AWS Compute and Storage Services

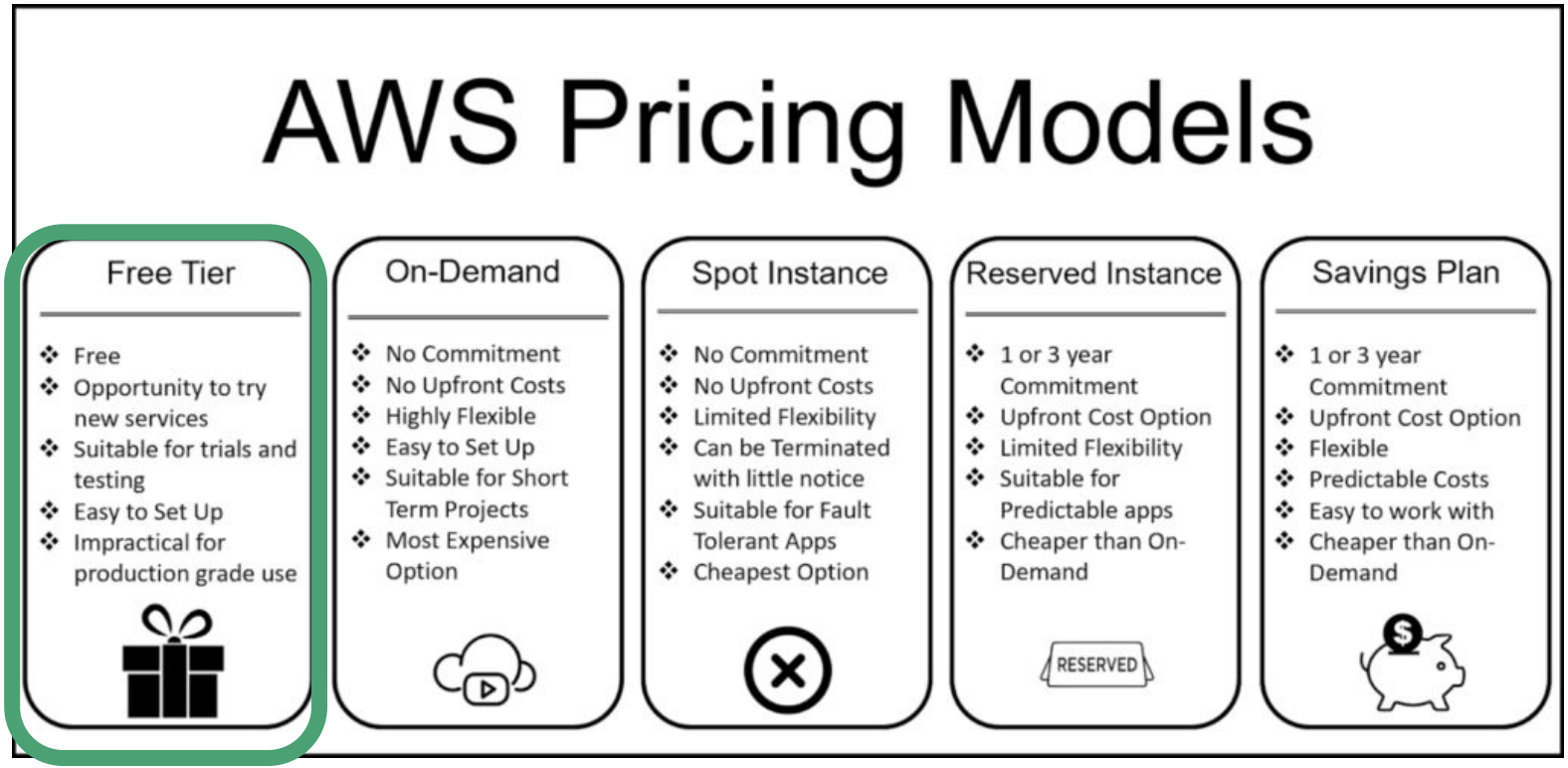
By Aditya Putra

me@adityaputra.com
https://github.com/adityaputra/praktisimengajar-cloudcomputing

