**Customer Gateway** là một thiết bị vật lý hoặc ứng dụng phần mềm nằm ở phía khách hàng trong kết nối VPN site-to-site. Nó đóng vai trò trung gian giữa mạng của khách hàng, Internet và VPN, đảm bảo giao tiếp an toàn bằng cách:

1. **Quản lý lưu lượng**: Điều hướng dữ liệu giữa mạng khách hàng và tài nguyên trên cloud.
2. **Mã hóa và giải mã**: Bảo vệ dữ liệu trong quá trình truyền tải.
3. **Duy trì kết nối VPN**: Đảm bảo kết nối ổn định và bảo mật.

Customer Gateway thường được sử dụng trong các bài toán hybrid cloud, nơi cần kết nối giữa hạ tầng tại chỗ và tài nguyên trên AWS.

======================================================================

Dung lượng ổ đĩa tạm thời (ephemeral disk capacity) trong thư mục **"/tmp"** của AWS Lambda có thể được cấu hình từ **512 MB đến tối đa 10,240 MB** cho mỗi lần thực thi (invocation). Dữ liệu trong **"/tmp"** được mã hóa khi lưu trữ và sẽ bị xóa sau khi môi trường thực thi kết thúc.

======================================================================

Khi bạn tạo một **custom security group** mới trong AWS, các quy tắc mặc định sẽ được áp dụng như sau:

1. **Không có quy tắc inbound**: Tất cả lưu lượng inbound (vào) sẽ bị từ chối theo mặc định, trừ khi bạn thêm quy tắc inbound cụ thể.
2. **Cho phép tất cả lưu lượng outbound**: Mặc định, tất cả lưu lượng outbound (ra) đều được phép, trừ khi bạn chỉnh sửa hoặc giới hạn quy tắc outbound.

======================================================================

Khi bạn tạo một **Network ACL (NACL)** mới trong AWS, các quy tắc mặc định sẽ được áp dụng như sau:

1. **Inbound Rules**: Tất cả lưu lượng inbound (vào) bị **từ chối** theo mặc định, trừ khi bạn thêm các quy tắc cụ thể để cho phép.
2. **Outbound Rules**: Tất cả lưu lượng outbound (ra) cũng bị **từ chối** theo mặc định, trừ khi bạn cấu hình quy tắc để cho phép.

======================================================================

Khi bạn không thể kết nối tới **EC2 instance** trong một **subnet** của VPC từ Internet, các yếu tố cấu hình chính cần kiểm tra đầu tiên bao gồm:

1. **Security Group**: Kiểm tra security group gắn với EC2 instance, đảm bảo rằng nó có quy tắc inbound cho phép lưu lượng từ địa chỉ IP hoặc port yêu cầu (ví dụ: SSH với port 22).
2. **Network ACL (NACL)**: Đảm bảo rằng NACL của subnet cho phép cả lưu lượng **inbound** và **outbound**. Vì NACL là stateless, cả hai loại lưu lượng đều phải được cấu hình rõ ràng.
3. **Route Table**: Kiểm tra route table của subnet, chắc chắn rằng nó có route chỉ đến **Internet Gateway (IGW)** để lưu lượng có thể đi ra Internet.

======================================================================

**Splunk** là một phần mềm mạnh mẽ dùng để thu thập, tìm kiếm, giám sát và phân tích dữ liệu lớn (big data) được tạo ra bởi máy móc, chẳng hạn như log từ ứng dụng, hệ thống, hoặc thiết bị mạng

### **Các tính năng chính của Splunk:**

1. **Phân tích log**: Hỗ trợ nhiều định dạng log như Syslog, CSV, Apache-log, giúp quản trị viên dễ dàng tìm kiếm và xử lý dữ liệu.
2. **Giám sát thời gian thực**: Cập nhật liên tục khi có thay đổi, cung cấp cảnh báo ngay lập tức khi phát hiện vấn đề.
3. **Tìm kiếm thông minh**: Sử dụng ngôn ngữ tìm kiếm riêng (Splunk Language) để truy xuất dữ liệu nhanh chóng.
4. **Hiển thị trực quan**: Tạo dashboard và báo cáo giúp người dùng dễ dàng hình dung tình trạng hệ thống.
5. **Tích hợp API**: Hỗ trợ phát triển ứng dụng tùy chỉnh trên nền tảng Splunk.

**CloudWatch Logs** là một dịch vụ trong AWS được thiết kế để thu thập, theo dõi và lưu trữ log từ các nguồn khác nhau, như ứng dụng, hệ điều hành, hoặc dịch vụ AWS. Nó giúp bạn giám sát và phân tích log một cách hiệu quả.

### **Tính năng của CloudWatch Logs:**

1. **Thu thập log**: Tự động lưu log từ các dịch vụ AWS (như Lambda, EC2, RDS) hoặc từ ứng dụng của bạn.
2. **Giám sát log**: Tạo **metric filter** để phát hiện các mẫu log cụ thể (ví dụ: lỗi hoặc sự kiện bất thường).
3. **Lưu trữ log**: Lưu trữ log an toàn, có thể tùy chỉnh thời gian giữ log (retention).
4. **Tìm kiếm và phân tích**: Sử dụng giao diện hoặc API để tìm kiếm và phân tích log theo yêu cầu.
5. **Tích hợp**: Kết hợp với các dịch vụ khác như **CloudWatch Alarms** để cảnh báo khi phát hiện vấn đề.

======================================================================

**Amazon Elastic Transcoder** là một dịch vụ **chuyển mã đa phương tiện (media transcoding)** được AWS quản lý hoàn toàn, giúp chuyển đổi video và âm thanh sang các định dạng khác nhau phù hợp với nhiều thiết bị và điều kiện mạng.

### **Tính năng chính:**

1. **Chuyển đổi định dạng (Transcoding)**: Chuyển đổi video/audio giữa các định dạng, ví dụ:
   * MP4 → HLS.
   * Thay đổi độ phân giải: 4K → 1080p.
2. **Tối ưu hóa chất lượng**: Cân bằng giữa chất lượng và kích thước file thông qua điều chỉnh bitrate.
3. **Hỗ trợ DRM (Digital Rights Management)**: Mã hóa nội dung để bảo vệ bản quyền và ngăn sao chép.
4. **Tích hợp với S3 & CloudFront**:
   * Lưu trữ input/output trên Amazon S3.
   * Phân phối nội dung qua CloudFront (CDN).

### **Trường hợp sử dụng:**

1. **Streaming đa nền tảng**: Tạo các phiên bản video phù hợp với iOS, Android, Smart TV.
2. **User-generated content**: Chuyển đổi video được tải lên từ người dùng (ví dụ: mạng xã hội).
3. **Lưu trữ file media**: Chuyển đổi các video cũ sang định dạng mới (FLV → MP4).

======================================================================

Trong AWS Identity and Access Management (IAM), hai loại **principals** có thể đảm nhận một role là:

1. **IAM Users**: Người dùng trong cùng tài khoản AWS hoặc tài khoản khác (cross-account) có thể assume một role nếu được cấp quyền thông qua chính sách Trust.
2. **AWS Services (Service Principals)**: Các dịch vụ AWS, ví dụ như EC2, Lambda, hoặc ECS, có thể assume một role để thực hiện tác vụ thay mặt bạn.

======================================================================

**DynamoDB is more cost effective for read heavy workloads** DynamoDB thường hiệu quả về chi phí đối với các workload **đọc nhiều (read-heavy)**, đặc biệt nếu bạn tận dụng các tính năng như DynamoDB Accelerator (DAX) để giảm tải chi phí đọc.

**Priced based on provisioned throughput (read/write) regardless of whether you use it or not** Nếu bạn chọn **Provisioned Capacity Mode**, DynamoDB sẽ tính phí dựa trên dung lượng đọc/ghi đã thiết lập trước (RCUs/WCUs), bất kể bạn có sử dụng hết hay không.

======================================================================

### **Bảng so sánh CloudTrail và CloudWatch Logs**

| **Tiêu chí** | **CloudTrail** | **CloudWatch Logs** |
| --- | --- | --- |
| **Mục đích chính** | Theo dõi hoạt động API và sự kiện trong tài khoản AWS | Thu thập, lưu trữ, phân tích log từ EC2 và máy chủ |
| **Nguồn dữ liệu** | Các hoạt động AWS API và CLI | Log từ EC2, Lambda, ứng dụng, hệ thống on-premises |
| **Lưu trữ** | Lưu trữ lịch sử sự kiện API | Lưu trữ log lâu dài với thời gian giữ tùy chỉnh |
| **Tìm kiếm và phân tích** | Giám sát sự kiện audit và tuân thủ bảo mật | Tìm kiếm log, phân tích mẫu với metric filter |
| **Tích hợp với Alarms** | Không hỗ trợ | Tích hợp với CloudWatch Alarms |
| **Phân tích thời gian thực** | Không có | Hỗ trợ thời gian thực (giám sát hoạt động log) |

======================================================================

**Chef** và **Puppet** là hai công cụ phổ biến trong lĩnh vực **quản lý cấu hình (Configuration Management)**, giúp tự động hóa việc triển khai, quản lý, và duy trì cơ sở hạ tầng IT. Chúng thường được sử dụng trong các môi trường DevOps.

