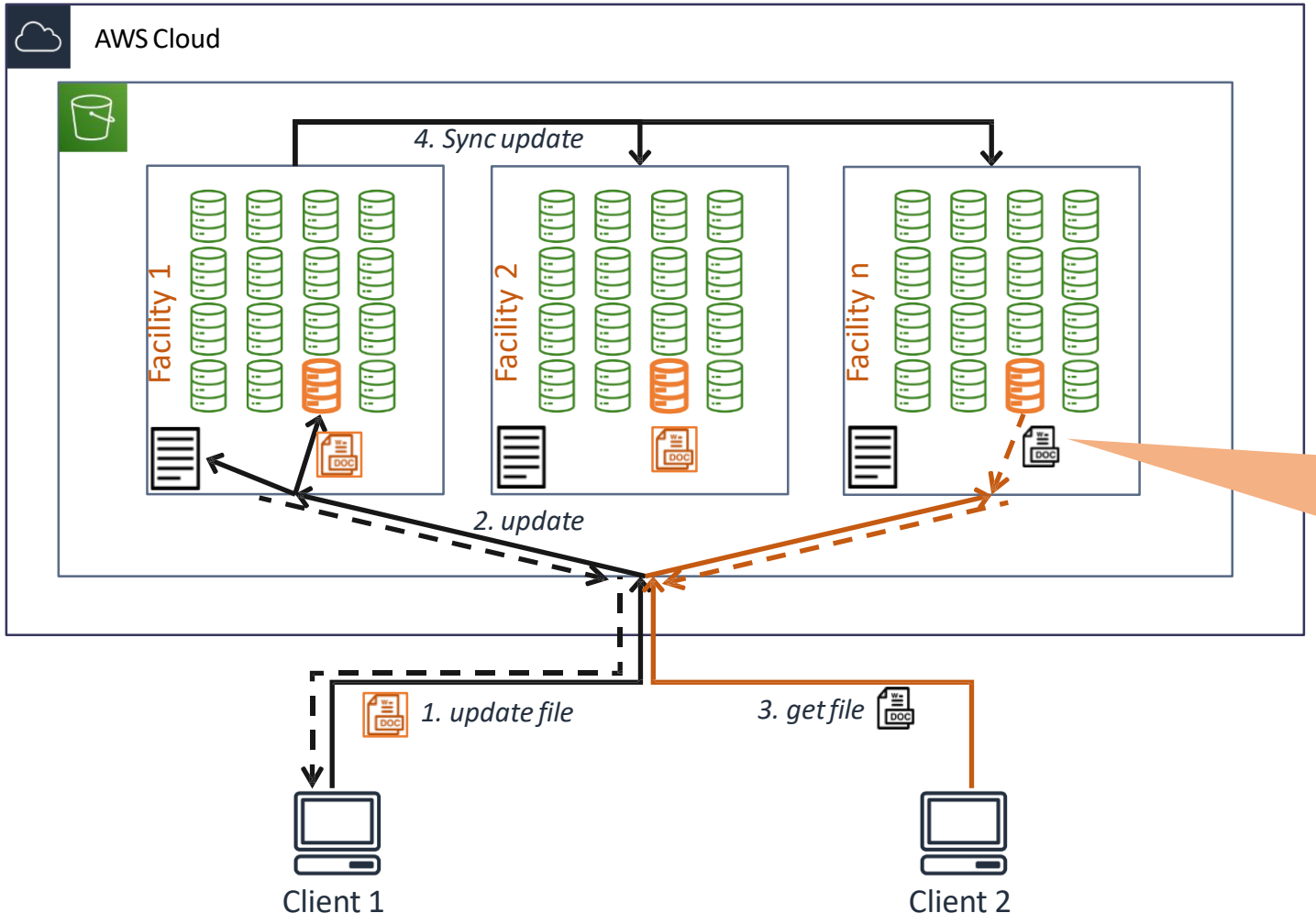
AWS S3 – Simple Storage Service

- Read-after-write consistency



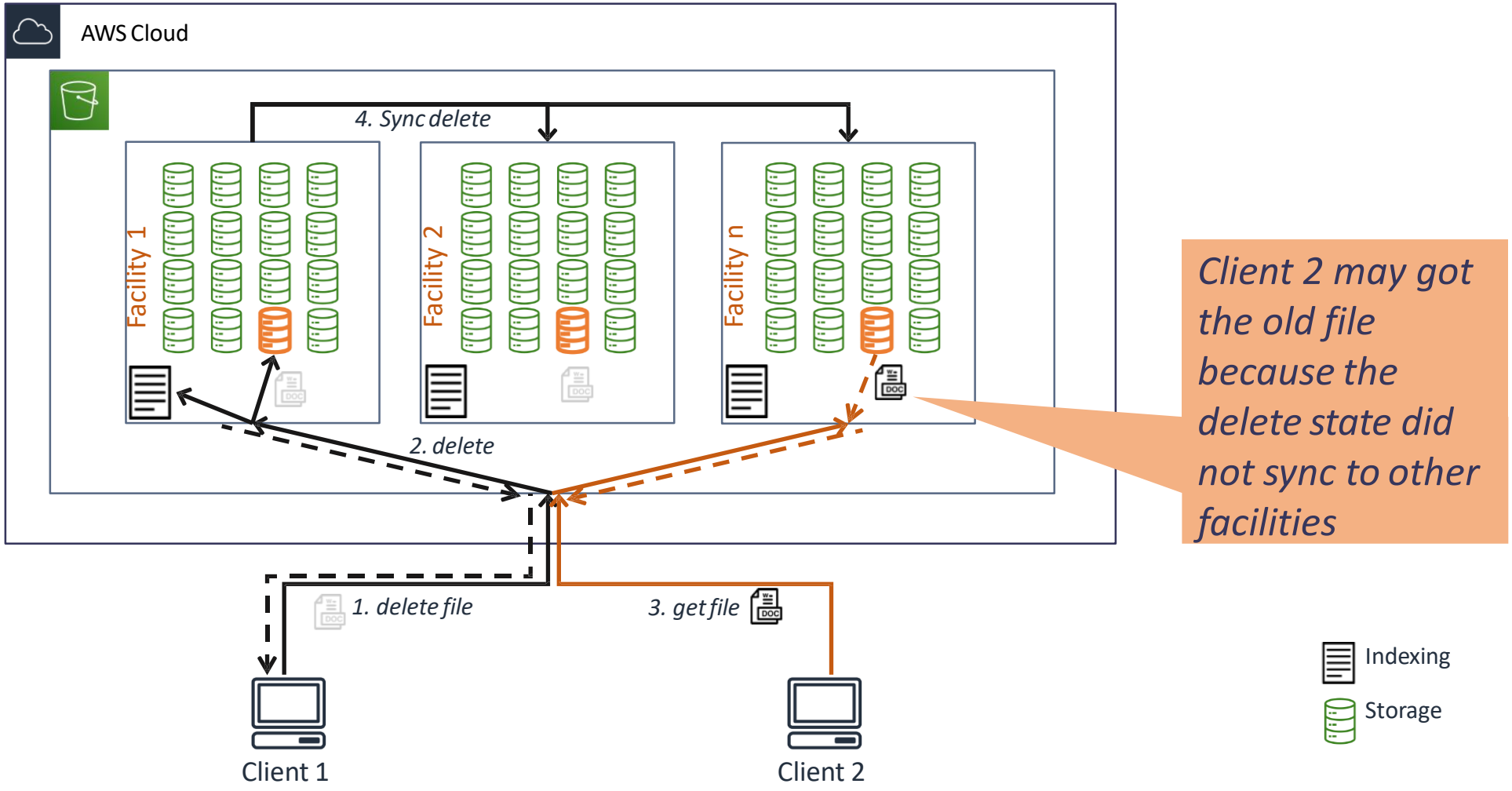
AWS S3 – Simple Storage Service

- Write-then-read & Overwrite-then-read

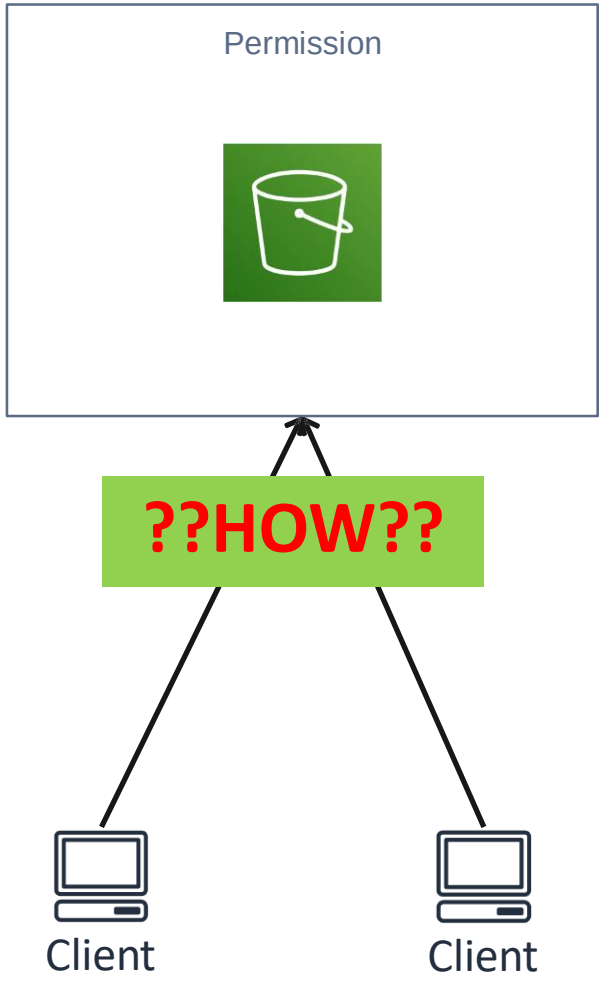


AWS S3 – Simple Storage Service

- Delete-then-read

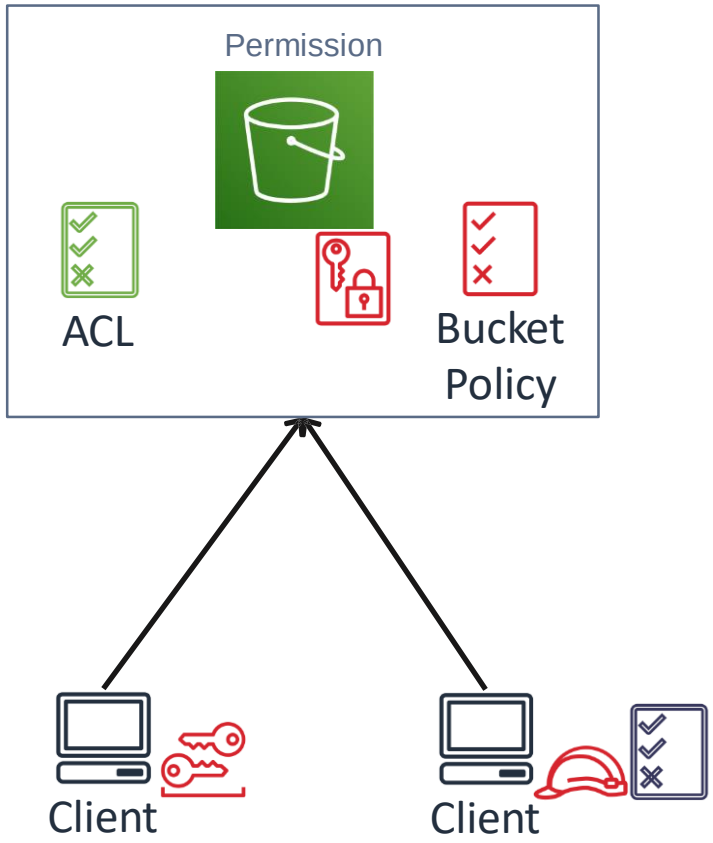


AWS S3 – Security



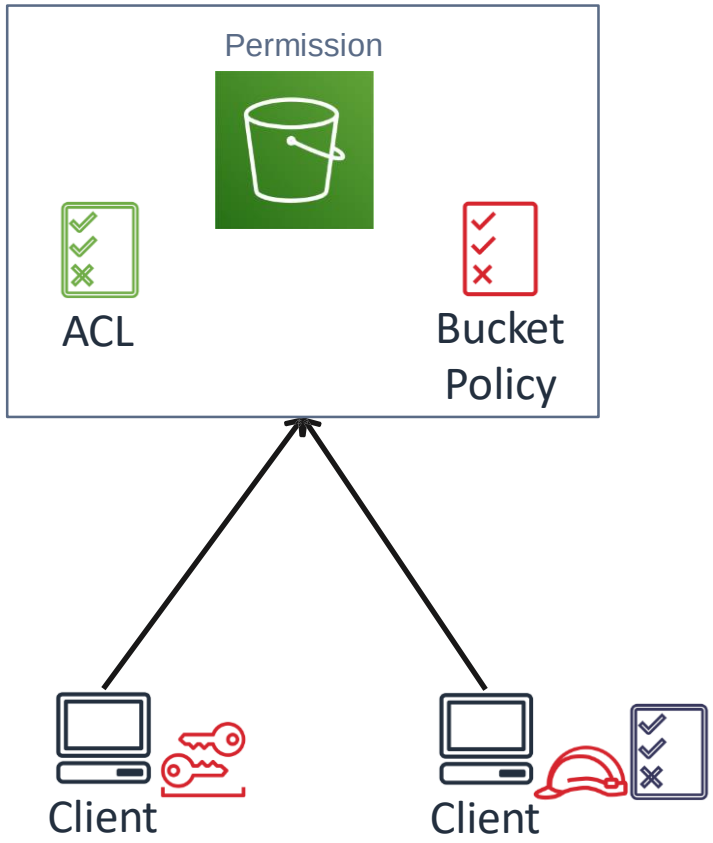
AWS S3 – Security

1. Policy
    1. **Bucket Policy**
    2. User Policy (IAM)
1. ACL – Access Control List
2. Encryption



AWS S3 – Access Management

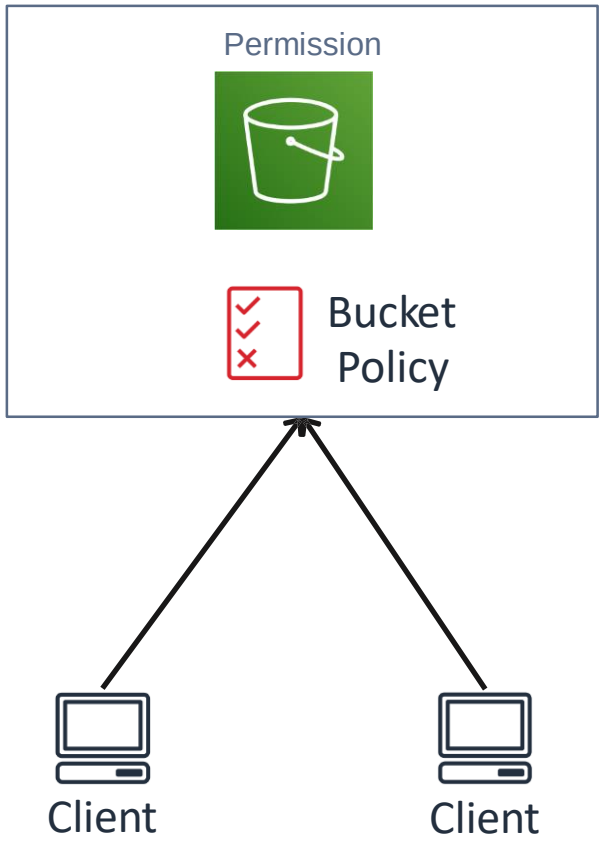
1. Policy
    1. **Bucket Policy**