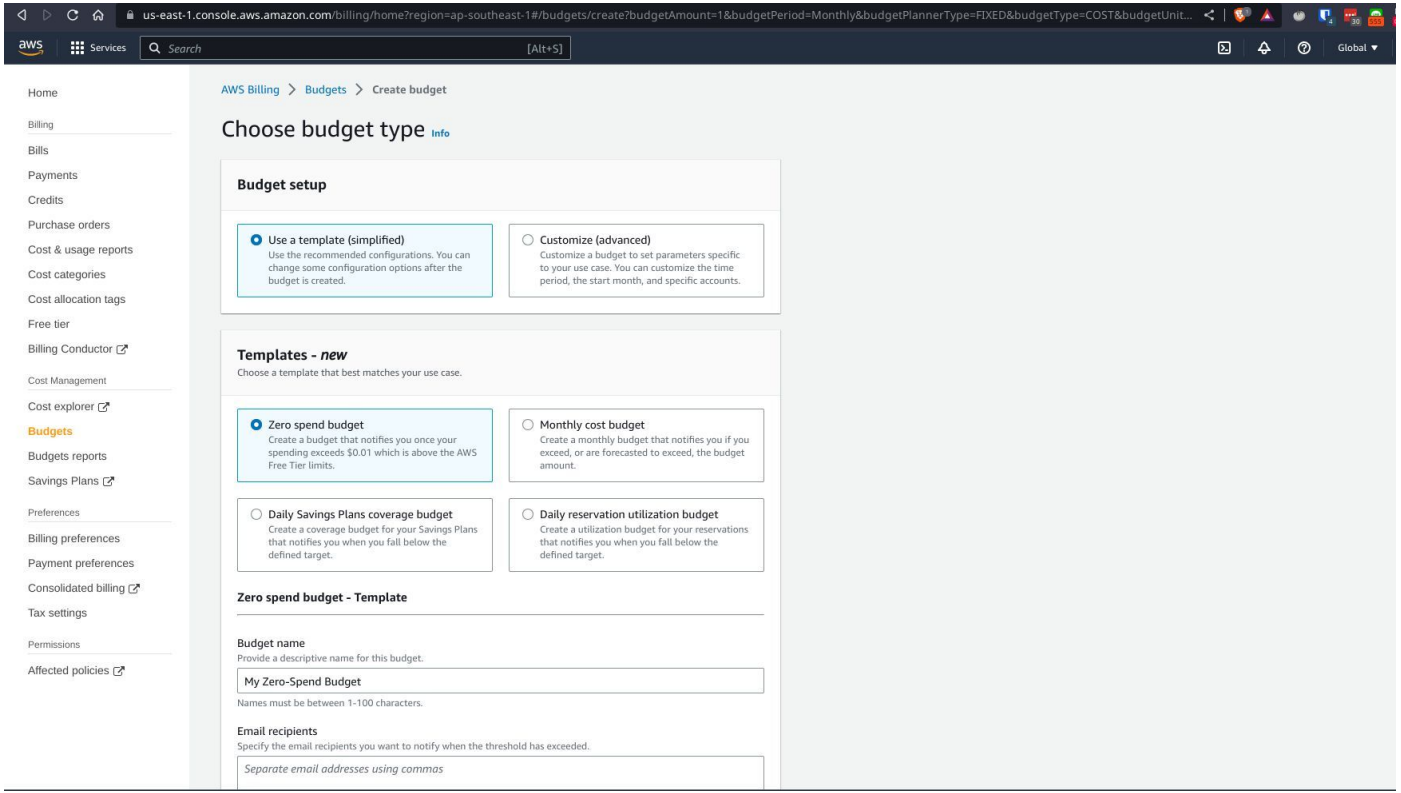
What we're learning today

- AWS Elastic Compute Cloud (**EC2**) instances and instance types
  - Amazon Machine Images (**AMI**s)
  - AWS Elastic Block Store (**EBS**) volumes and snapshots
  - AWS Auto Scaling and others
- AWS Simple Storage Service (**S3**) buckets, objects, and lifecycle policies
  - AWS **Glacier** for long-term data archiving
  - AWS **Snowball** for large-scale data transfer

Before that, let's learn briefly about AWS billing



Set your AWS budget before it’s too late!



After finishing all the tests,
be sure to remove ALL resources
to avoid unexpected charges
in the future

AWS Elastic Compute Cloud (EC2)

Amazon Elastic Compute Cloud (EC2) is a web service that provides **scalable computing** capacity in the cloud. It allows users to launch and manage **virtual machines, known as instances**, which can be configured with a wide range of operating systems and software.