### **Chef**

* **Cách hoạt động**: Chef sử dụng ngôn ngữ lập trình Ruby để định nghĩa "công thức" (**recipes**) và "sách nấu ăn" (**cookbooks**) nhằm quản lý các cấu hình hạ tầng.
* **Ứng dụng**:
  + Tự động cài đặt và cấu hình phần mềm trên máy chủ.
  + Quản lý các bản cập nhật và kiểm tra tính nhất quán của hệ thống.
* **Ưu điểm**: Mạnh mẽ, linh hoạt, và có cộng đồng người dùng lớn.
* **Nhược điểm**: Đường cong học tập dốc hơn, yêu cầu kiến thức về Ruby.

### **Puppet**

* **Cách hoạt động**: Puppet sử dụng ngôn ngữ khai báo (**declarative language**) hoặc ngôn ngữ riêng của Puppet (Puppet DSL) để định nghĩa trạng thái mong muốn của hệ thống.
* **Ứng dụng**:
  + Đảm bảo hệ thống đạt trạng thái mong muốn bằng cách tự động điều chỉnh khi có sai lệch.
  + Quản lý hàng ngàn máy chủ trên môi trường phức tạp.
* **Ưu điểm**: Dễ học hơn, tốt cho việc duy trì trạng thái nhất quán.
* **Nhược điểm**: Hạn chế linh hoạt hơn so với Chef.

### **Điểm chung**

* Tự động hóa cấu hình hạ tầng, giảm thiểu sai sót do thao tác thủ công.
* Hỗ trợ triển khai ở quy mô lớn trong môi trường hybrid hoặc đám mây.

======================================================================

**Pre-signed URL** trong AWS là một đường dẫn tạm thời, được tạo ra để cấp quyền truy cập giới hạn thời gian vào một đối tượng trong Amazon S3 mà không cần bật public access.

======================================================================

| **S3 Storage Class** | **Durability** | **Availability** |
| --- | --- | --- |
| **Standard** | 99.999999999% (11 9's) | 99.99% |
| **Intelligent-Tiering** | 99.999999999% (11 9's) | 99.9% |
| **Standard-IA (Infrequent Access)** | 99.999999999% (11 9's) | 99.9% |
| **One Zone-IA** | 99.999999999% (11 9's) | 99.5% |
| **Glacier** | 99.999999999% (11 9's) | N/A (retrieve via restore) |
| **Glacier Deep Archive** | 99.999999999% (11 9's) | N/A (retrieve via restore) |

======================================================================

**Giới hạn mặc định (Default Limit)**: Mỗi bảng DynamoDB được cấp **10,000 Write Capacity Units** mặc định cho phép ghi vào bảng. Đây là số lượng throughput tối đa mà bảng có thể hỗ trợ khi bạn chưa yêu cầu điều chỉnh giới hạn.

**Giới hạn tối đa (Maximum Supported Limit)**: Tối đa mà AWS hỗ trợ cho một bảng là **10,000 Write Capacity Units** mà không yêu cầu thay đổi giới hạn. Nếu bạn cần thêm throughput, bạn có thể yêu cầu **tăng giới hạn (Service Limit Increase)** từ AWS, và mức tối đa có thể đạt **80,000 Write Capacity Units** đối với một bảng duy nhất trong một khu vực.

**Giới hạn per AWS account**: Trong một tài khoản AWS, có giới hạn tổng thể về số lượng Write Capacity Units cho tất cả các bảng DynamoDB trong tài khoản đó. Giới hạn này thường là **40,000 Write Capacity Units** cho mỗi tài khoản. Nếu bạn có nhiều bảng và cần thêm throughput, bạn có thể yêu cầu tăng giới hạn này từ AWS.

======================================================================

* Các ổ đĩa EBS được sao chép trong Vùng khả dụng (AZ) của chúng để bảo vệ bạn khỏi lỗi thành phần

======================================================================

### **Amazon Elastic File System (EFS)**

* **Đặc điểm chính**:
  + EFS là một hệ thống file lưu trữ có khả năng mở rộng tự động và cung cấp throughput, IOPS cao.
  + EFS hỗ trợ chia sẻ file qua các **EC2 instances** trên **nhiều vùng sẵn sàng (AZs)** trong cùng một khu vực.
  + Cung cấp độ trễ thấp và khả năng nhất quán khi truy cập.
* **Phù hợp trong bài toán này**:
  + Đáp ứng yêu cầu mount lưu trữ vào EC2 instances trên nhiều AZ.
  + Đảm bảo khả năng mở rộng tự động và hiệu năng cao cho dữ liệu tải lên từ CMS.

======================================================================

**Amazon Elastic Container Registry (ECR)** là một dịch vụ được quản lý hoàn toàn của AWS, dùng để **lưu trữ, quản lý và phân phối container images** (hình ảnh container Docker). Nó giúp bạn dễ dàng triển khai và quản lý các container mà không phải xây dựng lại hình ảnh từ đầu mỗi lần.

======================================================================

Amazon Route 53 là dịch vụ DNS được quản lý hoàn toàn, với các tính năng nổi bật sau:

* **Quản lý DNS**: Phân giải tên miền thành địa chỉ IP. Hỗ trợ cả Public DNS và Private DNS cho VPC.
* **Định tuyến lưu lượng**: Nhiều loại như Weighted, Latency-based, Geolocation, và Failover routing.
* **Đăng ký tên miền**: Cho phép đăng ký và quản lý tên miền trực tiếp.
* **Kiểm tra sức khỏe**: Giám sát trạng thái endpoint và định tuyến lưu lượng đến endpoint khỏe mạnh.
* **Tích hợp AWS**: Kết nối sâu với các dịch vụ như CloudFront, S3, và Elastic Load Balancing.
* **Khả năng mở rộng**: Tự động xử lý khối lượng lớn truy vấn DNS mà không ảnh hưởng hiệu năng.

Route 53 đảm bảo tính bảo mật, độ sẵn sàng cao và hiệu suất vượt trội, phù hợp cho cả ứng dụng nhỏ lẫn toàn cầu.

không thể tự động đăng ký các phiên bản EC2 với các vùng lưu trữ riêng tư

Có thể được sử dụng để định tuyến lưu lượng truy cập Internet cho các tên miền đã đăng ký với một công ty đăng ký tên miền khác

======================================================================

### **1. Private IP Addresses trong EC2**

**Đặc điểm chính:**

* **Không thể truy cập từ Internet: Private IP chỉ được sử dụng cho giao tiếp nội bộ trong VPC, và không thể truy cập từ bên ngoài Internet.**
* **VPC: Private IP được cấp từ dải IP của subnet trong VPC và giữ nguyên khi instance bị dừng và khởi động lại (vẫn thuộc về network interface).**
* **Secondary Private IPs: Có thể gán thêm các secondary private IPs cho instance, ví dụ cho mục đích multi-homed instances (chạy nhiều dịch vụ với các IP khác nhau trên cùng một instance).**

**EC2-Classic (cũ, đã deprecated):**

* **Trong EC2-Classic, private IP bị release khi instance bị terminated, do đó nó không giữ lại khi instance khởi động lại.**

### **2. Public IP Addresses trong EC2**

**Đặc điểm chính:**

* **Tính động: Public IP là địa chỉ IP có thể thay đổi khi instance được stop/start (trừ khi bạn sử dụng Elastic IP).**
* **Released khi instance terminated: Khi instance bị terminated, public IP sẽ bị giải phóng.**
* **EC2-Classic vs VPC:**
  + **EC2-Classic: Tự động gán public IP cho instance và không thể tắt.**
  + **VPC: Tùy chọn gán public IP thông qua thuộc tính auto-assign của subnet hoặc cấu hình riêng của instance.**

### **Mẹo thi AWS SAA-C03**

1. **Private IP trong VPC:**
   * **Luôn giữ nguyên khi stop/start.**
   * **Muốn thay đổi, phải dùng secondary IP hoặc tạo ENI mới.**
2. **Public IP:**
   * **Dynamic: Thay đổi khi stop/start.**
   * **Nếu muốn cố định: Dùng Elastic IP.**
3. **EC2-Classic:**
   * **Đã deprecated: Ưu tiên chọn các đáp án liên quan đến VPC.**

