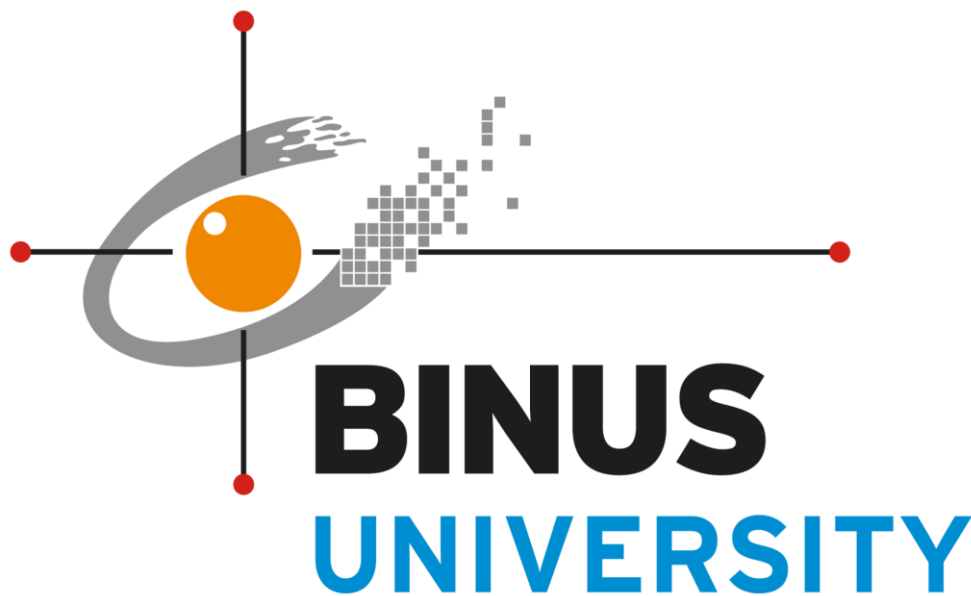


**Laporan Assessment of Learning (AoL)**  
**Big Data Infrastructure and Technology**



**Kelompok 6:**

**2501995093 - Caroline Angelina Sunarya**

**2502008006 - Jocelyn Verna Siswanto**

**2501980690 - Agnes Calista**

**2502020970 - Snow White Putri Safa**

**2502016935 - Adonia Gracio**

# Daftar Isi

## Bab I

### Pendahuluan

- A. Latar Belakang 3
- B. Tujuan Pengembangan 4

## Bab II

### Hybrid Data Hosting And Website Hosting

- A. Layanan AWS
  - 1. AWS Web Application Firewall (WAF) 5
  - 2. AWS Shield 5
  - 3. Amazon CloudFront 5
  - 4. Amazon Simple Storage Service (S3) 5
  - 5. Amazon Simple Notice Service (SNS) 6
  - 6. Amazon Route 53 6
  - 7. Amazon Elastic Container Registry (ECR) 6
  - 8. Amazon API Gateway 6
  - 9. Amazon Elastic Load Balancing (ELB) 7
  - 10. Amazon EC2 Auto Scaling 7
  - 11. Amazon ElasticSearch Services 7
  - 12. Amazon ElastiCache 8
  - 13. Amazon DynamoDB 8
  - 14. Amazon ECS Fargate 8
  - 15. AWS Site-to-Site VPN 8
  - 16. Customer Gateway 9
  - 17. Virtual Private Gateway 9
  - 18. VPC Endpoint 9
- B. Arsitektur
  - 1. Layanan AWS Cloud 10
  - 2. On-Premise Data Center 12

### Referensi

# Bab I

## Pendahuluan

### A. Latar Belakang

*Server* adalah sistem komputer yang memiliki layanan khusus berupa penyimpanan data. *Server* memainkan peran penting dalam menyediakan layanan akses yang lebih cepat untuk mengirim atau menerima data di *server*. Dalam bentuk fisiknya, *server* berbentuk jaringan komputer dan berukuran sangat besar serta memiliki beberapa komponen pendukung prosesor dan memori RAM berkapasitas besar [1]. Dikarenakan ukurannya yang sangat besar, *server* membutuhkan ruangan yang luas serta suasana lingkup yang memadai untuk menjaga kondisi *server* agar dapat beroperasi dengan optimal.

Memiliki *server* yang didedikasikan untuk bisnis kita sendiri tentunya dirasa ideal untuk sebuah perusahaan yang berbasis teknologi. Memiliki *server* fisik dapat berarti memiliki kendali terhadap sistem operasional dari bisnis serta bertanggung jawab secara penuh terhadap penyimpanan dan keamanan data. Meskipun memiliki keuntungannya sendiri, bergantung pada *server* fisik memiliki kekurangannya juga, seperti biaya operasional yang tinggi, kebutuhan untuk memelihara server secara teratur, skalabilitas terbatas, dan lain-lain [2]. Oleh karena kekurangan-kekurangan tersebut, banyak perusahaan-perusahaan beralih menggunakan layanan yang berbasis *cloud*.