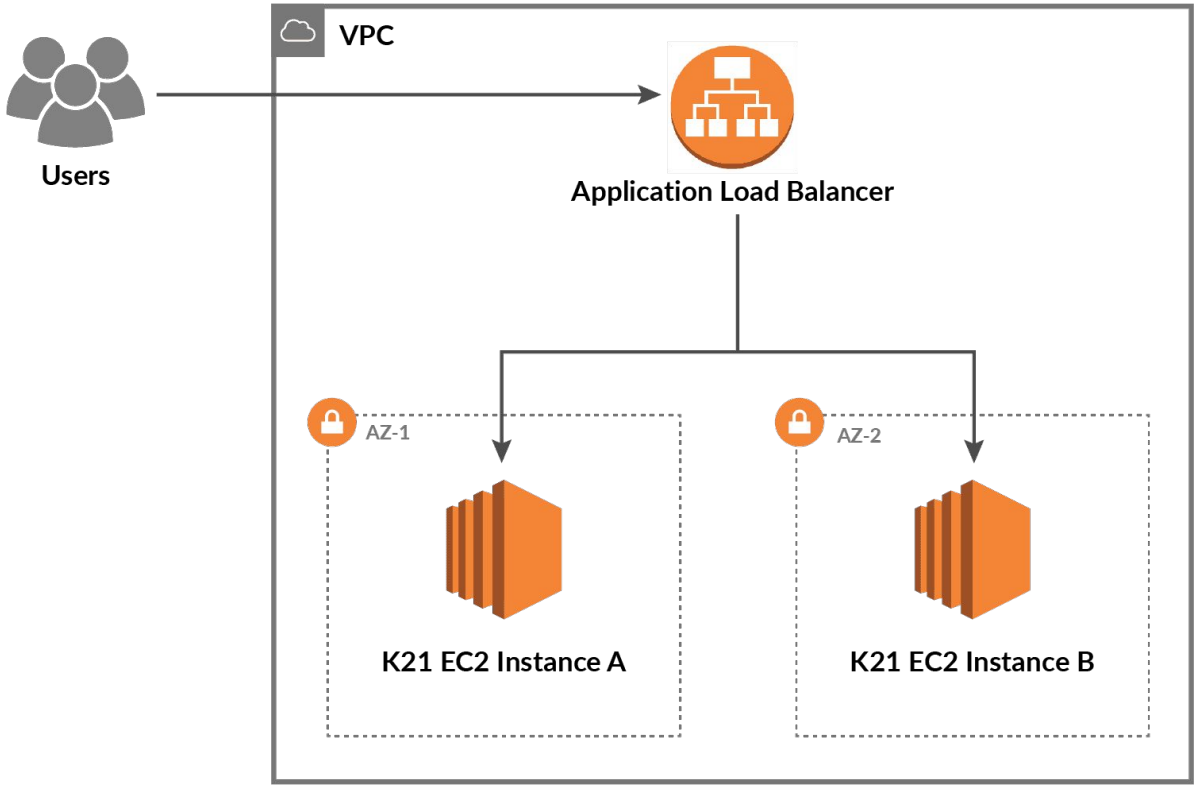
AWS EC2 features

- Elasticity and scalability
- Multiple instance types: general-purpose, compute-optimized, memory-optimized, and storage-optimized instances.
- Multiple operating systems: Windows, Linux, and macOS.
- Multiple regions and availability-zones
- Customizable security: virtual private clouds (VPCs), security groups, and network access control lists (ACLs).
- Integration with other AWS services
- Monitoring and management tools: Amazon CloudWatch and AWS Systems Manager
- Flexible pricing options
- Load balancing & auto-scaling