    2. User Policy (IAM)
2. ACL – Access Control List



AWS S3 – Bucket Policy

- Syntax: the same as IAM Policy
- Define allow/deny actions
- Target to bucket and bucket's object
- Apply for current bucket
- AWS S3 Bucker Policy Samples

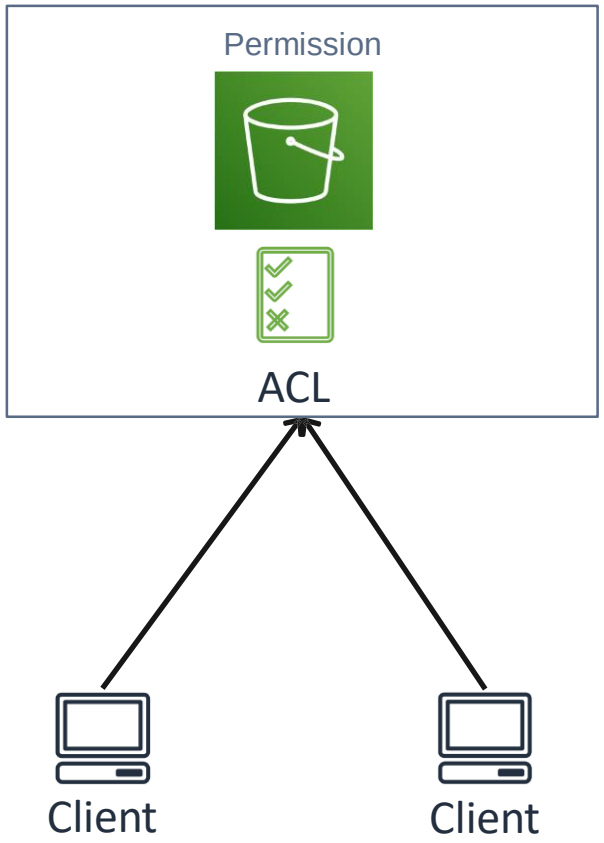
```
{
    "Version": "2012-10-17",
    "Id": "Policy1560327064924",
    "Statement": [
        {
            "Sid": "Stmt1560327027597",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::1111111111111:role/demo-ec2-role"
            },
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::aws01-2019",
                "arn:aws:s3:::aws01-2019/*"
            ]
        }
    ]
}
```