Layanan-layanan komputasi berbasis *cloud* dapat menjawab kekurangan-kekurangan dari menggunakan *on-premises server*. Penggunaan *cloud computing* dapat menggantikan biaya tetap dengan biaya variabel, kemampuan skalabilitas menjadi tidak terbatas, dan lain-lain [3]. Hal-hal yang termasuk sebagai biaya tetap adalah biaya yang dikeluarkan untuk merakit *server*, penyewaan atau pembelian tempat untuk menyimpan *server*, serta perlengkapan dan peralatan yang diperlukan untuk menjaga kualitas dari *server*, misalnya AC untuk menjaga sirkulasi dari ruangan agar *server* tidak *overheat*. Alih-alih membayar biaya tetap, perusahaan dapat menggunakan layanan *cloud computing* dan hanya membayar biaya variabel seperti biaya penggunaan layanan *cloud* tertentu. Selain itu, menggunakan layanan *cloud computing* juga dapat memberikan kemampuan skalabilitas yang tidak terbatas pada bisnis, sehingga apabila aktivitas pada website yang di-hosting pada sebuah *platform cloud* sedang tinggi, user dapat meningkatkan skala kapasitas penggunaan layanan *cloud* sesuai dengan kebutuhannya saat itu, dan menurunkan kapasitas penggunaan layanan *cloud* ketika aktivitas pada website sedang menurun.

Amazon Web Services (AWS) merupakan salah satu penyelenggara layanan-layanan komputasi berbasis *cloud*. AWS adalah *platform cloud computing services* paling populer saat ini, dan hal tersebut bukanlah tanpa alasan. AWS tersedia pada 22 wilayah (*region*) yang tersebar di seluruh dunia dan hal inilah yang menjadikan AWS sebagai platform penyedia *cloud computing* dengan infrastruktur *cloud* global yang

paling luas. AWS menyediakan lebih dari 200 layanan yang menjadikan platform ini memiliki layanan *cloud computing* yang sangat lengkap. Layanan-layanan pada AWS meliputi komputasi, basis data, manajemen infrastruktur, pengembangan aplikasi, dan keamanan informasi [4].

Permasalahan yang diangkat dari *project* ini adalah perusahaan A ingin melakukan *hosting* data secara *hybrid* pada platform penyedia layanan *cloud computing* AWS. Perusahaan A merupakan sebuah perusahaan e-commerce. Pada saat musim-musim liburan, *traffic* pada aplikasi e-commerce selalu tinggi. Meskipun telah mempersiapkan kapasitas yang lebih pada *server*-nya untuk mengantisipasi *traffic* pada saat liburan, jumlah customer yang mengakses aplikasi e-commerce terus meningkat setiap tahunnya. Hal tersebut berpengaruh kepada performa aplikasi dan pengalaman customer pada saat mengakses aplikasi tersebut. Belakangan ini, perusahaan A sering menerima kritik dari customer mengenai aplikasinya, seperti performa aplikasi yang lambat, error pada saat ingin melakukan pembayaran, dan lain-lain.

Perusahaan A mengakui bahwa kritik-kritik yang diterima tersebut dikarenakan *traffic* aplikasi yang tinggi, serta kapasitas *server* yang tidak memadai. Setelah berkonsultasi dengan konsultan *cloud computing services*, perusahaan A ingin melakukan data hosting secara *hybrid* dikarenakan musim liburan sudah dekat dan waktu yang dibutuhkan untuk merakit *server* tidak mencukupi serta tempat untuk menyimpan *server* tambahan sangat terbatas. Sehingga, *on-premises server* hanya akan menyimpan data-data personal dari customer, seperti data-data perbankan (nomor kartu kredit), nomor telepon, dan lain-lain. Data personal tersebut diputuskan untuk disimpan pada *on-premises server* dikarenakan perusahaan A merasa bahwa data personal dari customer sangat riskan untuk disimpan secara *cloud*, dan ingin mengantisipasi kebocoran data dari penyedia layanan *cloud*. Selain itu, perusahaan A juga ingin meng-*hosting* website-website terbarunya dalam *cloud*. Perusahaan A mempercayakan perancangan arsitektur *cloud* secara keseluruhan kepada konsultan *cloud computing*.