### **Ví dụ triển khai thực tế:**

1. **Web Server trong VPC:**
   * **Gán Elastic IP cho instance để giữ Public IP cố định (tránh tình trạng Public IP thay đổi khi instance stop/start).**
   * **Dùng secondary private IP khi cần chạy nhiều website trên cùng một instance (multi-website hosting).**

======================================================================

* API Gateway có thể mở rộng đến bất kỳ mức lưu lượng nào được API nhận được
* Điều tiết được sử dụng để bảo vệ các phiên bản back-end khỏi các đợt tăng đột biến lưu lượng truy cập
* Lambda sử dụng tỷ lệ liên tục – tỷ lệ mở rộng chứ không phải tăng lên
* Lambda mở rộng các chức năng thực thi đồng thời lên đến giới hạn mặc định của bạn (1000)

======================================================================

1. **VPC Peering Connection**:
   * **Cách hoạt động**: Thiết lập kết nối ngang hàng giữa hai VPC trong cùng hoặc khác tài khoản AWS, cho phép giao tiếp thông qua địa chỉ IP riêng.
   * **Ưu điểm**: Đơn giản, bảo mật cao, không yêu cầu sử dụng internet công cộng.
   * **Ứng dụng**: Là một trong các phương án phù hợp để kết nối API qua địa chỉ IP riêng.
2. **AWS Direct Connect**:
   * **Cách hoạt động**: Cung cấp kết nối chuyên dụng từ mạng doanh nghiệp đến AWS.
   * **Hạn chế**: Direct Connect chủ yếu được sử dụng để kết nối từ on-premises tới AWS, không phù hợp cho trường hợp giữa hai VPC.
3. **ClassicLink**:
   * **Cách hoạt động**: Cho phép kết nối các instance trong Classic EC2 tới VPC.
   * **Hạn chế**: ClassicLink chỉ áp dụng cho hệ thống cũ (EC2-Classic) và không phù hợp cho tình huống này.
4. **PrivateLink**:
   * **Cách hoạt động**: Tạo giao diện mạng (endpoint) trong VPC của client, cho phép giao tiếp với dịch vụ trong VPC khác thông qua **AWS PrivateLink**.
   * **Ưu điểm**: Tăng bảo mật, dễ cấu hình và hoạt động hoàn toàn qua mạng riêng.
   * **Ứng dụng**: Là một trong các phương án phù hợp để kết nối API.
5. **AWS Resource Access Manager (RAM)**:
   * **Cách hoạt động**: Cho phép chia sẻ tài nguyên như Subnet hoặc Transit Gateway giữa các tài khoản.
   * **Hạn chế**: RAM chỉ hỗ trợ chia sẻ tài nguyên, không phải công cụ trực tiếp để kết nối API.

======================================================================

### **1. AWS Global Accelerator**

* **Cách hoạt động**:
  + AWS Global Accelerator sử dụng mạng lưới toàn cầu của AWS để định tuyến lưu lượng đến endpoint gần nhất (chẳng hạn EC2, ALB/NLB hoặc Elastic IP) dựa trên định tuyến tối ưu.
  + Nó cung cấp **địa chỉ IP tĩnh** toàn cầu và sử dụng bất kỳ endpoint nào trong các vùng AWS.
* **Ưu điểm**:
  + Tối ưu hóa độ trễ với mạng AWS backbone.
  + Khả năng xử lý lỗi tự động, định tuyến lưu lượng đến endpoint khỏe mạnh nhất.
* **Khi nào dùng**:
  + Khi ứng dụng yêu cầu IP tĩnh và muốn giảm độ trễ toàn cầu.

### **2. AWS Direct Connect**

* **Cách hoạt động**:
  + Direct Connect thiết lập kết nối chuyên dụng giữa mạng on-premises và AWS.
* **Hạn chế**:
  + Direct Connect không hữu ích trong trường hợp này vì vấn đề ở đây là cải thiện truy cập qua internet, không phải kết nối từ on-premises đến AWS.

### **3. Amazon CloudFront**

* **Cách hoạt động**:
  + CloudFront là một dịch vụ **CDN** (Content Delivery Network), lưu trữ nội dung tĩnh và cung cấp từ các edge location gần người dùng.
* **Ưu điểm**:
  + Cải thiện độ trễ cho nội dung tĩnh (images, videos, etc.).
* **Hạn chế**:
  + Không phù hợp cho ứng dụng sử dụng địa chỉ IP tĩnh hoặc nội dung động mà cần giảm độ trễ.
* **Khi nào dùng**:
  + Phù hợp nếu ứng dụng chủ yếu dựa trên nội dung tĩnh.

### **4. Amazon Route 53 geoproximity routing**

* **Cách hoạt động**:
  + Route 53 sử dụng **geoproximity routing** để định tuyến lưu lượng dựa trên vị trí của người dùng so với vùng ứng dụng.
* **Hạn chế**:
  + Route 53 chỉ định tuyến DNS mà không trực tiếp cải thiện hiệu suất mạng như Global Accelerator.
* **Khi nào dùng**:
  + Dùng để kiểm soát tuyến đường DNS, nhưng không phải giải pháp hiệu quả nhất trong câu này.

======================================================================

### **1. Amazon RDS for MySQL**

* **Đặc điểm**:
  + Là cơ sở dữ liệu quan hệ được quản lý, phù hợp với các ứng dụng sử dụng cấu trúc dữ liệu truyền thống.
  + Hỗ trợ replication qua Multi-AZ hoặc Read Replicas.
* **Hạn chế**:
  + Không phải cơ sở dữ liệu **in-memory**, không tối ưu cho dữ liệu cần truy cập với tốc độ cực cao.

### **2. Amazon RDS for PostgreSQL**

* **Đặc điểm**:
  + Tương tự MySQL, là cơ sở dữ liệu quan hệ được quản lý.
  + Hỗ trợ replication thông qua Read Replicas hoặc Multi-AZ.
* **Hạn chế**:
  + Không phải cơ sở dữ liệu **in-memory**, nên không tối ưu cho hiệu suất cực cao mà ứng dụng gaming yêu cầu.

### **3. Amazon ElastiCache for Redis**

* **Đặc điểm**:
  + Là cơ sở dữ liệu **in-memory**, lý tưởng cho tốc độ truy cập dữ liệu cao.
  + Hỗ trợ **data replication** (chức năng clustering và read replicas).
  + Tích hợp tốt với các ứng dụng real-time như gaming, analytics, và caching.
* **Ưu điểm**:
  + Đáp ứng yêu cầu về hiệu suất và độ trễ thấp, tối ưu cho việc xử lý dữ liệu thời gian thực.

### **4. Amazon ElastiCache for Memcached**

* **Đặc điểm**:
  + Là cơ sở dữ liệu **in-memory**, nhưng không hỗ trợ data replication hoặc clustering như Redis.
  + Phù hợp cho các ứng dụng yêu cầu cache phân tán đơn giản.
  + does not support high availability or multi-AZ
* **Hạn chế**:
  + Không đáp ứng yêu cầu về **data replication**, vì vậy không phù hợp cho tình huống này.

======================================================================

1. **Security Group**:
   * Là công cụ bảo mật cấp **instance** trong AWS, kiểm soát lưu lượng vào (inbound) và lưu lượng ra (outbound).
   * Security Group **chỉ hỗ trợ allow rules** (cho phép), không hỗ trợ deny rules (từ chối). Vì vậy, không thể sử dụng Security Group để từ chối traffic từ CIDR range.
2. **Network ACL (Access Control List)**:
   * Là công cụ bảo mật cấp **subnet**, kiểm soát lưu lượng vào (inbound) và ra (outbound).
   * **Hỗ trợ cả allow và deny rules**, cho phép chặn traffic từ IP hoặc CIDR range dựa trên thứ tự ưu tiên (rule number).

### **Giải pháp phù hợp:**

* **Network ACL** nên được sử dụng để chặn traffic từ CIDR range. Bạn có thể thêm **deny rule** với số thứ tự thấp hơn các rule khác để đảm bảo nó được xử lý trước.
  + **Inbound table**: Chặn lưu lượng từ CIDR range vào subnet (là giải pháp phù hợp nhất nếu vấn đề xảy ra ở traffic vào).
  + **Outbound table**: Chặn lưu lượng từ subnet ra ngoài (sử dụng trong trường hợp traffic outbound độc hại).