AWS S3 – Access Control List

- Syntax: XML
- Manages access to bucket and object
- Each object has it own default ACL
- Grantee: account, Amazon S3 Predefined Groups
- Permission: READ, WRITE, FULL_CONTROLL,…

```xml
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Owner>
    <ID>*** Owner-Canonical-User-ID ***</ID>
    <DisplayName>owner-display-name</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:type="Canonical User">
        <ID>*** Owner-Canonical-User-ID ***</ID>
        <DisplayName>display-name</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```



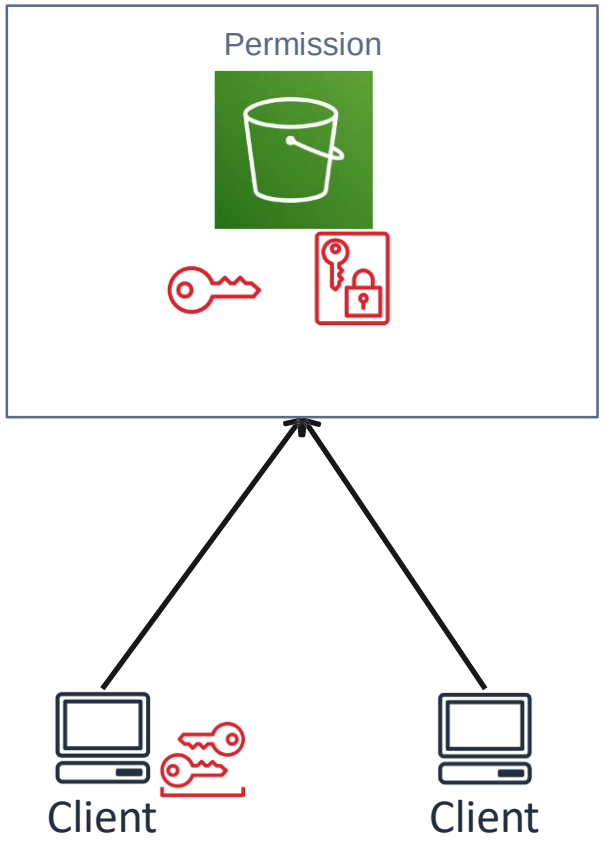
AWS S3 Security – Practice Time

- Create a IAM role for EC2
- Setting Bucket Policy for S3
  - Allow IAM Role can get, put object
- Setting ACL to public S3 bucket

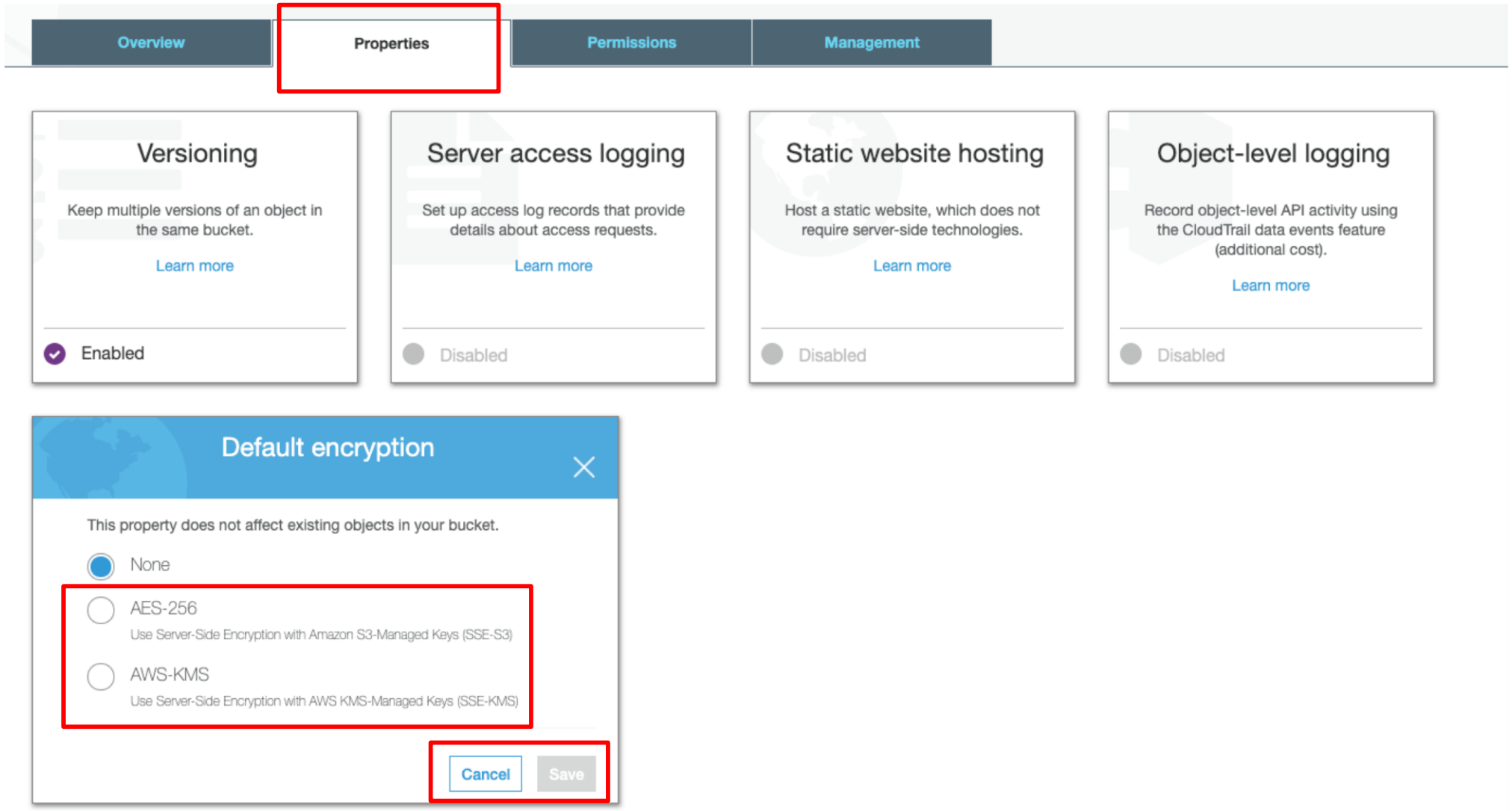
AWS S3 – Encryption

- SSE – Server-Side Encryption
  - Amazon S3-managed keys (SSE-S3 AES-256)
  - AWS KMS-managed keys (SSE-KMS)

- CSE – Client Side Encryption
  - User does encrypt data before put to S3

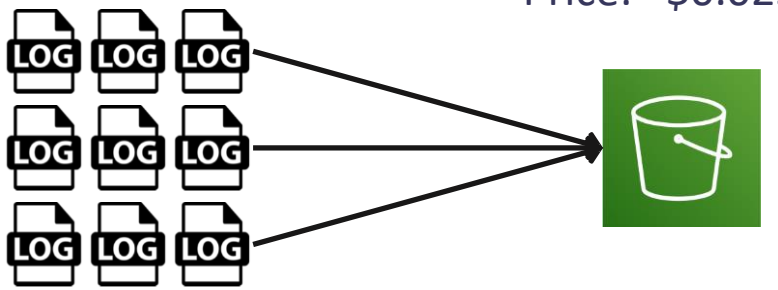


AWS S3 – Encryption



AWS Glacier

New 10GB/Month



Durability: 99.999999999%
Price: ~$0.025 per GB

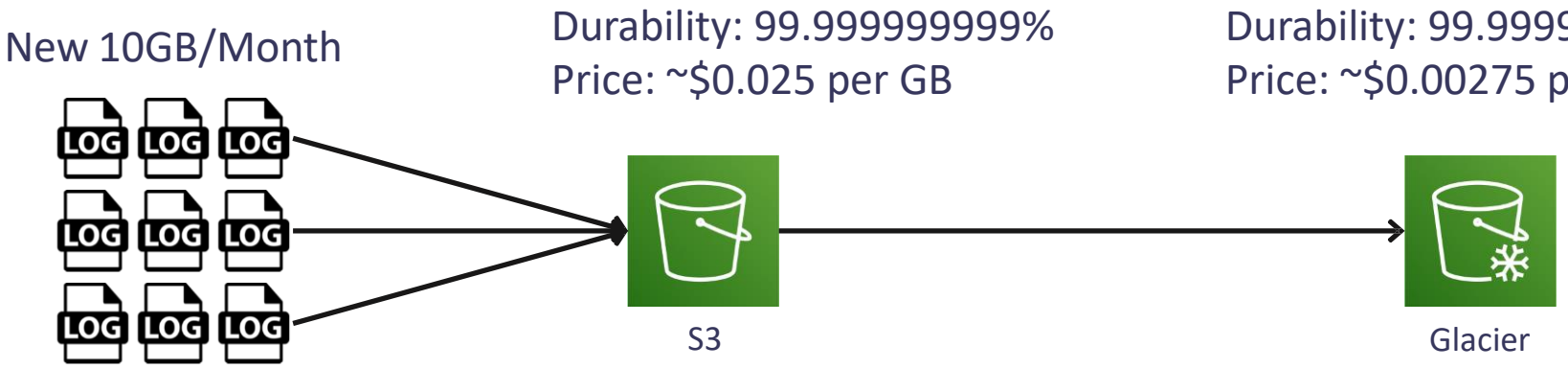
Do we need to read logs of last 12 months ago?

1st Month: 10GB ~ 0.25$
2nd Month: 20GB ~ 0.50$