## **B. Tujuan Pengembangan**

Adapun tujuan-tujuan dari pengembangan arsitektur *cloud computing* tersebut sebagai berikut:

1. Memberikan gambaran layanan-layanan apa saja yang dibutuhkan dan keterkaitan yang dimiliki antar layanan *cloud* untuk melakukan *hosting* data secara *hybrid*.
2. Memberikan gambaran besaran biaya yang dikeluarkan untuk meng-*hosting* data secara *hybrid*.

## **Bab II**

### **Hybrid Data Hosting And Website Hosting**

#### **A. Layanan AWS**

##### **1. AWS Web Application Firewall (WAF)**

AWS Web Application Firewall atau AWS WAF merupakan layanan firewall aplikasi web yang mengontrol akses ke konten dengan mengizinkan atau menolak permintaan web berdasarkan kriteria yang ditentukan, seperti alamat IP asal permintaan. AWS WAF membantu melindungi aplikasi web dari serangan yang dapat mempengaruhi ketersediaan aplikasi, membahayakan keamanan data, atau menghabiskan terlalu banyak sumber daya. AWS WAF melindungi aplikasi web dengan memantau permintaan HTTP yang diarahkan ke sumber daya aplikasi web [5].

##### **2. AWS Shield**

Dengan membangun aplikasi di AWS, user dapat memanfaatkan perlindungan AWS tanpa biaya tambahan. DDoS (Distributed Denial of Service) adalah salah satu serangan paling umum terhadap aplikasi. Oleh karena itu, perlindungan terhadap serangan DDoS sangat penting untuk aplikasi. AWS Shield dapat memberikan layanan perlindungan ancaman terkelola yang meningkatkan kondisi keamanan dengan kemampuan perlindungan, mitigasi, dan respons DDoS tambahan. AWS Shield dapat secara otomatis membersihkan lalu lintas internet yang gagal pada tingkat tertentu, meminimalkan *downtime* dan latensi aplikasi, serta memantau dan melindungi hingga 1.000 jenis *resource* [6].

##### **3. Amazon CloudFront**

Amazon CloudFront adalah layanan jaringan pengiriman konten (CDN) yang dirancang untuk kinerja tinggi, keamanan, dan kenyamanan bagi pengembang. AWS CloudFront mengirimkan perangkat lunak, SDK, video, dan konten lainnya kepada pelanggan dengan aman dan dengan kecepatan transfer tinggi. AWS CloudFront memungkinkan pengiriman situs web yang cepat dan aman serta diimplementasikan untuk pengiriman konten dinamis dan API yang dipercepat, streaming video langsung dan sesuai permintaan, serta distribusi tambahan dan pembaruan [7].

##### **4. Amazon Simple Storage Service (S3)**

Amazon Simple Storage Service (S3) adalah layanan penyimpanan objek yang menawarkan skalabilitas, ketersediaan data, keamanan, dan kinerja terdepan di industri. Amazon S3 menyimpan semua konten katalog statis, seperti gambar produk, panduan pengguna, dan video, serta semua file log layanan mikro. Amazon

S3 dapat menyederhanakan, mengatur, dan menyesuaikan akses ke data dengan menyediakan kemampuan manajemen untuk memastikan layanan memenuhi kebutuhan spesifik bisnis, organisasi, maupun kebutuhan pribadi user [8].

## **5. Amazon Simple Notice Service (SNS)**

Amazon Simple Notice Service (SNS) adalah layanan terkelola yang memungkinkan penerbit mengirim pesan ke pelanggan. Dengan menggunakan Amazon SNS, pengaturan, pengaksesan, dan pengiriman notifikasi melalui *cloud* dapat dengan mudah dilakukan. Amazon SNS mengirimkan pemberitahuan dengan dua cara: aplikasi ke aplikasi atau *application-to-application* (A2A) dan aplikasi ke orang atau *application-to-person* (A2P). Amazon SNS dapat digunakan untuk mengintegrasikan aplikasinya dengan perpesanan FIFO, mengenkripsi pesan notifikasi dengan aman, mengumpulkan dan menyegmentasikan kejadian dari lebih dari 60 layanan AWS, dan mengirim pesan teks ke pelanggan di 240 negara [9].

## **6. Amazon Route 53**

Permintaan DNS ke situs web ditangani oleh Amazon Route 53. Layanan tersebut menghubungkan permintaan pengguna ke aplikasi web yang berjalan di AWS atau *on-premises*. Amazon Route 53 digunakan untuk mengelola lalu lintas jaringan di seluruh dunia, membangun aplikasi dengan ketersediaan tinggi, dan mengkonfigurasi DNS pribadi. AWS Route 53 menerjemahkan nama URL seperti [www.wordpress.com](http://www.wordpress.com) ke alamat IP numerik yang sesuai (198.143.164.252 dalam contoh ini). Dengan cara ini, AWS Route 53 menyederhanakan perutean pengguna Anda ke aplikasi web melalui arsitektur cloud Anda. Layanan DNS AWS Route 53 menghubungkan permintaan Anda ke penyeimbang beban ELB, *instance* Amazon EC2, bucket Amazon S3, dan infrastruktur lain yang berjalan di AWS [10].