======================================================================

1. **Enable an ALB health check**:
   * **Cách hoạt động**: ALB có khả năng kiểm tra tình trạng của các backend instances.
   * **Hạn chế**: Health check của ALB chỉ kiểm tra trạng thái của instance phía sau ALB, không giúp định tuyến traffic giữa các vùng hoặc tự động hóa failover DNS.
   * **Không phù hợp** trong trường hợp này.
2. **Enable an Amazon Route 53 health check**:
   * **Cách hoạt động**: Route 53 sử dụng health check để kiểm tra tình trạng endpoint (bao gồm ALB) và tự động cập nhật bản ghi DNS khi endpoint chính không khả dụng.
   * **Ưu điểm**:
     + Tự động failover dựa trên kết quả health check.
     + Route 53 có thể kiểm tra các endpoint trong hai vùng AWS và định tuyến traffic tới endpoint khỏe mạnh nhất.
3. **Create a CNAME record on Amazon Route 53 pointing to the ALB endpoint**:
   * **Cách hoạt động**: CNAME record cho phép ánh xạ domain tới endpoint như ALB.
   * **Hạn chế**: CNAME không hỗ trợ tự động failover mà chỉ ánh xạ tên miền tới một endpoint tĩnh.
   * **Không phù hợp** để tự động hóa failover.
4. **Create a latency-based routing policy on Amazon Route 53**:
   * **Cách hoạt động**: Route 53 định tuyến traffic tới endpoint có độ trễ thấp nhất.
   * **Hạn chế**:
     + Chỉ tối ưu hóa về hiệu suất (latency), không tự động failover khi endpoint gặp sự cố.

======================================================================

#### **1. Deploy a NAT instance in the private subnet:**

* **Cách hoạt động**:
  + NAT instance là một EC2 instance được cấu hình để hoạt động như một cổng NAT.
* **Hạn chế**:
  + Yêu cầu quản lý và bảo trì thủ công (ví dụ: cập nhật bản vá, scaling).
  + Không tối ưu về hiệu suất hoặc chi phí so với NAT Gateway.

#### **2. Create an internet gateway and attach it to the VPC. Modify the private subnet route table to direct internet traffic to the internet gateway:**

* **Cách hoạt động**:
  + Internet gateway cho phép kết nối trực tiếp ra internet.
* **Hạn chế**:
  + Private subnet không được phép kết nối trực tiếp với Internet Gateway (phạm lỗi bảo mật).

#### **3. Deploy a NAT gateway in the public subnet. Modify the route table in the private subnet to direct all internet traffic to the NAT gateway:**

* **Cách hoạt động**:
  + NAT Gateway là dịch vụ được quản lý, cho phép các instance trong private subnet gửi yêu cầu ra ngoài internet qua địa chỉ IP công cộng của NAT Gateway.
* **Ưu điểm**:
  + **Bảo mật**: Private subnet không trực tiếp lộ IP công cộng.
  + **Ít quản lý thủ công**: AWS quản lý tính sẵn sàng và mở rộng của NAT Gateway.

#### **4. Create a virtual private gateway and attach it to the VPC. Modify the private subnet route table to direct internet traffic to the virtual private gateway:**

* **Cách hoạt động**:
  + Virtual Private Gateway chủ yếu dùng để kết nối giữa AWS VPC và mạng on-premises qua VPN hoặc Direct Connect.
* **Hạn chế**:
  + Không hỗ trợ truy cập internet.

======================================================================

#### **1. Add an additional VPC peering connection between the two VPCs**

* **Cách hoạt động**: Tạo thêm một kết nối VPC peering giữa Production VPC và Pre-Production VPC.
* **Hạn chế**:
  + AWS không hỗ trợ nhiều kết nối VPC peering giữa hai VPC (một quan hệ peering là đủ).
  + Không giải quyết được vấn đề single point of failure, vì bản thân VPC peering đã có độ tin cậy cao.

#### **2. Add additional VPNs to the Production VPC from a second customer gateway device**

* **Cách hoạt động**: Tạo thêm một hoặc nhiều VPN kết nối từ một thiết bị cổng khách hàng thứ hai trong trung tâm dữ liệu on-premises đến Production VPC.
* **Ưu điểm**:
  + Cung cấp dự phòng cho kết nối VPN hiện tại.
  + Nếu thiết bị đầu tiên gặp lỗi, thiết bị thứ hai sẽ đảm bảo kết nối không bị gián đoạn.
* **Phù hợp**: Giải pháp này trực tiếp loại bỏ single point of failure trong Production VPC.

#### **3. Add a set of VPNs between the Production and Pre-Production VPCs**

* **Cách hoạt động**: Tạo VPN giữa hai VPC.
* **Hạn chế**:
  + Không loại bỏ được single point of failure liên quan đến on-premises.
  + Chỉ thêm kết nối giữa hai VPC, không liên quan đến mục tiêu chính của vấn đề.

#### **4. Add a second virtual private gateway and attach it to the Production VPC**

* **Cách hoạt động**: Gắn thêm virtual private gateway (VGW) vào Production VPC.
* **Hạn chế**:
  + AWS không hỗ trợ gắn nhiều VGW vào cùng một VPC.
  + Không khả thi trong trường hợp này.

======================================================================

#### **1. Use AWS Transfer Acceleration to replicate files to Amazon S3 and enable public access**

* **Cách hoạt động**: AWS Transfer Acceleration tăng tốc truyền dữ liệu vào S3 bằng cách sử dụng mạng lưới edge location của AWS.
* **Hạn chế**:
  + Giải pháp này chủ yếu hỗ trợ truyền dữ liệu nhanh vào S3, không thay thế hoàn toàn dịch vụ lưu trữ on-premises.
  + Công cụ này không cung cấp cơ chế truy cập từ xa hoặc kết nối trực tiếp với hệ thống on-premises.

#### **2. Use Amazon FSx to create an SMB file share. Connect remote clients to the file share over a client VPN**

* **Cách hoạt động**: Amazon FSx cung cấp dịch vụ file system được quản lý hoàn toàn, hỗ trợ giao thức SMB để chia sẻ tệp.
* **Ưu điểm**:
  + Hỗ trợ giao thức SMB cho các ứng dụng hoặc hệ thống cần chia sẻ file.
  + Có thể kết nối từ xa qua VPN.
* **Hạn chế**:
  + Yêu cầu phải thiết lập và quản lý VPN client, tăng thêm phức tạp.
  + Không phải là giải pháp kết nối liền mạch giữa on-premises và lưu trữ đám mây.
* **Không tối ưu** trong trường hợp này.

#### **3. Use AWS DataSync to synchronize data between the on-premises service and Amazon S3**

* **Cách hoạt động**: AWS DataSync tự động đồng bộ hóa dữ liệu giữa on-premises và Amazon S3.
* **Hạn chế**:
  + Giải pháp này chỉ tập trung vào việc đồng bộ dữ liệu, không cung cấp khả năng truy cập dễ dàng cho nhân viên làm việc từ xa.
  + Không thay thế hoàn toàn dịch vụ lưu trữ on-premises.

#### **4. Use AWS Storage Gateway to create a volume gateway to store and transfer files to Amazon S3**

* **Cách hoạt động**: AWS Storage Gateway cung cấp một cầu nối giữa lưu trữ on-premises và lưu trữ đám mây, cho phép dữ liệu được lưu trữ trên S3 và truy cập từ on-premises.
* **Ưu điểm**:
  + Là giải pháp **fully managed**, giảm bớt hoạt động vận hành.
  + Tích hợp dễ dàng với các hệ thống on-premises, đồng thời hỗ trợ lưu trữ dữ liệu trên S3.
  + Phù hợp cho cả làm việc từ xa và sử dụng hệ thống on-premises.

======================================================================

### **Phân biệt Alias Record và CNAME Record trong Amazon Route 53**

### **Alias Record:**

* **Dùng khi nào?**
  + Khi muốn ánh xạ tên miền (domain) đến tài nguyên AWS được quản lý, như:
    - **Amazon S3 bucket** (website endpoint).
    - **CloudFront distribution**.
    - **ELB (Elastic Load Balancer)**.
    - **Route 53 record trong cùng một hosted zone**.
  + **Không mất phí truy vấn**: Alias record trong Route 53 không tạo thêm phí truy vấn DNS khi ánh xạ tới tài nguyên AWS.
  + Alias record hỗ trợ **điểm đến (target)** có IP động, giúp cập nhật tự động mà không cần chỉnh sửa record.
* **Điểm nổi bật**:
  + Hỗ trợ root domain (ví dụ: example.com).
  + Tích hợp tốt với các tài nguyên AWS.

### **CNAME Record:**

* **Dùng khi nào?**
  + Khi muốn ánh xạ tên miền tới một **tên miền khác** (domain name), ví dụ:
    - Ánh xạ www.example.com tới example.com.
    - Ánh xạ một tên miền phụ tới tên miền quản lý bởi bên thứ ba.
  + CNAME không hỗ trợ ánh xạ root domain (như example.com), chỉ dùng cho tên miền phụ (như www.example.com).
* **Điểm nổi bật**:
  + CNAME chỉ ánh xạ tới một tên miền khác, không ánh xạ trực tiếp tới tài nguyên AWS.
  + Cần dùng khi không liên quan tới tài nguyên AWS hoặc cấu hình phức tạp hơn.