12nd Month: 120GB ~ 3.00$

If new 1TB/Month =>

1st Month: 1TB ~ 25$
2nd Month: 2TB ~ 50$
12nd Month: 12TB ~ 300$

AWS Glacier



1st Month:	10GB ~ 0.25$	10GB ~ 0.0275$
2nd Month:	20GB ~ 0.50$	20GB ~ 0.0550$
12nd Month:	120GB ~ 3.00$	120GB ~ 0.3300$

If new 1TB/Month =>

1st Month:	1TB ~ 25$	1TB ~ 0.25$
2nd Month:	2TB ~ 50$	2TB ~ 5.50$
12nd Month:	12TB ~ 300$	12TB ~ 33.00$

Cost down: ~90%

AWS Glacier

- When to use?
  - Read less
  - Long term archive
  - Big amount
  - Not require to fast access
  - Reduce storage cost

- Cons:
  - Take time to access/retrieve file

AWS Database Services

<RDBMS>

RDS

<NoSQL>

DynamoDB

DocumentDB
(MongoDB compatibility)

<Key-Value DB>

ElastiCache

<Data Warehouse>

Redshift

<DB Migration Tool>

Database Migration Service
(DMS)

AWS Compute Services

<Instance Based>

EC2

Beanstalk

Lightsail

ECS
(Instance)

<Serverless>

Lambda

ECS Fargate

Batch

Amazon EC2 – Management

- Resources
  - Instance
  - Tag
  - AMI
  - EBS
  - Snapshot
  - Network

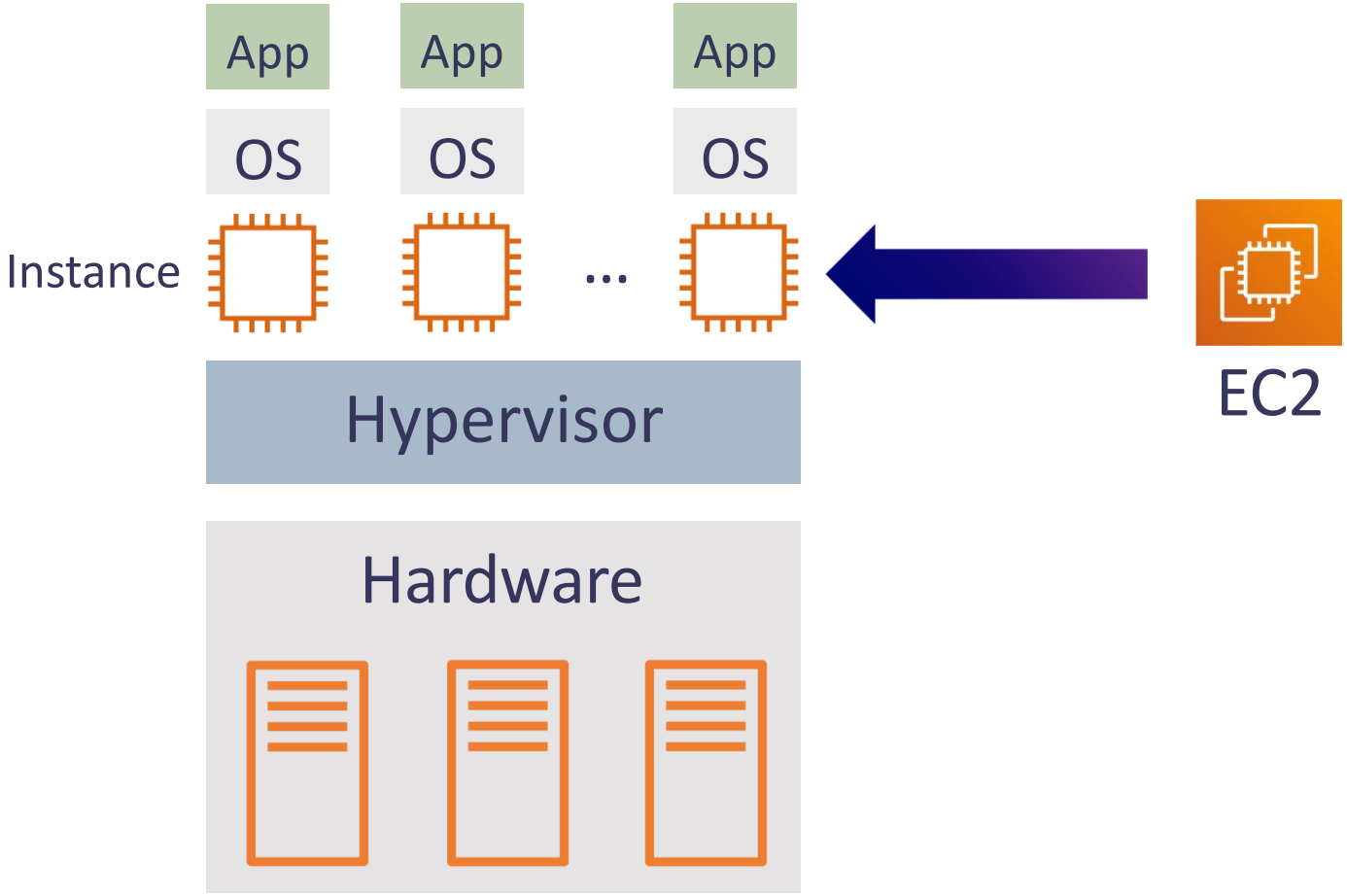
- Availability
  - Region
  - AZ

- Management
  - Administration
  - Monitoring
  - Logging

- Pricing Model
  - On Demand
  - Spot
  - Reserved (RI)

Amazon EC2

- EC2 – **E**lastic **C**ompute **C**loud is virtual server in the cloud



EC2 Instance Specifications

Instance Generation

General Purpose