## **7. Amazon Elastic Container Registry (ECR)**

Amazon Elastic Container Registry (ECR) memudahkan untuk menyimpan, membagikan, dan menerapkan gambar produk. Amazon ECR adalah kontainer yang dikelola sepenuhnya yang menawarkan *hosting* berkinerja tinggi sehingga user dapat mengirimkan gambar dan objek aplikasi dengan di mana saja. Layanan ini dapat digunakan untuk mengelola kerentanan perangkat lunak, mengoptimalkan beban kerja penerapan, dan mengelola kebijakan siklus hidup gambar [11].

## **8. Amazon API Gateway**

Amazon API Gateway memudahkan untuk membangun, menerbitkan, mengelola, memantau, dan mengamankan API dalam skala apa pun. API Gateway menangani semua pekerjaan yang terlibat dalam menerima dan memproses hingga ratusan ribu panggilan API bersamaan, termasuk manajemen lalu lintas, dukungan

CORS, otorisasi dan kontrol akses, perutean, pemantauan, dan pembuatan versi API. Dengan Amazon API Gateway, Anda dapat mengembangkan API secara efisien, memantau metrik kinerja dengan mudah, menskalakan kinerja dan biaya, mengotorisasi secara fleksibel, dan membangun RESTful API menggunakan HTTP atau REST API [12].

## **9. Amazon Elastic Load Balancing (ELB)**

Amazon Elastic Load Balancing (ELB) adalah layanan yang mendistribusikan lalu lintas jaringan masuk ke beberapa tujuan dan perangkat virtual dalam satu atau beberapa Availability Zone, seperti Amazon EC2 *instances*, kontainer, alamat IP, atau fungsi Lambda untuk memastikan ketersediaan dan skalabilitas tinggi aplikasi. Amazon ELB dapat digunakan untuk memodernisasi aplikasi dan kontainer, meningkatkan skalabilitas jaringan *cloud hybrid*, dan mempertahankan peralatan jaringan yang ada [13].

## **10. Amazon EC2 Auto Scaling**

Aplikasi *front-end* yang diterapkan oleh AWS EC2 Auto Scaling, dapat secara otomatis mengelola detail peningkatan kapasitas, penyeimbangan muatan, penskalaan otomatis, dan pemantauan kesehatan aplikasi. Auto Scaling Amazon EC2 adalah layanan yang mengatasi batasan server dalam memproses beban aplikasi, yang dapat menyebabkan *latency* dan kesalahan. Penskalaan otomatis memastikan bahwa Anda memiliki cukup *instance* Amazon EC2 untuk menjalankan aplikasi Anda. Kumpulan skala otomatis berisi kumpulan *instance* EC2 dan dapat disesuaikan dengan menentukan jumlah *instance* EC2 minimum dan maksimum [14].

## **11. Amazon ElasticSearch Services**

Amazon ElasticSearch Service memungkinkan pengguna untuk menerapkan, mengoperasikan, dan menskalakan kluster ElasticSearch di cloud. ElasticSearch adalah mesin pencari dan analisis terdistribusi sumber terbuka yang banyak digunakan untuk berbagai kasus penggunaan seperti analisis log, pencarian teks lengkap, dan pemantauan aplikasi waktu nyata. Layanan ini menangani tugas-tugas seperti penerapan kluster, penambalan perangkat lunak, pencadangan dan pemantauan otomatis, sehingga Anda dapat fokus menggunakan Elasticsearch untuk aplikasi dan analisis data Anda. Layanan ElasticSearch menyediakan kemampuan pencarian yang cepat dan dapat diskalakan untuk mencari data dalam jumlah besar dengan cepat [15].

## **12. Amazon ElastiCache**

Amazon ElastiCache adalah layanan web untuk mengatur, menjalankan dan menskalakan cache dalam memori dengan lancar di cloud, dirancang untuk membantu dalam mengoptimalkan kinerja aplikasi web dengan menyediakan

penyimpanan cache yang dapat diakses dengan cepat. Dengan ElastiCache dapat mengurangi beban pada basis data backend, kemudian dapat menyederhanakan akses data ketika melakukan cache pada data yang sering diakses untuk mempersingkat pengambilan data, serta digunakan sebagai penyimpanan set data dalam memori. Secara keseluruhan, Amazon ElastiCache adalah layanan yang berguna dalam meningkatkan performa aplikasi dengan memanfaatkan cache dalam memori [16].