### **Kết luận:**

* **Dùng Alias**: Khi ánh xạ domain tới tài nguyên AWS (S3, ELB, CloudFront) hoặc root domain.
* **Dùng CNAME**: Khi ánh xạ domain tới một tên miền khác, đặc biệt là tên miền phụ.

======================================================================

#### **1. Simple Scaling Policy**

* **Cách hoạt động**:
  + Dựa trên các tiêu chí cơ bản, như CPU hoặc mạng vượt ngưỡng.
  + Khởi chạy hoặc chấm dứt các instance dựa trên một điều kiện duy nhất.
* **Hạn chế**:
  + Đơn giản nhưng không đủ linh hoạt để duy trì CPU utilization ở mức chính xác (gần 60%).
  + Thường gây chậm trễ trong phản ứng do thời gian chờ giữa các hoạt động scaling.
* **Không phù hợp** cho yêu cầu này.

#### **2. Step Scaling Policy**

* **Cách hoạt động**:
  + Dựa trên nhiều ngưỡng (thresholds), thực hiện scaling theo từng bước dựa trên mức độ nghiêm trọng của điều kiện.
  + Ví dụ: Tăng 2 instance nếu CPU vượt 70%, hoặc tăng 5 instance nếu CPU vượt 90%.
* **Hạn chế**:
  + Phù hợp với các điều kiện biến động lớn, nhưng không tối ưu để duy trì một mức CPU cụ thể.
  + Khó đạt độ chính xác cao, như duy trì gần 60% CPU utilization.
* **Không tối ưu** cho tình huống này.

#### **3. Scheduled Scaling Policy**

* **Cách hoạt động**:
  + Thực hiện scaling dựa trên lịch trình định trước (ví dụ: tăng số lượng instance vào giờ cao điểm).
* **Hạn chế**:
  + Phù hợp cho các kịch bản dự đoán được (như traffic cố định theo giờ/ngày).
  + Không đáp ứng được yêu cầu về việc giữ CPU ở một mức cụ thể.

#### **4. Target Tracking Policy**

* **Cách hoạt động**:
  + Sử dụng một mục tiêu cụ thể (ví dụ: 60% CPU utilization) để tự động điều chỉnh kích thước của Auto Scaling group.
  + Auto Scaling liên tục giám sát và thay đổi số lượng instance để giữ CPU utilization gần với mục tiêu.
* **Ưu điểm**:
  + Tự động hóa chính xác và phản ứng nhanh với sự thay đổi tài nguyên.
  + Hoạt động dựa trên một mục tiêu cụ thể, rất phù hợp với yêu cầu giữ CPU utilization ở 60%.

======================================================================

#### **1. AWS DataSync**

* **Cách hoạt động**:
  + Tự động hóa việc di chuyển dữ liệu từ on-premises sang AWS qua internet hoặc Direct Connect.
* **Hạn chế**:
  + Phụ thuộc vào tốc độ internet; không phù hợp với khối lượng lớn (30 TB) trong thời gian ngắn.

#### **2. AWS Storage Gateway**

* **Cách hoạt động**:
  + Cầu nối giữa lưu trữ on-premises và đám mây, có thể đồng bộ dữ liệu với Amazon S3.
* **Hạn chế**:
  + Cũng phụ thuộc vào internet. Không tối ưu cho việc di chuyển lượng lớn dữ liệu trong thời gian ngắn.

#### **3. AWS Snowball**

* **Cách hoạt động**:
  + Dịch vụ lưu trữ vật lý quy mô lớn (Snowball Edge) được vận chuyển đến địa điểm on-premises để nhập dữ liệu, sau đó gửi đến AWS.
* **Ưu điểm**:
  + Không phụ thuộc vào băng thông internet.
  + Có thể chuyển hàng chục terabyte dữ liệu nhanh chóng.

#### **4. Site-to-site VPN**

* **Cách hoạt động**:
  + Thiết lập kết nối VPN giữa on-premises và AWS để di chuyển dữ liệu.
* **Hạn chế**:
  + Dựa vào internet nên thời gian di chuyển quá lâu (223 ngày) không đáp ứng được giới hạn 20 ngày.

======================================================================

#### **2. AWS Lake Formation**

* **Cách hoạt động**:
  + AWS Lake Formation là dịch vụ được thiết kế để **xây dựng và quản lý Data Lake** một cách đơn giản và hiệu quả.
* **Ưu điểm**:
  + Tích hợp nhiều nguồn dữ liệu, bao gồm Amazon RDS MySQL, S3, và các nguồn khác.
  + Quản lý tập trung dữ liệu với khả năng kiểm soát quyền truy cập, bảo mật, và chuẩn bị dữ liệu cho Machine Learning.
  + Hỗ trợ dữ liệu sẵn sàng cho phân tích Machine Learning qua các dịch vụ khác như Amazon Athena và Amazon SageMaker.

#### **3. Amazon Quantum Ledger Database (QLDB)**

* **Cách hoạt động**:
  + QLDB là cơ sở dữ liệu bất biến (immutable ledger) được thiết kế để lưu trữ lịch sử giao dịch.
* **Hạn chế**:
  + Không phù hợp để lưu trữ và xử lý dữ liệu đa nguồn hoặc phân tích Machine Learning.
  + Thiết kế chủ yếu để đảm bảo tính bất biến của dữ liệu, không phải làm trung tâm dữ liệu phân tích.

#### **4. Amazon Neptune**

* **Cách hoạt động**:
  + Neptune là cơ sở dữ liệu đồ thị (graph database) được thiết kế để lưu trữ và phân tích mối quan hệ phức tạp giữa các thực thể.
* **Hạn chế**:
  + Không phải giải pháp lưu trữ trung tâm cho dữ liệu đa nguồn.
  + Chỉ phù hợp cho các bài toán đặc thù liên quan đến đồ thị, ví dụ: phân tích mạng xã hội, mối quan hệ.

======================================================================

#### **1. Amazon FSx for Windows File Server**

* **Cách hoạt động**:
  + Hệ thống file được thiết kế cho ứng dụng Windows, hỗ trợ giao thức SMB (Server Message Block).
* **Hạn chế**:
  + Không tối ưu cho các ứng dụng yêu cầu hiệu suất cao trong tính toán chuyên sâu.
  + Không cung cấp tích hợp liền mạch với S3 cho việc xử lý dữ liệu hiệu suất cao.

#### **2. Amazon FSx for Lustre**

* **Cách hoạt động**:
  + Hệ thống file hiệu suất cao được tối ưu cho các ứng dụng tính toán chuyên sâu (HPC).
  + Tích hợp liền mạch với Amazon S3, cho phép ánh xạ dữ liệu từ S3 dưới dạng file system.
* **Ưu điểm**:
  + Hiệu suất cao, độ trễ thấp, phù hợp với ứng dụng mô hình tài chính.
  + Hỗ trợ xử lý dữ liệu lớn và tích hợp với S3.

#### **3. Amazon Elastic File System (EFS)**

* **Cách hoạt động**:
  + Hệ thống file phân tán, được quản lý hoàn toàn, phù hợp cho nhiều máy chủ (instances) trong cùng một region.
* **Hạn chế**:
  + Không có tích hợp liền mạch với Amazon S3 như FSx for Lustre.
  + Hiệu suất thấp hơn FSx for Lustre, không tối ưu cho ứng dụng yêu cầu hiệu suất cao.
* **Không tối ưu**.

#### **4. Amazon Elastic Block Store (EBS)**

* **Cách hoạt động**:
  + Dịch vụ lưu trữ block-level gắn với các EC2 instances, thường được sử dụng cho database hoặc ứng dụng cần lưu trữ nhất quán.
* **Hạn chế**:
  + Không phải là file system dùng chung và không tích hợp trực tiếp với Amazon S3.

======================================================================

#### **1. Use the AWS Organizations API**

* **Cách hoạt động**:
  + AWS Organizations API cho phép tự động hóa việc tạo tài khoản thành viên, di chuyển tài khoản vào OU, và quản lý chính sách liên kết.
* **Ưu điểm**:
  + Tự động hóa đầy đủ thông qua lập trình.
  + Kết hợp với các dịch vụ khác như CloudFormation hoặc AWS SDK để cấu hình tài nguyên (VPC, Subnet).
  + Khả năng mở rộng và lặp lại cao.

**2. Use CloudFormation with scripts**

* **Cách hoạt động**:
  + CloudFormation có thể tự động triển khai tài nguyên như VPC và Subnet. Tuy nhiên, nó không thể tự tạo tài khoản trong AWS Organization.
* **Hạn chế**:
  + Không hỗ trợ việc tạo và quản lý tài khoản AWS Organizations.
  + Chỉ phù hợp để cấu hình tài nguyên trong tài khoản đã tồn tại.