T2 M4 A1

Memory Optimized

X1 R4 z1d HMI

Accelerated Computing

P3 G3 F1

Compute Optimized

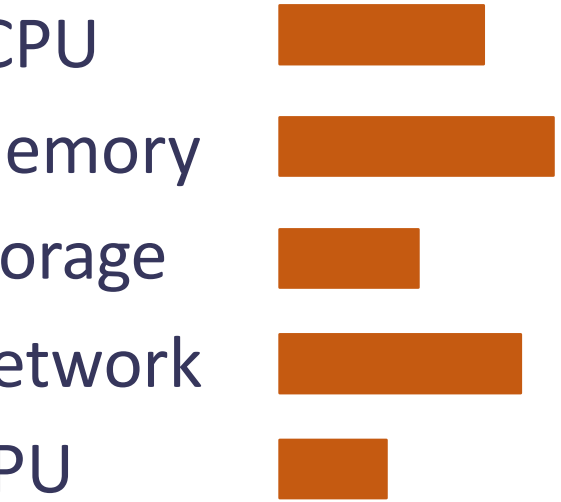
C5

Storage Optimized

I3 D2 H1

Instance Types

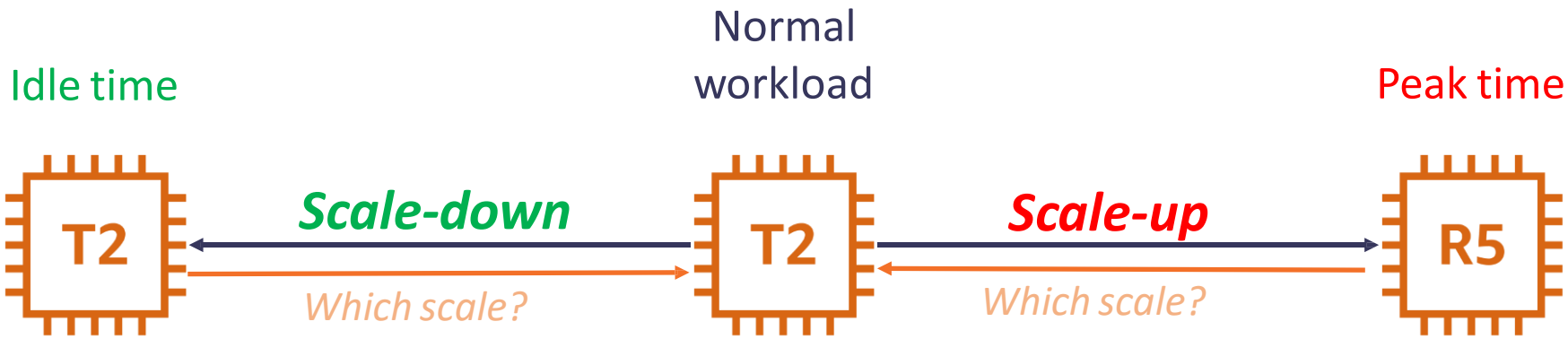
- vCPU
- Memory
- Storage
- Network
- GPU



t2.2xlarge

Instance Generation | Instance Size

EC2 Instance – Vertical Scaling



t2.nano instance

vCPU: 1
Mem: 0.5GB
Storage: EBS-only
Network: low

t2.medium instance

vCPU: 2
Mem: 4GB
Storage: EBS-only
Network: Low to Moderate

r5d.12xlarge instance

- *vCPU: 48*
- *Mem: 384GB*
- *Storage:* 2 x 900 NVMe SSD (*)
- *Network: 10*Gbps

EC2 Instance Specifications

Scope

Availability Zone

Large Number of Instances

OS

Amazon Linux

Windows

Ubuntu

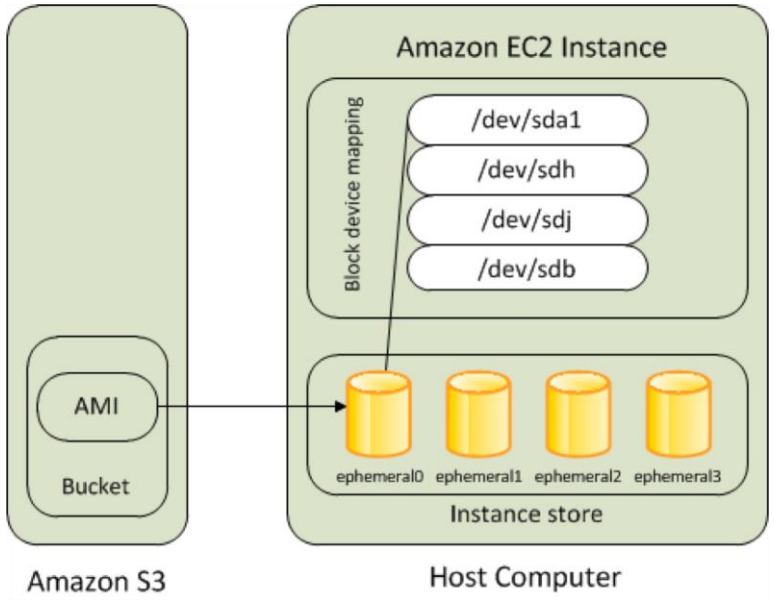
Red Hat

SUSE Linux

...

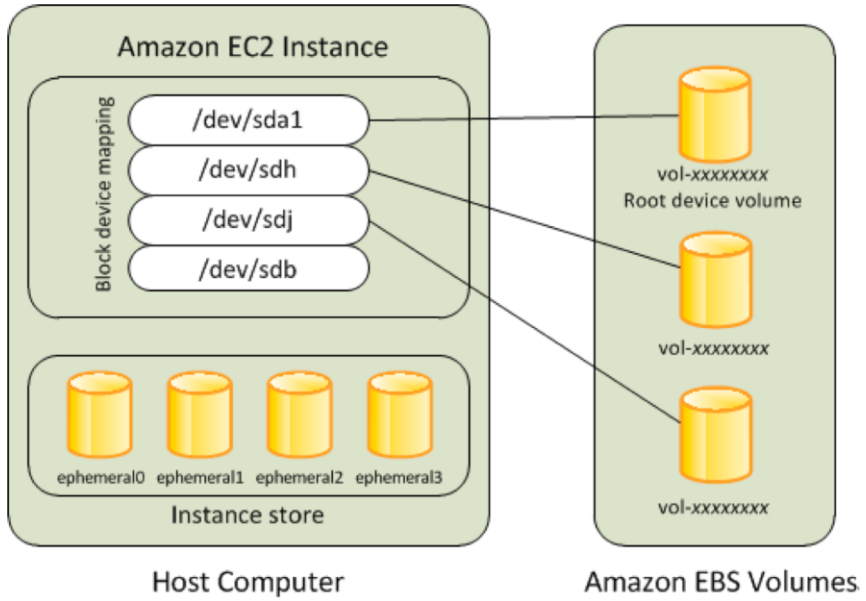
EC2 Instance - Root Device Volume

Instance Store-backed Instance



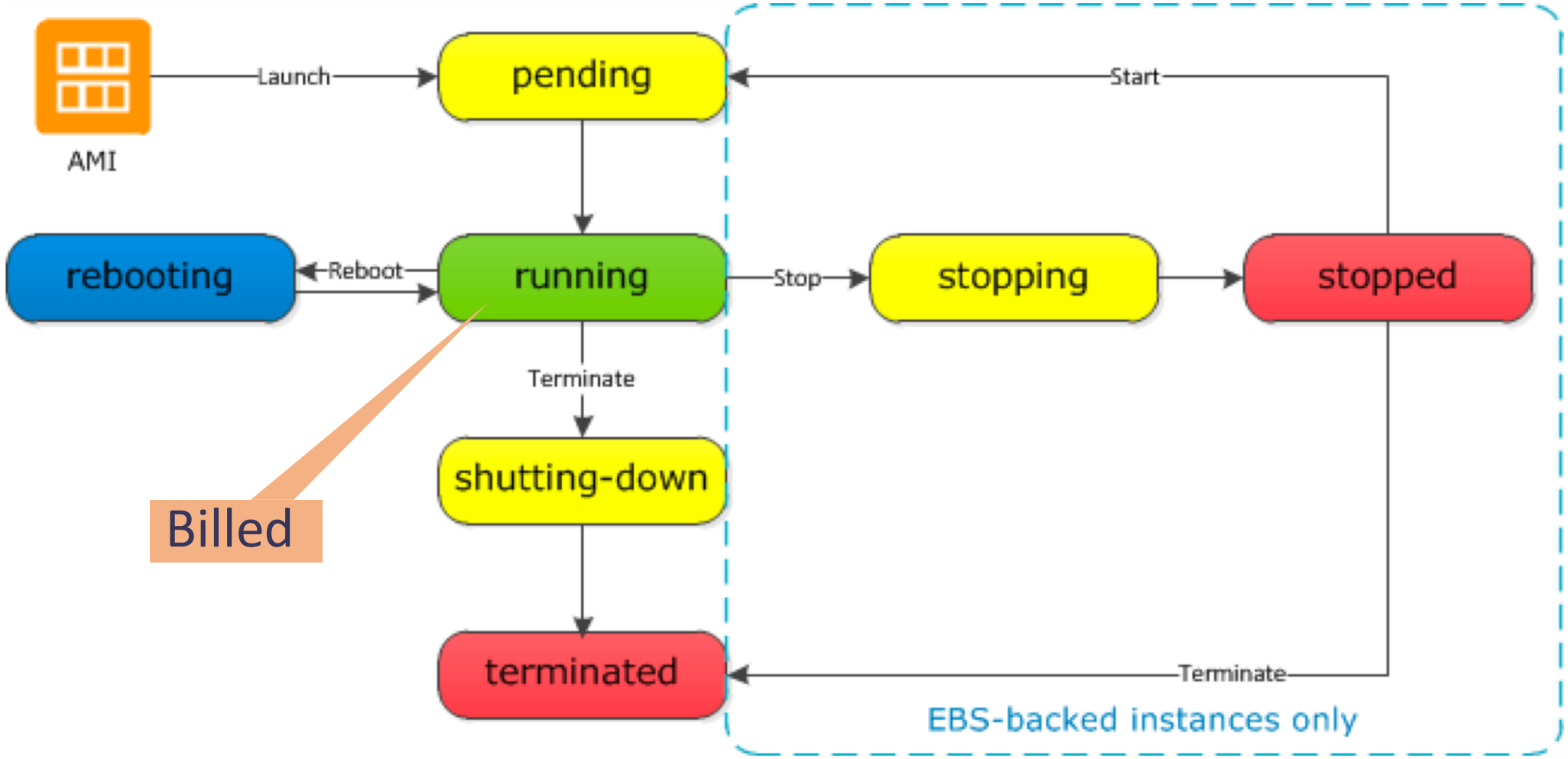
- Do not support stop action
- Cannot be restored when instance failed
- Clear data after instance termination
- Recommend: Temporary storage of information that changes frequently, such as buffers, caches, scratch data

EBS-backed Instance



- Full life cycle of EC2 instance
- Can be restored by EBS reusing
- Data can be remained in EBS
- Recommend: any purpose

EC2 Instance Life Cycle