### **13. Amazon DynamoDB**

Amazon DynamoDB adalah layanan database NoSQL yang dikelola sepenuhnya oleh AWS. Layanan tersebut menawarkan kinerja yang cepat dan dapat diprediksi serta memiliki skalabilitas yang lancar. DynamoDB memberikan user kemampuan untuk membuat tabel database yang dapat menyimpan dan mengambil sejumlah data dan melayani setiap tingkat lalu lintas permintaan. DynamoDB mengurangi beban administratif dalam menjalankan dan menskalakan database terdistribusi, sehingga user tidak perlu mengkhawatirkan hal-hal seperti perangkat keras, konfigurasi, replikasi, dan lain sebagainya [17].

### **14. Amazon ECS Fargate**

AWS Fargate adalah teknologi yang dapat digunakan dengan Amazon ECS untuk menjalankan kontainer tanpa harus mengelola server atau cluster instance Amazon EC2. Pengguna tidak perlu lagi menyediakan, mengonfigurasi, atau menskalakan cluster mesin virtual untuk menjalankan kontainer sehingga pengguna tidak perlu mengatur atau mengelola kapasitas server secara manual. Dalam pemakaian AWS Fargate, pengguna hanya memberikan tugas dan pengaturan untuk menjalankan kontainer menggunakan ECS [18].

### **15. AWS Site-to-Site VPN**

AWS Site-to-Site VPN adalah layanan yang dapat membangun koneksi aman dan terenkripsi antar jaringan lokal dan AWS, sehingga dapat membuat komunikasi yang aman dan terlindungi antar kedua lingkungan tersebut. Pengguna dapat mengaktifkan akses ke jaringan jarak jauh dari VPC, untuk memulainya pengguna harus membuat koneksi AWS Site-to-Site VPN Site-to-Site VPN, setelah koneksi VPN dibangun, pengguna dapat mengonfigurasi perutean untuk melewati lalu lintas melalui koneksi VPN tersebut. Secara keseluruhan AWS Site-to-Site VPN dapat membangun koneksi aman antara jaringan lokal dan AWS [19].

### **16. Customer Gateway**

Customer Gateway adalah bagian dari AWS Site-to-Site VPN yang bertanggung jawab untuk membuat koneksi Site-to-Site VPN antara jaringan lokal



dan Virtual Private Cloud (VPC) di AWS. Customer gateway juga bertanggung jawab dalam mengenkripsi lalu lintas yang melewati koneksi VPN antara jaringan lokal dan AWS. Pada saat membuat customer gateway pengguna perlu memberikan informasi tentang perangkat ke AWS, seperti Alamat IP publik, protokol enkripsi, dan lainnya untuk menonfigurasi koneksi VPN dan menyinkronkannya dengan konfigurasi gateway pelanggan [21].