#### **3. Use the AWS Management Console**

* **Cách hoạt động**:
  + Quản lý tài khoản và OU thông qua giao diện AWS Management Console.
* **Hạn chế**:
  + Quy trình thủ công, không lặp lại được.
  + Không đáp ứng yêu cầu tự động hóa hoặc khả năng quản lý hàng loạt.

#### **4. Use the AWS CLI**

* **Cách hoạt động**:
  + AWS CLI có thể sử dụng để tạo tài khoản và di chuyển tài khoản vào OU thông qua lệnh. Kết hợp CLI với script bash hoặc Python để tạo quy trình tự động.
* **Hạn chế**:
  + AWS CLI tự động hóa tốt, nhưng phức tạp hơn khi cần kết hợp với việc cấu hình tài nguyên (VPC/Subnet).

======================================================================

**App2Container (A2C)** là một công cụ do AWS cung cấp, giúp chuyển các ứng dụng truyền thống (monolithic) đang chạy trên hệ thống on-premises hoặc trên các máy chủ, thành các ứng dụng container để triển khai trên môi trường đám mây.

#### **Khi nào dùng App2Container:**

* Khi cần chuyển đổi các ứng dụng đang chạy trên hệ thống on-premises hoặc Windows/Linux server sang môi trường container trong AWS.
* Lý tưởng cho các doanh nghiệp muốn hiện đại hóa ứng dụng mà không cần viết lại từ đầu.

**App Runner** là dịch vụ **serverless** của AWS được thiết kế để triển khai và chạy các ứng dụng container hóa (hoặc ứng dụng web/API) mà không cần phải quản lý cơ sở hạ tầng như Amazon ECS hay Amazon EKS.

#### **Khi nào dùng App Runner:**

* Khi muốn triển khai ứng dụng container hóa với chi phí thấp và ít yêu cầu vận hành.
* Phù hợp với các ứng dụng nhỏ hoặc vừa không yêu cầu nhiều tùy chỉnh về kiến trúc container.

======================================================================

FIFO queues: 3000mess/persecond

Amazon SQS **Standard Queue** có thể xử lý **số lượng tin nhắn không giới hạn mỗi giây**.

======================================================================

#### **1. Backup and Restore**

* **Cách hoạt động**:
  + Dữ liệu được sao lưu định kỳ và khôi phục khi có sự cố.
* **Hạn chế**:
  + Không đảm bảo hệ thống cốt lõi sẵn sàng hoạt động ngay lập tức tại DR site.
  + Thời gian khôi phục lâu hơn so với chiến lược khác.
* **Không phù hợp**: Vì yêu cầu là sẵn sàng hoạt động ngay lập tức.

#### **2. Pilot Light**

* **Cách hoạt động**:
  + Một **phiên bản tối thiểu** của hệ thống cốt lõi được triển khai sẵn ở DR site. Các dịch vụ khác sẽ được khởi động khi cần.
* **Ưu điểm**:
  + DR site duy trì dịch vụ cốt lõi ở trạng thái tối thiểu, giảm chi phí vận hành.
  + Các dịch vụ cốt lõi sẵn sàng tiếp nhận khối lượng công việc ngay lập tức nếu xảy ra thảm họa.

#### **3. Warm Standby**

* **Cách hoạt động**:
  + DR site duy trì một **phiên bản hoạt động** của toàn bộ hệ thống nhưng ở trạng thái giảm tải (ít tài nguyên hơn).
* **Hạn chế**:
  + Bao gồm cả dịch vụ không cốt lõi (nhiều hơn yêu cầu).
  + Tăng chi phí vận hành vì tất cả hệ thống đều chạy liên tục.

#### **4. Multi-site**

* **Cách hoạt động**:
  + Hệ thống được triển khai đồng thời ở cả site chính và DR site, cả hai site đều hoạt động song song.
* **Hạn chế**:
  + Không có dịch vụ nào bị tắt, không phù hợp với yêu cầu.

======================================================================

#### **1. Amazon SWF (Simple Workflow Service)**

* **Cách hoạt động**:
  + Amazon SWF được thiết kế để quản lý quy trình công việc (workflows) phức tạp với nhiều bước, bao gồm cả việc phối hợp các dịch vụ và logic kinh doanh.

#### **2. Amazon SNS (Simple Notification Service)**

* **Cách hoạt động**:
  + SNS là dịch vụ **pub/sub** (publish/subscribe) cho phép truyền thông tin đến nhiều subscribers.
  + Có thể được sử dụng để kích hoạt Lambda một cách không đồng bộ.
* **Ưu điểm**:
  + Dễ dàng tích hợp với Lambda để thực hiện **asynchronous invocation**.
  + SNS giúp tách biệt các thành phần trong kiến trúc, đảm bảo **decoupling**.

#### **3. Amazon Kinesis**

* **Cách hoạt động**:
  + Dịch vụ được thiết kế để thu thập và xử lý dữ liệu thời gian thực từ nhiều nguồn.
* **Hạn chế**:
  + Tập trung vào phân tích dữ liệu theo thời gian thực hơn là thực hiện nhiệm vụ kích hoạt Lambda.
  + Không phù hợp với bài toán xử lý tin nhắn không đồng bộ đơn giản.

#### **4. Amazon OpsWorks**

* **Cách hoạt động**:
  + OpsWorks là dịch vụ quản lý cấu hình ứng dụng bằng cách sử dụng Chef hoặc Puppet.
* **Hạn chế**:
  + Không liên quan đến việc tách biệt compute services hoặc xử lý tin nhắn không đồng bộ.
  + Không phải là dịch vụ được thiết kế để kích hoạt Lambda.

======================================================================

### **AWS AppSync**

AWS **AppSync** là một dịch vụ **managed GraphQL** của AWS, giúp bạn xây dựng ứng dụng với khả năng truy cập và đồng bộ hóa dữ liệu một cách dễ dàng, nhanh chóng, và hiệu quả. Nó cho phép các ứng dụng truy cập dữ liệu từ nhiều nguồn, chẳng hạn như DynamoDB, Lambda, hoặc các REST APIs, thông qua GraphQL.

======================================================================

**AWS Wavelength**: Tập trung vào việc cung cấp dịch vụ đám mây ở rìa mạng (edge computing) trong mạng 5G công cộng, không phải cho mạng riêng.

**AWS Outposts**: Cung cấp cơ sở hạ tầng AWS tại chỗ (on-premises), không dành riêng cho mạng di động 5G.

**AWS CloudHSM**: Tập trung vào việc quản lý khóa mã hóa phần cứng, không liên quan đến mạng di động.

### **Amazon Keyspaces**

Amazon Keyspaces là một dịch vụ cơ sở dữ liệu NoSQL được quản lý hoàn toàn, tương thích với **Apache Cassandra** trên AWS. Nó cho phép bạn chạy các ứng dụng sử dụng Cassandra mà không cần phải tự thiết lập hoặc quản lý cơ sở hạ tầng.

======================================================================

#### **1. AWS Directory Service**

* **Cách hoạt động**:
  + AWS Directory Service chủ yếu được sử dụng để tích hợp với Active Directory và quản lý quyền truy cập cho ứng dụng và dịch vụ.

#### **2. AWS IAM Policy**

* **Cách hoạt động**:
  + AWS Identity and Access Management (IAM) kiểm soát quyền truy cập dựa trên vai trò (role) và chính sách (policy).

#### **3. Redis AUTH Command**

* **Cách hoạt động**:
  + Redis cung cấp lệnh **AUTH** để xác thực bằng mật khẩu.
  + ElastiCache hỗ trợ **AUTH** với tính năng "Redis AUTH" nhằm yêu cầu mật khẩu từ các client trước khi cho phép kết nối.

#### **4. VPC Security Group**

* **Cách hoạt động**:
  + Security Group kiểm soát lưu lượng mạng (inbound và outbound) đến các tài nguyên trong một VPC.

======================================================================

Tạo chính sách vòng đời cho các đối tượng trong thùng S3" .Chính sách vòng đời sẽ di chuyển dữ liệu sang một lớp lưu trữ khác nhưng không bảo vệ chống lại việc xóa.

======================================================================

#### **1. Reserved Instances**

* **Cách hoạt động**:
  + Reserved Instances cho phép đặt trước tài nguyên EC2 trong khoảng thời gian dài (1 hoặc 3 năm) với giá giảm đáng kể so với On-Demand.
* **Hạn chế**:
  + Batch processing công việc chỉ diễn ra trong một khoảng thời gian ngắn (5 ngày mỗi quý), vì vậy Reserved Instances không phù hợp, vì tài nguyên bị đặt trước cho toàn thời gian nhưng không sử dụng hết.

#### **2. Spot Instances**

* **Cách hoạt động**:
  + Spot Instances cung cấp EC2 với giá rẻ hơn, nhưng có thể bị thu hồi khi nhu cầu cao từ các khách hàng khác.
* **Hạn chế**:
  + Không đảm bảo tính sẵn sàng liên tục cho workload, do Spot Instances có thể bị dừng bất cứ lúc nào.
  + Không đáp ứng yêu cầu **không bị gián đoạn**.