EC2 Instance – Userdata vs Metadata

Instance Userdata

- Can be 

or 

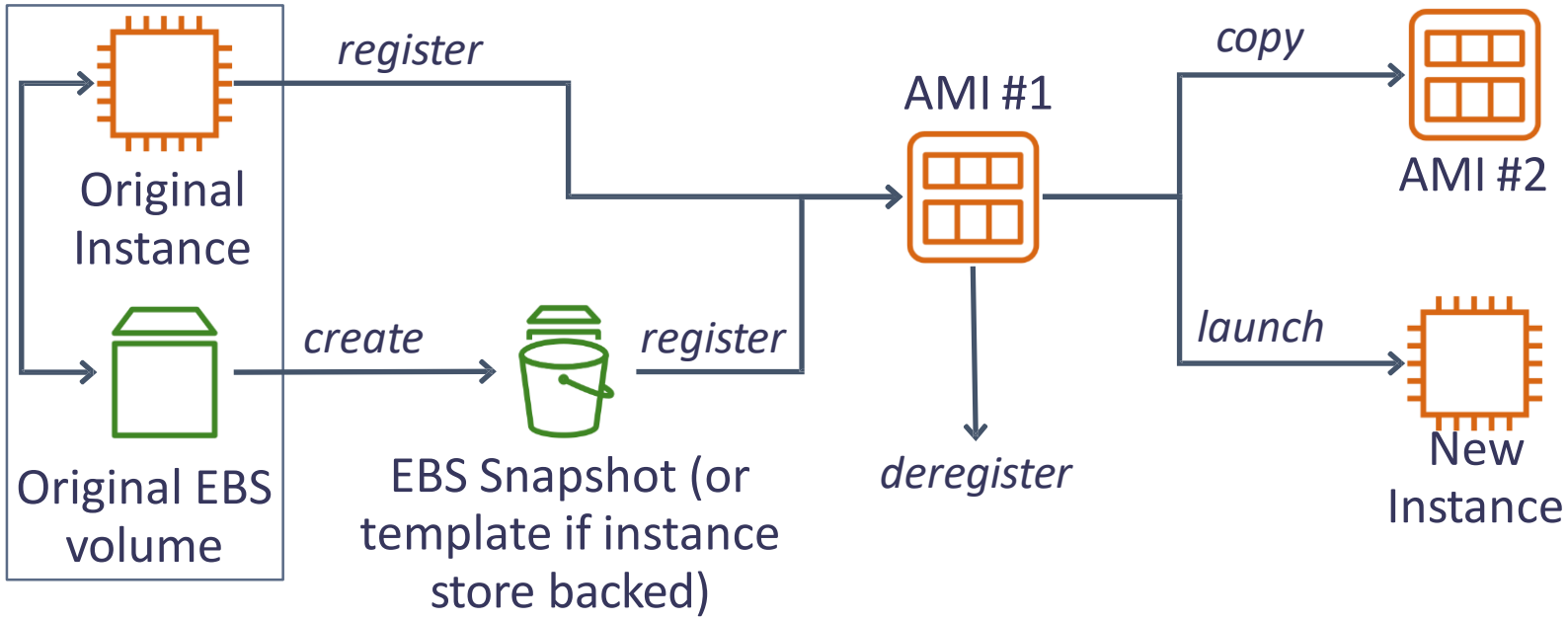
- Purpose: running commands on your Linux instance at launch time
- How to use:
  - Shell Script: start with `#!/bin/bash`
  - Cloud-init Directives
  - AWS Console
  - AWS CLI (*need base64 encoded userdata*)
- By **default**, run **only** during the **first boot cycle** when an instance is launched (*)

Instance Metadata

- Is **data about your instance**
- Purpose: You can use to configure or manage the running instance
- How to use: access to http://169.254.169.254/latest/meta-data/
- Content: ami-id, hostname, instance-id, local-hostname, local-ipv4, profile, public-ipv4, security-groups,…

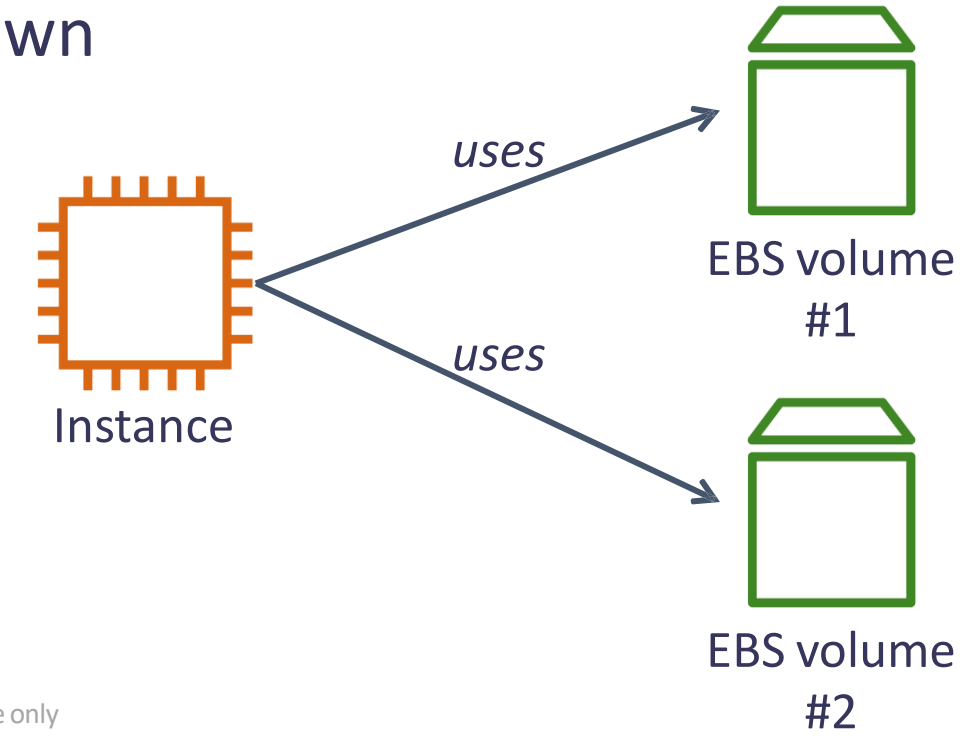
AWS EC2 - AMI

- AMI – **A**mazon **M**achine **I**mages
- Provides the information required to launch an instance
- Including:
  - 1 or more EBS snapshot
  - All OS configuration of the original EC2 instance



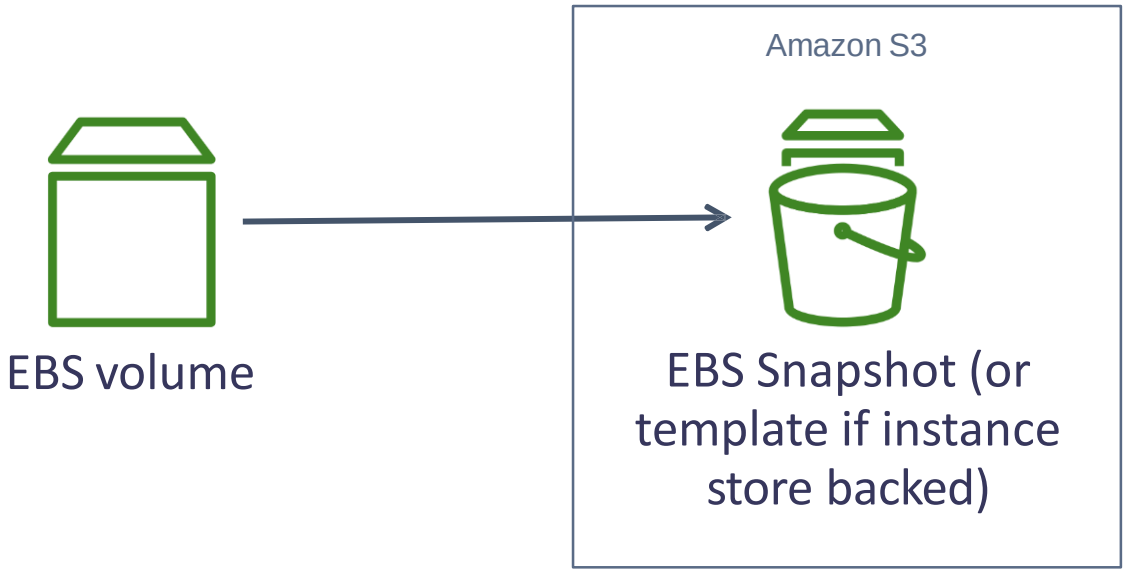
AWS EBS

- EBS – **E**lastic **B**lock **S**torage
- Provides persistence storage volumes for EC2
- Data is stored in data blocks. Several blocks, segments build a file
- Low Price
- Fast scale up-down
- AZ scope



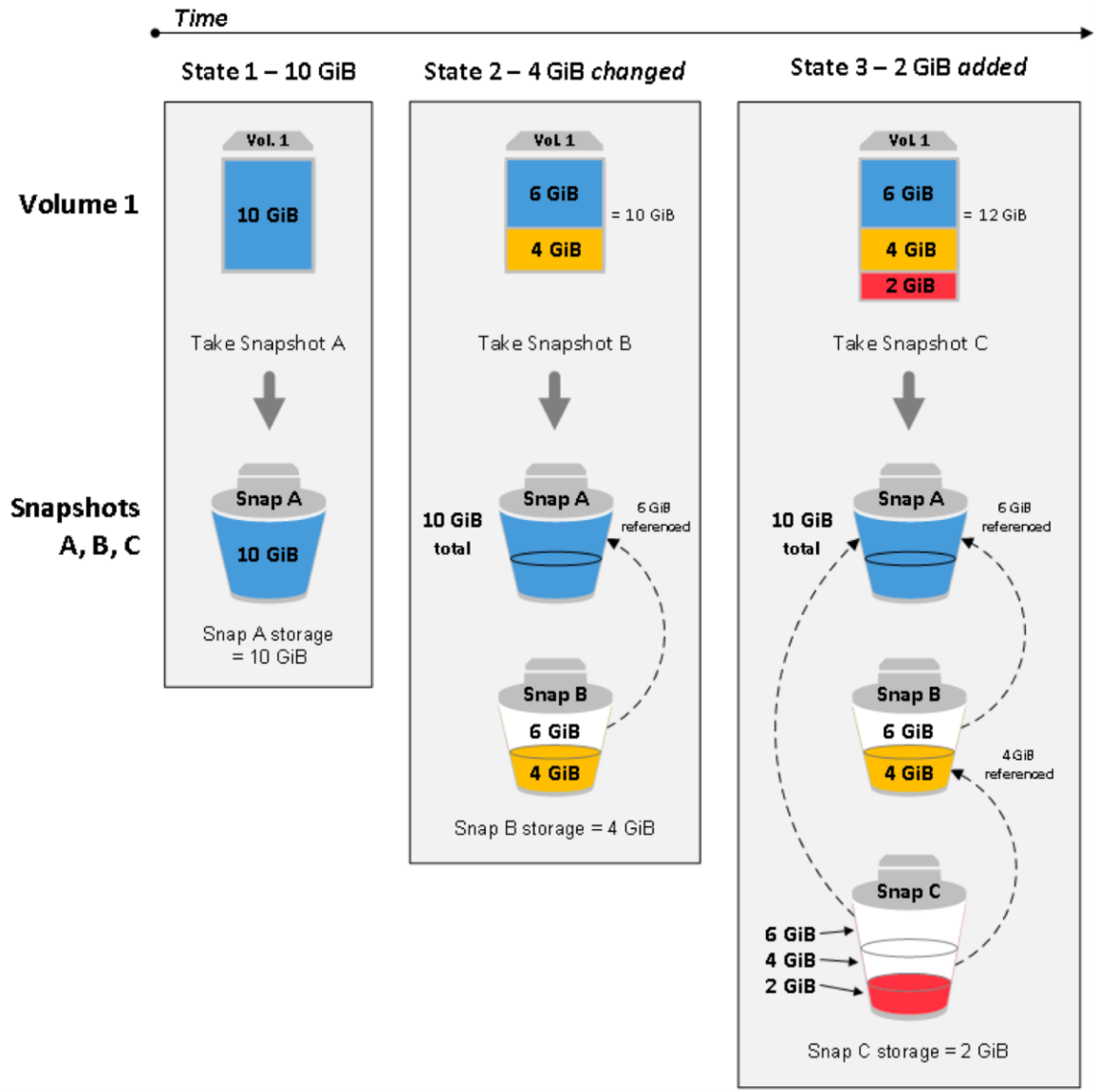
AWS EBS - Snapshot