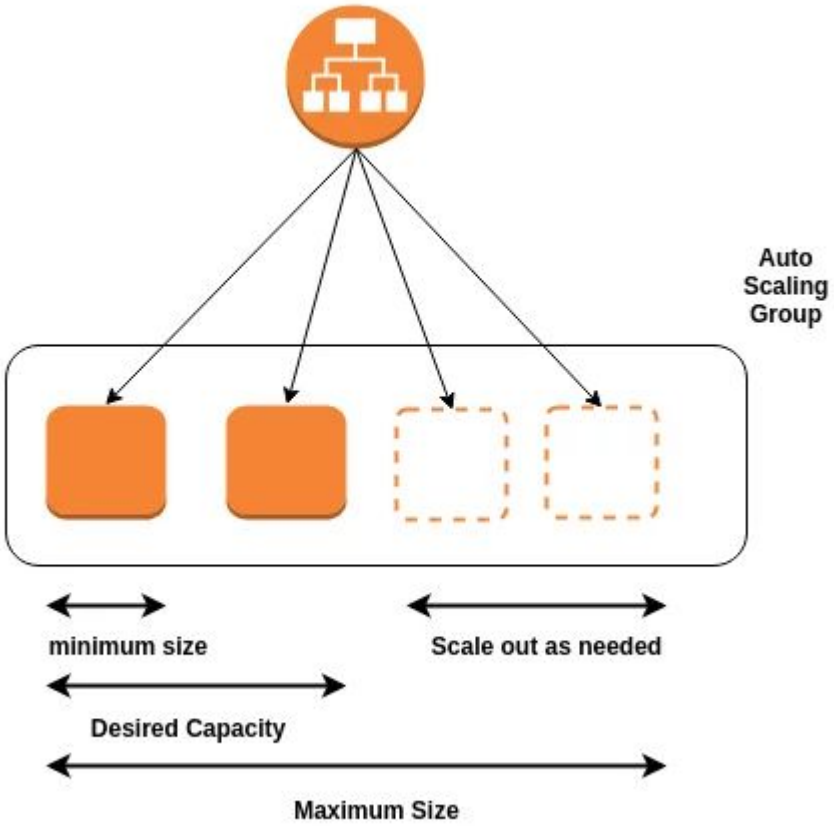
AWS EC2 common use-cases

- Hosting web applications and websites
- Running batch processing jobs
- Machine learning and AI
- Big data processing
- Gaming and media streaming
- DevOps and testing
- High Performance Computing

Load balancing with EC2



Auto-scaling with EC2



Demo: EC2

- Regions
- Dashboard
- Instances
- AMIs
- EBS: volumes, snapshots, lifecycle manager
- Elastic IPs
- Security groups
- Load Balancing
- Auto Scaling

Assignment 1

For each working groups, please do the following tasks. Make screenshot of each steps, and finally make screenshot of the server that you have just created:

1. Register for an AWS account, make sure that billing is enabled and use only free tier
2. Enable AWS budget for **zero spend budget**
3. Create an EC2 instance with the following specification:
    a. Region name: singapore
    b. Instance size: t2.micro
    c. Elastic IP: yes
    d. OS: Ubuntu Server 22.04 LTS
    e. By using security groups, block access to the server from the internet except for SSH port 22
4. Access the server via SSH
    a. ssh ubuntu@elastic_ip_address -i /path/to/ssh/key
5. Submit the practicum report

AWS Simple Storage Service (S3)

Amazon S3 is a highly **scalable and available object storage service** that allows users to store and retrieve any amount of data from anywhere on the web. It provides a simple web services interface, and comprehensive **security** and **compliance** capabilities. S3 objects can be any type of file or data, and can be stored in a **bucket**, which is associated with a unique name across all of AWS. S3 provides features like **versioning, lifecycle policies, and cross-region replication, and integrates with other AWS services** to provide additional functionality and security. It is used for backup and restore, **disaster recovery**, **big data** analytics, multimedia **storage and distribution**, and mobile and web **application hosting**.

AWS S3 features

- Highly Scalable
- Security and Compliance
- Durability and Availability: 99.999999999% durability
- Object Lifecycle Management
- Versioning: S3 provides versioning
- Access Management
- Integration with Other AWS Services

AWS S3 common use-cases

- Data Backup and Recovery
- Archiving
- Big Data Analytics
- Content Distribution
- Application Hosting
- Disaster Recovery
- Media Storage for Application

Demo S3

- Create an S3 bucket
- Setup lifecycle policy
- Setup versioning

Assignments 2

- Create an S3 bucket with the following specs:
  - Bucket name: praktisimengajar-pnm-cloudcomputing-kelas4[B/C]-namakelompok
  - Allow public access to the bucket
  - Upload a file called index.html into the bucket containing your names and student ID numbers
  - Make sure that the index.html file can be accessed publicly
  - Submit the complete URL of the index.html and write the report for the practicum

Thank you

me@adityaputra.com