#### **3. On-Demand Instances**

* **Cách hoạt động**:
  + On-Demand Instances cho phép sử dụng EC2 theo nhu cầu mà không cần đặt trước. Bạn chỉ trả tiền khi sử dụng.
* **Ưu điểm**:
  + Đáp ứng tính sẵn sàng và khả năng chạy liên tục cho công việc xử lý theo lô.
  + Không cần cam kết dài hạn, linh hoạt với công việc chỉ chạy vào cuối mỗi quý.
* **Phù hợp**: Đây là lựa chọn tốt nhất để đảm bảo tính liên tục và giảm chi phí.

#### **4. Dedicated Instances**

* **Cách hoạt động**:
  + Dedicated Instances cung cấp tài nguyên EC2 chỉ dành riêng cho một khách hàng trên phần cứng vật lý.
* **Hạn chế**:
  + Chi phí cao hơn nhiều so với các tùy chọn khác.
  + Không cần thiết nếu không có yêu cầu bảo mật cao hoặc tuân thủ đặc biệt.

======================================================================

CloudFront là dịch vụ CDN (Content Delivery Network) của AWS, hỗ trợ tính năng **Geo-Restriction** để chặn truy cập theo quốc gia.

Có thể định cấu hình CloudFront để từ chối traffic từ các quốc gia bị cấm mà không cần quản lý IP.

======================================================================

### **So sánh Network ACL và Security Groups**

| **Đặc điểm** | **Network ACL** | **Security Groups** |
| --- | --- | --- |
| **Phạm vi áp dụng** | Subnet | Instance |
| **Hỗ trợ deny rules** | Có | Không |
| **Hỗ trợ allow rules** | Có | Có |
| **Quy tắc direction** | Incoming và Outgoing | Incoming và Outgoing |
| **Kiểm soát IP/CIDR** | Có | Có |

### **Khi nào sử dụng?**

* **Network ACL**: Khi cần bảo mật toàn bộ subnet hoặc chặn các dải IP lớn.
* **Security Groups**: Khi cần quản lý bảo mật chi tiết hơn ở cấp instance.

======================================================================

### **Bài toán cho Network ACL**

#### **Tình huống:**

Một công ty muốn bảo vệ subnet của mình khỏi lưu lượng truy cập từ các quốc gia cụ thể do các mối đe dọa về bảo mật mạng.

Chặn toàn bộ traffic đến từ dải IP của quốc gia A và quốc gia B (ví dụ: dải IP 192.168.0.0/16 và 10.0.0.0/8).

======================================================================

### **Bài toán cho Security Groups**

#### **Tình huống:**

Một tổ chức sử dụng một ứng dụng web chạy trên EC2 instance. Chỉ có nhân viên nội bộ trong công ty được phép truy cập ứng dụng qua SSH (cổng 22).

#### **Giải pháp:**

* **Yêu cầu**:
  + Chỉ cho phép truy cập SSH từ IP nội bộ của công ty (ví dụ: 203.0.113.0/24).

======================================================================

1. **Legacy HPC application**:
   * Ứng dụng tính toán hiệu suất cao (High Performance Computing) yêu cầu giao tiếp mạng nhanh, độ trễ cực thấp.
   * Các ứng dụng HPC thường có các thành phần được liên kết chặt chẽ và phụ thuộc nhiều vào thông tin truyền qua mạng.
2. **Mục tiêu**:
   * Tăng cường hiệu suất mạng và giảm độ trễ để đáp ứng yêu cầu khắt khe của HPC trên AWS.

### **Phân tích từng tùy chọn**

#### **1. Elastic Network Interface (ENI)**

* **Cách hoạt động**:
  + ENI cung cấp kết nối mạng cơ bản giữa các EC2 instances trong VPC.
* **Hạn chế**:
  + ENI không tối ưu cho các tác vụ HPC yêu cầu độ trễ thấp và thông lượng cao.
  + Không cung cấp các tính năng đặc biệt cần thiết cho HPC.

#### **2. Elastic Network Adapter (ENA)**

* **Cách hoạt động**:
  + ENA cung cấp hiệu năng mạng cao, hỗ trợ thông lượng lên tới 100 Gbps.
* **Hạn chế**:
  + ENA rất tốt cho nhiều ứng dụng đòi hỏi hiệu suất cao, nhưng không hỗ trợ giao tiếp liên kết chặt chẽ cần thiết cho HPC.
  + Không hỗ trợ giao thức MPI (Message Passing Interface), vốn thường được sử dụng trong HPC.

#### **3. Elastic Fabric Adapter (EFA)**

* **Cách hoạt động**:
  + EFA được thiết kế đặc biệt cho ứng dụng HPC, cung cấp độ trễ thấp và hiệu suất mạng cao.
  + Hỗ trợ giao thức MPI, giúp tăng cường hiệu suất giao tiếp giữa các thành phần liên kết chặt chẽ của HPC.
* **Ưu điểm**:
  + Tối ưu cho HPC và các ứng dụng Machine Learning đòi hỏi giao tiếp nhanh và hiệu quả giữa các instances.
  + Cung cấp giao tiếp hiệu năng cao, giảm thiểu độ trễ mạng đáng kể.

======================================================================

#### **1. AWS Database Migration Service (DMS)**

* **Cách hoạt động**:
  + AWS DMS được thiết kế để di chuyển cơ sở dữ liệu từ on-premises hoặc dịch vụ đám mây khác sang AWS.
* **Hạn chế**:
  + Chỉ hỗ trợ di chuyển **dữ liệu cơ sở dữ liệu**, không hỗ trợ file shares SMB hoặc dữ liệu dạng file.

#### **2. AWS CloudFormation**

* **Cách hoạt động**:
  + AWS CloudFormation được sử dụng để tự động hóa triển khai tài nguyên AWS bằng các mẫu (templates).
* **Hạn chế**:
  + Không phải là dịch vụ di chuyển dữ liệu. CloudFormation không hỗ trợ chuyển file shares SMB vào S3.

#### **3. AWS Snowball**

* **Cách hoạt động**:
  + AWS Snowball là giải pháp vật lý để di chuyển lượng lớn dữ liệu qua thiết bị lưu trữ cứng khi băng thông mạng bị giới hạn.
* **Hạn chế**:
  + Không cần thiết cho trường hợp này vì tổ chức sử dụng AWS Direct Connect (đã có băng thông mạng mạnh).
  + Thích hợp hơn cho các trường hợp dữ liệu rất lớn và không thể di chuyển trực tuyến.

#### **4. AWS DataSync**

* **Cách hoạt động**:
  + AWS DataSync được thiết kế để tự động di chuyển dữ liệu giữa on-premises storage (bao gồm SMB file shares) và các dịch vụ AWS như Amazon S3.
  + Hỗ trợ giao thức SMB, NFS và Amazon S3.
* **Ưu điểm**:
  + Tự động hóa quy trình di chuyển dữ liệu.
  + Tối ưu hiệu suất khi sử dụng mạng tốc độ cao như AWS Direct Connect.
  + Theo dõi quá trình và xử lý lỗi tự động.

======================================================================

### **So sánh Amazon EFS và Amazon FSx for Windows File Server**

| **Tiêu chí** | **Amazon Elastic File System (EFS)** | **Amazon FSx for Windows File Server** |
| --- | --- | --- |
| **Hệ điều hành hỗ trợ** | Linux (tương thích POSIX) | Windows (không hỗ trợ Linux) |
| **Giao thức hỗ trợ** | NFS (Network File System v4/v4.1) | SMB (Server Message Block v2.0, v2.1, v3.0) |
| **Sử dụng phổ biến** | Container workloads, phân tích dữ liệu, môi trường Linux | Chia sẻ file doanh nghiệp, Windows applications |
| **Tích hợp Active Directory** | Không | Có (hỗ trợ quản lý người dùng Windows bằng AD) |
| **DFS (Distributed File System)** | Không hỗ trợ | Có hỗ trợ đầy đủ với DFSN (DFS Namespaces) |
| **Tính năng bảo mật** | Quản lý thông qua IAM, POSIX ACLs | Hỗ trợ Windows ACLs, tích hợp Active Directory |
| **Ứng dụng phổ biến** | Workload Linux như phân tích dữ liệu, machine learning | Hệ thống file Windows: Microsoft DFS, file shares |
| **Định giá** | Trả phí theo dung lượng lưu trữ | Trả phí theo dung lượng lưu trữ và throughput |
| **Tự động mở rộng (scaling)** | Có, tự động tăng/giảm kích thước lưu trữ | Có, nhưng tối ưu cho workload Windows |