- Back up the data on EBS volumes to Amazon S3 by taking point-in-time snapshots
- I*ncremental* backups
- When delete a snapshot, only the data unique to that snapshot is removed



AWS EBS – Incremental Snapshot

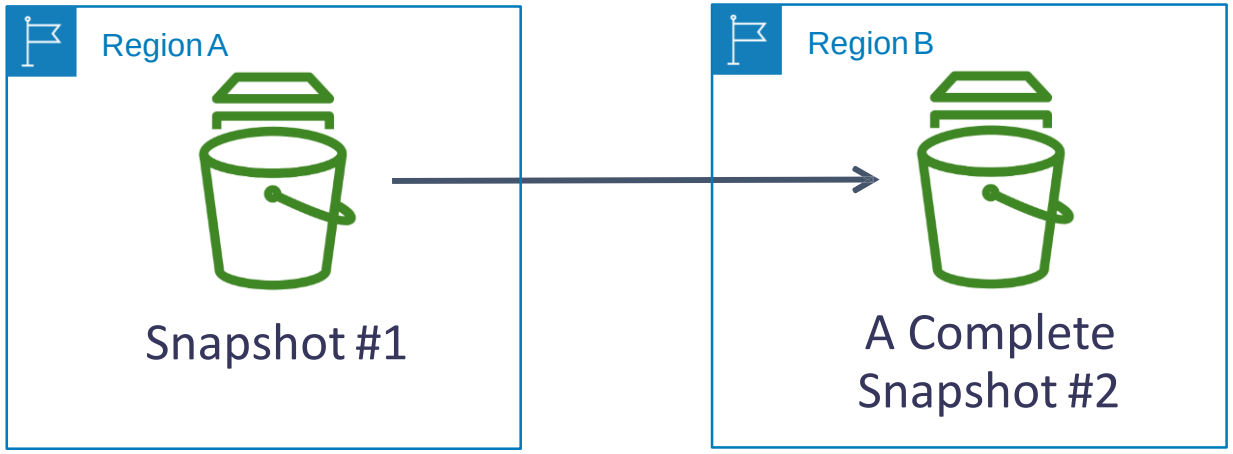
- Reduce cost
- Fast
- Reference snapshot



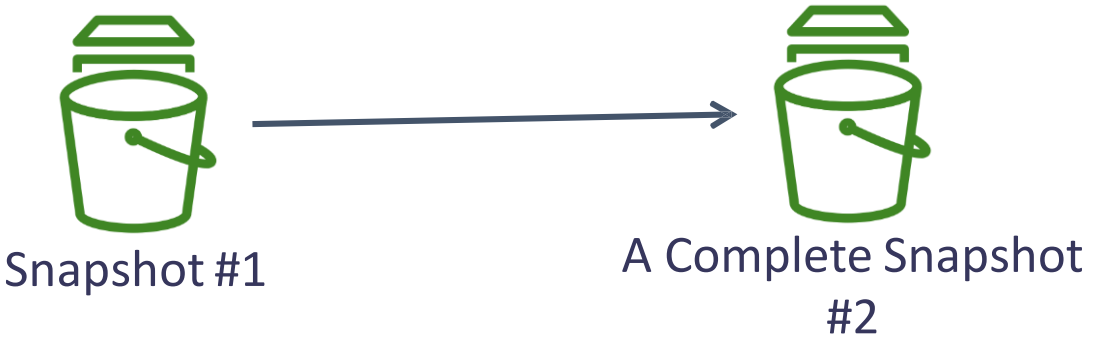
(From AWS Document)

AWS EBS – Copy Snapshot

- Copy to other **region** or other **account**

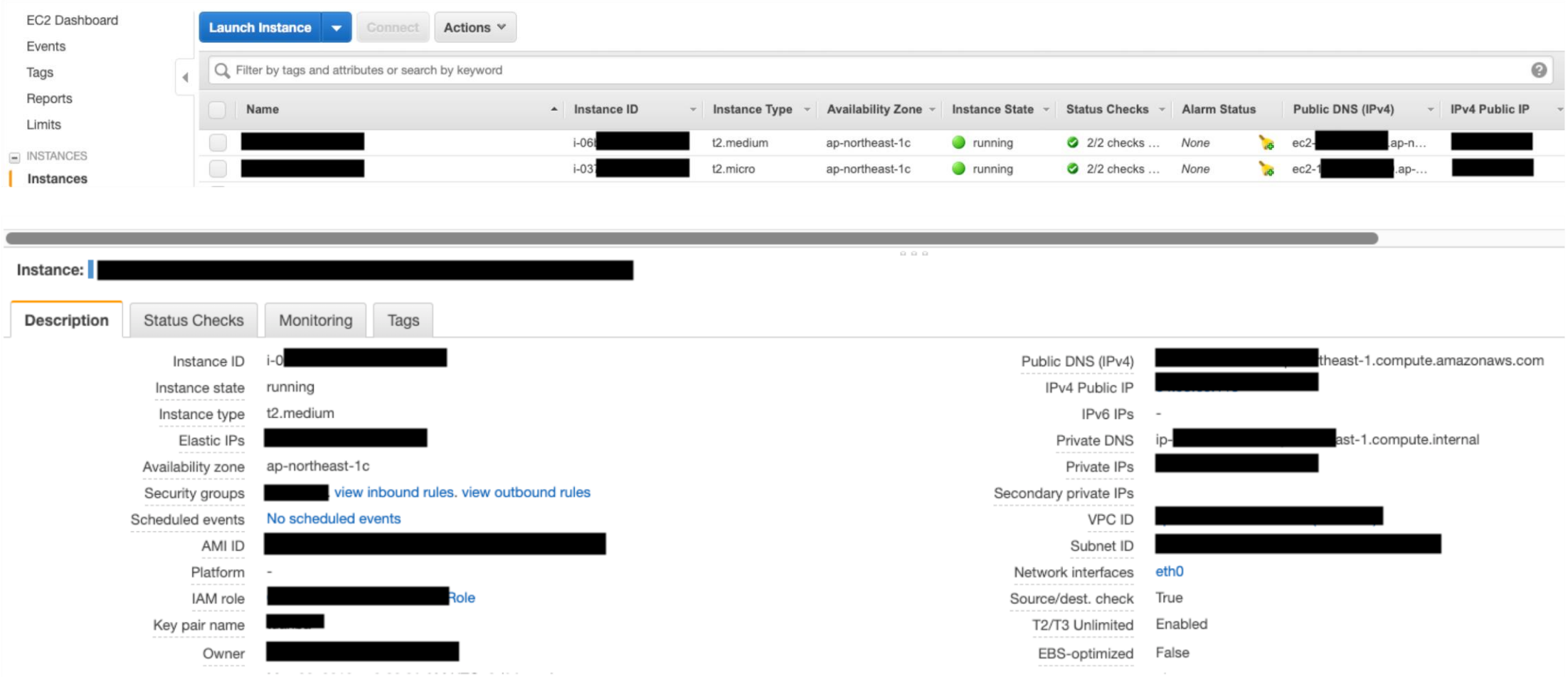


- Copy with new encrypt key (CMK)

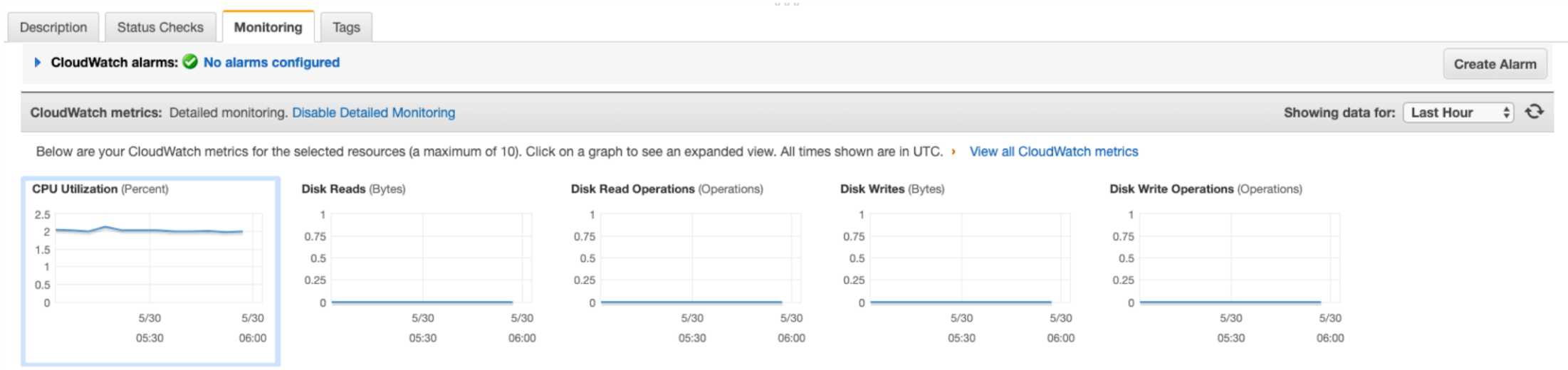


- Other: reference snapshot

AWS EC2 Management



AWS EC2 Management



AWS EC2 Purchasing Option

https://aws.amazon.com/ec2/pricing/on-demand/

On-Demand

- Pay for compute capacity by the second
- No commitment
- No Upfront Cost
- Short-term, dev/test env,…

Spot

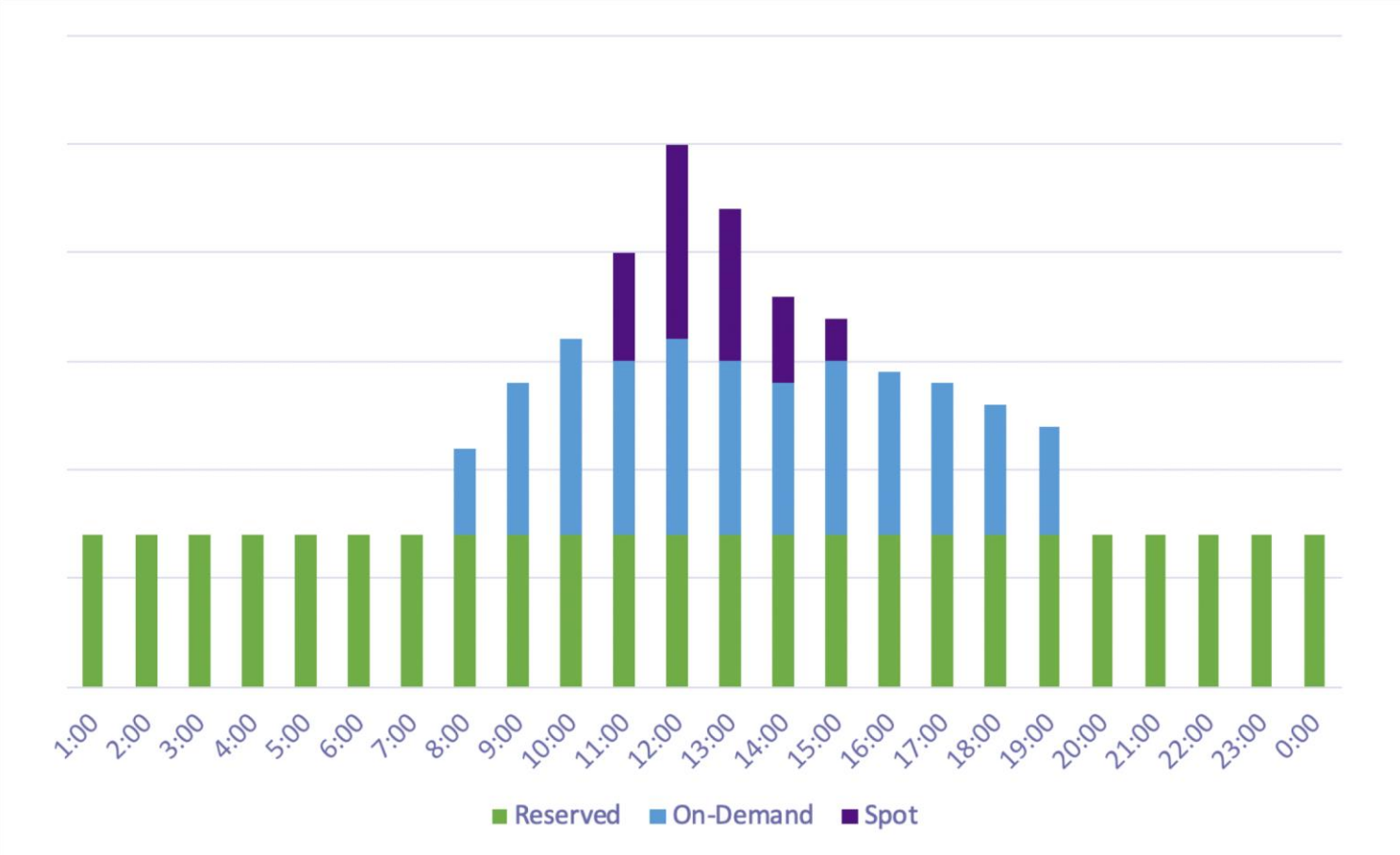
- Pay less with the possible price when reach it (~90% off)
- App must has flexible start and end times

Reserved

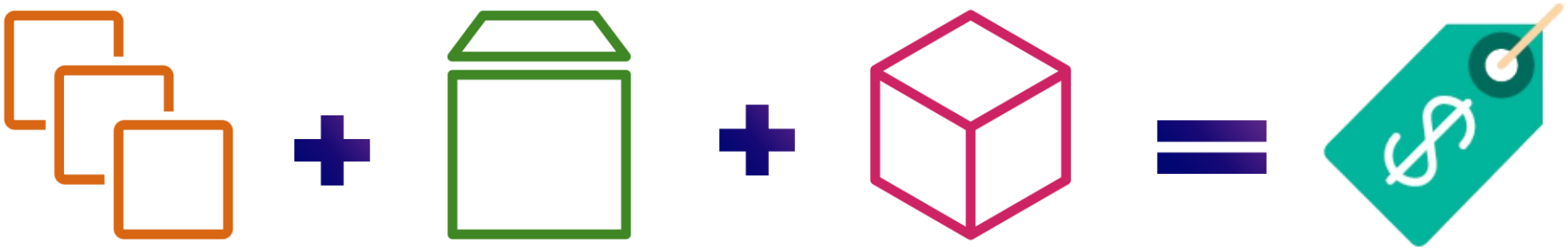
- Get significant discount (~75% off)
- long-term commitment (1-3 years)
- Steady state usage
- Production env

AWS EC2 – Best Cost

- Combine 3 options to get the best cost:
  1. Steady-state workloads: Use Reserved Instances
  2. Scale on peak: use flexible On-demand and Spot instance



AWS EC2 Cost Calculation



AWS EC2 Practice Time