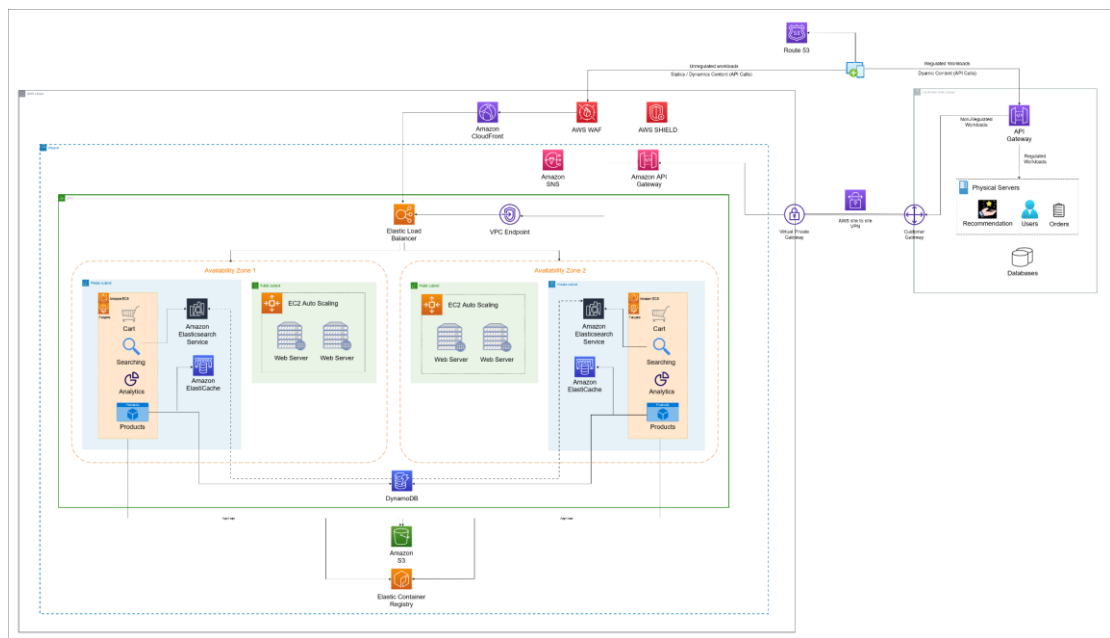
## **17. Virtual Private Gateway**

Virtual Private Gateway adalah komponen yang digunakan dalam layanan AWS untuk menghubungkan Virtual Private Cloud (VPC) di AWS dengan jaringan lokal melalui koneksi Site-to-Site VPN. Dengan adanya Virtual Private Gateway, organisasi dapat menghubungkan dan mengintegrasikan sumber daya cloud yang ada di VPC dengan infrastruktur yang telah mereka miliki sebelumnya di jaringan lokal mereka. Hal ini memungkinkan terjadinya komunikasi yang aman dan terenkripsi antara jaringan lokal dan VPC di lingkungan cloud AWS, sehingga organisasi dapat memperluas kapabilitas dan skala infrastruktur mereka ke dalam cloud dengan tetap memanfaatkan sumber daya yang sudah ada. Dengan demikian, Virtual Private Gateway menjadi solusi penting dalam menghubungkan dan menggabungkan lingkungan cloud dengan infrastruktur lokal yang ada [21].

## **18. VPC Endpoint**

VPC Endpoint adalah komponen dalam Amazon VPC yang dirancang untuk memfasilitasi komunikasi yang aman dan efisien antara instance di dalam Amazon VPC dan layanan AWS tanpa melewati internet publik. VPC Endpoint juga merupakan perangkat virtual dan merupakan komponen Amazon VPC yang diskalakan secara horizontal, redundant, dan tersedia yang memungkinkan komunikasi antara instance di Amazon VPC dan layanan tanpa membebankan risiko ketersediaan atau batasan bandwidth pada lalu lintas jaringan [22].

## B. Arsitektur



Link gambar: <https://drive.google.com/file/d/1ngjsNdVVaRqAM-DVE5IVG6ae3HwLdFfA/view?usp=sharing>

Pada arsitektur diatas, pengguna dapat mengakses aplikasi e-commerce melalui perangkat yang terhubung ke Route 53, yang akan merutekan permintaan pengguna ke sumber daya yang sesuai. Saat permintaan termasuk unregulated workloads, maka permintaan tersebut akan dialirkan menuju layanan AWS Cloud. Sedangkan jika permintaan termasuk regulated workloads, maka akan dialirkan ke on-premise Customer Data Center.

### 1. Layanan AWS Cloud

Lalu lintas yang mengalir ke layanan AWS Cloud akan melewati AWS Web Application Firewall (WAF) dan AWS Shield. AWS WAF membantu melindungi aplikasi dari serangan web umum, sedangkan AWS Shield akan memberikan perlindungan dari serangan DDoS. Kemudian, lalu lintas tersebut mencapai Amazon CloudFront yang akan menyimpan konten dalam cache di edge locations agar pengiriman konten ke pengguna akhir bisa lebih cepat.

Dari CloudFront, lalu lintas akan diteruskan ke Elastic Load Balancing (ELB) yang terletak di dalam Virtual Private Cloud (VPC) di suatu region. Pada region tersebut, terdapat Amazon Simple Notification Service (SNS) dengan tujuan agar memungkinkan pengiriman pesan ke pelanggan, serta Amazon API Gateway untuk membangun, menerbitkan, mengelola, memantau, dan mengamankan API dalam skala apa pun. Amazon API Gateway akan mengatur lalu lintas yang masuk dari on-premise Customer Data Center ke layanan di AWS Cloud.

Amazon API Gateway terhubung dengan VPC Endpoint, yang memungkinkan konektivitas secara private antara VPC dan API Gateway tanpa

perlu memaparkan layanan ke internet publik. VPC Endpoint terhubung secara langsung ke Elastic Load Balancer (ELB) untuk menjaga arus lalu lintas yang masuk sehingga dapat didistribusikan dengan aman.

ELB akan mendistribusikan lalu lintas tersebut di beberapa instance dalam 2 availability zones, dengan tujuan untuk memastikan skalabilitas dan ketersediaan layanan yang tinggi. Adanya 2 Availability Zones dapat menjamin redundansi dan toleransi kesalahan.