- Create EC2 instance
- Access into that instance
- Create backup of instance, EBS
- Restore instance, EBS

Self Practice at home

- Create and attach new EBS to existing instance

Monitoring

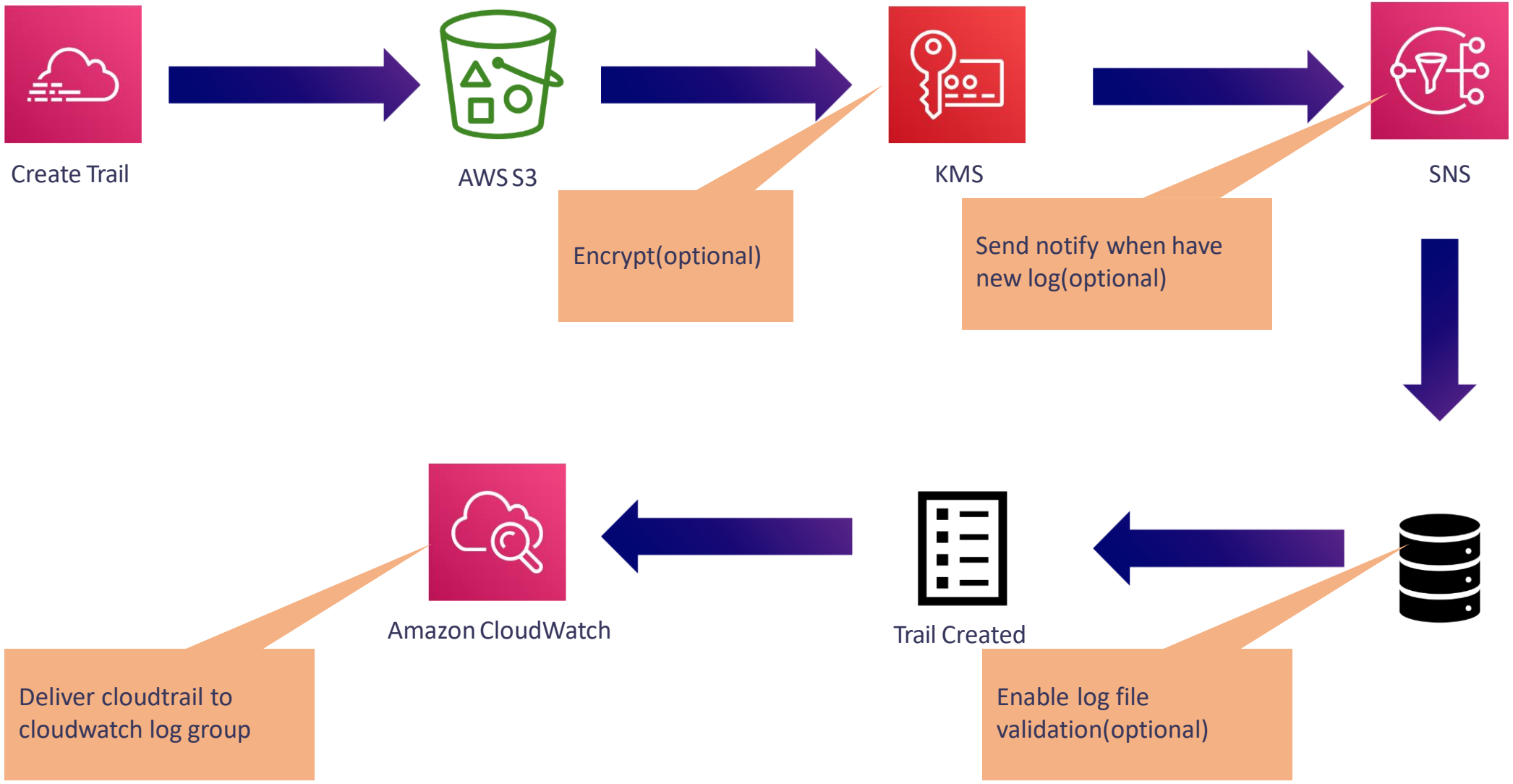
AWS CloudTrail

- Collect all behaviors of aws account
- With default setting can query 3 months event
- Can config with export to S3,Cloudwatch
- Base on events can describe abnormal behaviors

Amazon CloudWatch

- Can collect metrics of critical service(EC2, RDS, Beanstalk, ElasticSearch,…
- Base on metrics can create dashboard to monitoring
- Base on metric can create alarm to send operators(CPU is high, Free disk low …)

AWS CloudTrail-Process Flow



AWS Cloud Trail-Monitoring use case

- Log API call
- Starting, stopping, rebooting and terminating EC2
- Change security policies within IAM and S3
- Failed login attempts to the Management Console
- Integrate with Lambda and SNS to alert for failed authorization

AWS CloudWatch

Alarm

Base on metric can trigger actions: send sns, trigger auto scalingz

Event

Event to trigger action: send sns, sqs,…

Metric

Metric of aws service: EC2, RDS, AutoScaling Group, …

Log

Log group to collect log of service, application collect by cloudwatch agent

CW Agent

Collecting Metrics and Logs from Amazon EC2 Instances and On-Premises Servers