Dalam tiap availability zone, terdapat public dan private subnet. Pada public subnet terdapat Amazon EC2 Auto Scaling Group, yang didalamnya berisi komponen-komponen web server (yaitu komponen-komponen dari aplikasi frontend e-commerce) yang berjalan pada instans. EC2 Autoscaling akan membantu memastikan bahwa jumlah instans dapat ditambahkan atau dikurangi secara otomatis berdasarkan kebutuhan, serta secara otomatis menangani penskalaan dan pemantauan kesehatan aplikasi.

Private subnet berisi Amazon ECS Fargate yang menjalankan 4 layanan, yaitu "Cart", "Searching", "Analytics", dan "Products". ECS Fargate digunakan untuk penerapan dan pengelolaan layanan tanpa server, sehingga kita tidak perlu mengelola infrastruktur dasar atau menyediakan instans EC2 secara eksplisit untuk keempat layanan tersebut.

Layanan Cart memungkinkan pengguna untuk menambah, menghapus, dan memperbarui item di keranjang mereka saat menelusuri produk. Layanan Searching memungkinkan pengguna untuk mencari produk berdasarkan berbagai kriteria seperti kata kunci, kategori, filter, dan opsi penyortiran. Layanan Analytics akan mengumpulkan dan menganalisis data seperti demografi pelanggan, produk yang terjual, tren pendapatan, dan data lainnya untuk digunakan sebagai wawasan bisnis agar dapat mengoptimalkan platform e-commerce. Sedangkan layanan Products menjalankan tugas-tugas seperti pembuatan, pembaruan, dan penghapusan produk. Layanan produk juga menyimpan informasi produk seperti nama, deskripsi, harga, ketersediaan, gambar, dan detail lainnya.

Data-data yang dimiliki keempat layanan tersebut akan disimpan di Amazon S3 dan Amazon Elastic Container Registry (ECR). Amazon S3 akan menyimpan semua konten-konten katalog, seperti manual, video, serta semua file log dari keempat layanan tersebut dan informasi aliran klik dari Amazon CloudFront. Namun, untuk gambar-gambar produk akan disimpan di Amazon ECR agar gambar-gambar tersebut lebih mudah diakses dan cepat tersedia.

Layanan "Products" menggunakan DynamoDB dan Amazon ElastiCache untuk menyimpan data-datanya. DynamoDB digunakan untuk menyimpan data sesi yang persisten seperti keranjang belanja, serta digunakan sebagai database produk. Hal ini karena DynamoDB tidak memiliki skema, sehingga akan memiliki banyak fleksibilitas ketika menambahkan kategori dan atribut produk baru ke katalog.

Sedangkan Amazon ElastiCache digunakan untuk melakukan caching atau menyimpan data produk yang sering diakses agar permintaan selanjutnya untuk data yang sama dapat diambil secara langsung dari cache, sehingga tidak perlu mengakses DynamoDB setiap saat. Hal ini dapat mengurangi waktu respons dan beban pada database DynamoDB.

Pada DynamoDB dan layanan “Searching”, diberikan layanan tambahan yaitu Amazon ElasticSearch Service. Pada DynamoDB, layanan ElasticSearch Service berfungsi untuk memudahkan pencarian data dengan menggunakan teks. Hal ini karena DynamoDB tidak memberikan kemampuan pencarian data dalam bentuk teks. Sedangkan pada layanan “Searching”, diberikan layanan ElasticSearch Service untuk memudahkan pengguna melakukan pencarian produk atau layanan lainnya dengan menggunakan teks.

## **2. On-Premise Data Center**

Lalu lintas yang mengalir ke layanan on-premise Customer Data Center akan terhubung dengan API Gateway, yang akan menangani perutean lalu lintas agar dapat mendistribusikan permintaan masuk ke Physical Server atau ke AWS Cloud Server.

Permintaan yang masuk ke AWS Cloud akan melalui AWS site-to-site VPN, yang terdiri dari Customer Gateway dan Virtual Private Gateway. Hal ini bertujuan untuk membangun saluran komunikasi yang aman dan terenkripsi untuk lalu lintas yang masuk dari on-premise Customer Data Center ke AWS Cloud. Di dalam AWS Cloud, Virtual Private Gateway terhubung ke Amazon API Gateway, yang akan mengatur lalu lintas yang masuk dari on-premise Customer Data Center ke layanan-layanan yang ada di AWS Cloud.

Permintaan yang masuk ke Physical Server akan tetap berada di on-premise data center. Dalam Physical Server terdapat 3 layanan, yaitu “Recommendations”, “Users”, dan “Orders”. Layanan-layanan tersebut melakukan berbagai tugas sesuai dengan fungsinya masing-masing, seperti membuat rekomendasi yang dipersonalisasi untuk pengguna, mengelola akun dan profil pengguna, serta menangani manajemen pesanan.

Di bawah fungsi Physical Server terdapat Database, yang menunjukkan sistem database dasar yang digunakan oleh keempat layanan tersebut. Database akan menyimpan data-data yang dihasilkan atau dikelola layanan-layanan tersebut. Contohnya, layanan Recommendations akan mengambil data preferensi pengguna dan riwayat pembelian sebelumnya dari database untuk menghasilkan rekomendasi yang dipersonalisasi. Layanan Users dapat menyimpan dan mengambil data informasi akun pengguna. Sedangkan layanan Orders akan mengelola data pesanan dan transaksi pengguna.

## Referensi

- [1] Dicoding. (2020, July 1). Apa Itu Server? Dicoding Blog. Retrieved from <https://www.dicoding.com/blog/apa-itu-server/>
- [2] Forbes Tech Council. (2023, March 27). Cloud Versus On-Premises: Advantages and Disadvantages of Both Models. Forbes. Retrieved from <https://www.forbes.com/sites/forbestechcouncil/2023/03/27/cloud-versus-on-premises-advantages-and-disadvantages-of-both-models/?sh=11dd1cd45df6>
- [3] Amazon Web Services. Six Advantages of Cloud Computing. Retrieved from <https://docs.aws.amazon.com/whitepapers/latest/aws-overview/six-advantages-of-cloud-computing.html>
- [4] TechTarget. Amazon Web Services (AWS). Retrieved from <https://www.techtarget.com/searchaws/definition/Amazon-Web-Services>
- [5] Amazon Web Services. AWS WAF – Web Application Firewall. Retrieved from <https://aws.amazon.com/waf/?ref=wellarchitected>
- [6] Amazon Web Services. AWS Shield. Retrieved from <https://aws.amazon.com/shield/>
- [7] Amazon Web Services. Amazon CloudFront - Content Delivery Network (CDN) Service. Retrieved from <https://aws.amazon.com/cloudfront/>
- [8] Amazon Web Services. Amazon Simple Storage Service (S3) documentation. Retrieved from <https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html>
- [9] Amazon Web Services. Amazon Simple Notification Service (Amazon SNS). Retrieved from <https://aws.amazon.com/sns/>
- [10] Avinetworks. AWS Route 53. Avi Networks Glossary. Retrieved from <https://avinetworks.com/glossary/aws-route-53/>
- [11] Amazon Web Services. Amazon Elastic Container Registry (ECR). Retrieved from <https://aws.amazon.com/ecr/>
- [12] Amazon Web Services. Amazon API Gateway. Retrieved from <https://aws.amazon.com/api-gateway/>
- [13] Amazon Web Services. Elastic Load Balancing. Retrieved from <https://aws.amazon.com/elasticloadbalancing/>
- [14] Jawad, A. (2018, July 24). Amazon EC2 Auto Scaling. Medium. Retrieved from <https://medium.com/@jawad846/amazon-ec2-auto-scaling-884ea50d2d>
- [15] Amazon Web Services. What is Amazon Elasticsearch Service? Retrieved from <https://aws.amazon.com/what-is/elasticsearch/>
- [16] Amazon Web Services. What is Amazon ElastiCache? Retrieved from <https://aws.amazon.com/id/pm/elasticache>
- [17] Amazon Web Services. Amazon DynamoDB - Fast and Flexible NoSQL Database Service. Retrieved from <https://www.amazonaws.cn/en/dynamodb/>
- [18] Amazon Web Services. What is AWS Fargate? Retrieved from <https://docs.aws.amazon.com/AmazonECS/latest/userguide/what-is-fargate.html>
- [19] Amazon Web Services. What is AWS Site-to-Site VPN? Retrieved from [https://docs.aws.amazon.com/id\\_id/vpn/latest/s2svpn/VPC\\_VPN.html](https://docs.aws.amazon.com/id_id/vpn/latest/s2svpn/VPC_VPN.html)
- [20] Amazon Web Services. How AWS Site-to-Site VPN works. Retrieved from [https://docs.aws.amazon.com/vpn/latest/s2svpn/how\\_it\\_works.html](https://docs.aws.amazon.com/vpn/latest/s2svpn/how_it_works.html)
- [21] Amazon Web Services. Virtual private gateway associations. Retrieved from <https://docs.aws.amazon.com/directconnect/latest/UserGuide/virtualgateways.html>
- [22] Amazon Web Services. What are VPC endpoints? Retrieved from <https://docs.aws.amazon.com/whitepapers/latest/aws-privatelink/what-are-vpc-endpoints.html>