### **So sánh Amazon FSx for Lustre và Amazon FSx for Windows File Server**

| **Tiêu chí** | **Amazon FSx for Lustre** | **Amazon FSx for Windows File Server** |
| --- | --- | --- |
| **Mục đích sử dụng** | Được thiết kế cho các ứng dụng đòi hỏi hiệu năng cao, như phân tích dữ liệu lớn (big data), AI/ML và HPC (High-Performance Computing). | Được thiết kế để cung cấp dịch vụ chia sẻ tệp cho các ứng dụng và người dùng dựa trên Windows. |
| **Hệ điều hành hỗ trợ** | Tương thích với các ứng dụng trên cả Linux và Windows. | Chỉ dành cho các ứng dụng Windows, hỗ trợ đầy đủ giao thức SMB. |
| **Hiệu năng** | Hiệu năng cao với độ trễ thấp, throughput lớn, tối ưu cho workload yêu cầu truy cập dữ liệu nhanh. | Hiệu suất tốt nhưng tối ưu cho nhu cầu chia sẻ và lưu trữ tệp thông thường. |
| **Giao thức** | Sử dụng giao thức Lustre File System để tối ưu hóa việc truy cập đồng thời vào dữ liệu. | Sử dụng giao thức SMB (Server Message Block), phù hợp với các ứng dụng và người dùng Windows. |
| **Tích hợp với AWS** | Tích hợp với Amazon S3: Có thể liên kết dữ liệu với S3 để truy cập nhanh và tiết kiệm chi phí lưu trữ. | Không tích hợp với S3, chủ yếu phục vụ nhu cầu chia sẻ file trong môi trường Windows. |
| **Ứng dụng phổ biến** | Phân tích dữ liệu lớn, xử lý video, mô phỏng khoa học, Machine Learning, HPC workloads. | Chia sẻ file doanh nghiệp, lưu trữ file cho máy chủ Active Directory hoặc ứng dụng Windows khác. |
| **Chi phí** | Tối ưu chi phí cho các workload đòi hỏi hiệu năng cao nhưng có thể cao hơn nếu không sử dụng đúng cách. | Tiết kiệm chi phí cho các ứng dụng Windows thông thường, dễ quản lý. |

======================================================================

AWS Config là một dịch vụ của Amazon Web Services giúp bạn quản lý và giám sát cấu hình của tài nguyên AWS. Nó cung cấp khả năng theo dõi, ghi lại lịch sử cấu hình (configuration history) và đánh giá trạng thái tài nguyên trong môi trường AWS, đồng thời đảm bảo chúng tuân thủ các chính sách quy định.

======================================================================

Patch Manager là một phần của AWS Systems Manager, được thiết kế để tự động hóa quy trình vá lỗi cho các tài nguyên AWS (như EC2, RDS).

**Ưu điểm**:

* Tự động phát hiện, tải xuống, và cài đặt bản vá theo lịch trình hoặc khi cần.
* Thích hợp cho việc quản lý và vá lỗi quy mô lớn.

Maintenance Window trong AWS Systems Manager cho phép lên lịch các nhiệm vụ quản lý (như vá lỗi) trong một khung thời gian cụ thể.

**Hạn chế**:

* Phụ thuộc vào thời gian đã lên lịch, không phù hợp nếu cần vá lỗi ngay lập tức để xử lý lỗ hổng bảo mật khẩn cấp.

======================================================================

Amazon QuickSight là một dịch vụ phân tích và trực quan hóa dữ liệu được quản lý hoàn toàn bởi AWS. QuickSight cho phép bạn tạo báo cáo, biểu đồ, và dashboard tương tác từ dữ liệu của bạn, đồng thời cung cấp khả năng phân tích chuyên sâu để đưa ra các quyết định dựa trên dữ liệu.

======================================================================

### **So sánh AWS SNS (Simple Notification Service) và AWS SQS (Simple Queue Service)**

| **Tiêu chí** | **Amazon SNS** | **Amazon SQS** |
| --- | --- | --- |
| **Loại hình dịch vụ** | Dịch vụ nhắn tin (pub/sub - publish/subscribe). | Dịch vụ xếp hàng thông điệp (message queue). |
| **Mô hình giao tiếp** | Đa điểm (fan-out): Một publisher có thể gửi thông báo đến nhiều subscriber. | Điểm-điểm (point-to-point): Một producer gửi thông điệp vào queue, một consumer xử lý từng thông điệp. |
| **Ứng dụng** | Gửi thông báo hoặc sự kiện đến nhiều hệ thống/phần mềm hoặc người nhận. | Xử lý và xếp hàng các yêu cầu để đảm bảo tính tuần tự và độ bền. |
| **Tính năng chính** | - Fan-out messages đến nhiều subscriber. <br> - Hỗ trợ email, SMS, Lambda, HTTP/HTTPS endpoints. | - Xếp hàng thông điệp để xử lý bất đồng bộ. <br> - Message durability: Lưu giữ thông điệp cho đến khi được xử lý. |
| **Tích hợp Lambda** | Tích hợp trực tiếp để kích hoạt Lambda dựa trên sự kiện. | Sử dụng **SQS Trigger** để kích hoạt Lambda function khi có thông điệp mới. |
| **Thông điệp tồn tại** | Không lưu trữ lâu, chỉ gửi đi ngay lập tức đến subscriber. | Lưu trữ thông điệp trong queue cho đến khi được xử lý hoặc hết thời hạn tồn tại (visibility timeout). |
| **Độ bền của thông điệp** | Không đảm bảo nếu các subscriber không nhận được thông báo. | Đảm bảo độ bền cao, thông điệp không bị mất trước khi được xử lý. |
| **Sử dụng phổ biến** | - Gửi thông báo hệ thống. <br> - Cảnh báo thời gian thực qua SMS, email. | - Xử lý batch, yêu cầu đồng bộ hoặc bất đồng bộ. <br> - Điều phối các tác vụ dựa trên hàng đợi thông điệp. |
| **Chi phí** | Miễn phí cho số lượng thông báo nhỏ, tính phí theo số lượng thông điệp và loại giao thức. | Tính phí dựa trên số lượng yêu cầu và thời gian lưu trữ thông điệp trong queue. |

### **Tóm tắt**

* **AWS SNS**: Lý tưởng cho giao tiếp đa điểm (fan-out), như gửi thông báo tới nhiều hệ thống hoặc người dùng. Phù hợp với các trường hợp yêu cầu **thời gian thực**.
* **AWS SQS**: Tối ưu cho xử lý bất đồng bộ với mô hình xếp hàng (queue), đảm bảo thông điệp không bị mất và xử lý theo thứ tự. Phù hợp với các workload cần điều phối hoặc xử lý batch.

======================================================================

#### **Virtual Private Gateway**

* **Cách hoạt động**:
  + Virtual Private Gateway (VGW) là điểm cuối trên Amazon VPC cho phép kết nối với mạng on-premises thông qua **AWS Site-to-Site VPN**.
* **Ưu điểm**:
  + Cấu hình nhanh chóng, không yêu cầu thiết lập vật lý như Direct Connect.
  + Hỗ trợ kết nối bằng **private IP addresses**, đáp ứng yêu cầu bảo mật và khả năng liên lạc với on-premises.

#### **AWS Direct Connect connection**

* **Cách hoạt động**:
  + AWS Direct Connect cung cấp kết nối mạng chuyên biệt và bảo mật cao giữa on-premises và AWS bằng cách sử dụng một liên kết vật lý (fiber-optic).
* **Hạn chế**:
  + Mất thời gian để thiết lập kết nối vì cần hợp tác với nhà cung cấp mạng để triển khai vật lý.
  + Không phải là giải pháp nhanh nhất.

======================================================================

### **AWS Systems Manager Run Command là gì?**

AWS Systems Manager **Run Command** là một tính năng mạnh mẽ trong AWS Systems Manager cho phép bạn chạy các lệnh (commands) trên các **managed instances** (EC2, hybrid servers hoặc on-premises servers) một cách đồng thời mà không cần truy cập SSH hoặc RDP vào từng instance.

### **Ứng dụng**

* **Quản lý và bảo trì**:
  + Thực hiện các tác vụ quản trị như cài đặt phần mềm, cập nhật cấu hình hoặc kiểm tra trạng thái hệ thống.
* **Khắc phục sự cố**:
  + Triển khai các script để xử lý lỗi hoặc sự cố mà không cần truy cập trực tiếp vào từng instance.
* **Quản lý trên quy mô lớn**:
  + Hỗ trợ xử lý đồng thời trên nhiều EC2 instances hoặc máy chủ hybrid.

### **Ưu điểm**

* **Bảo mật cao**: Không cần mở cổng SSH hoặc RDP, giảm rủi ro bảo mật.
* **Tiết kiệm thời gian**: Quản lý tập trung và xử lý đồng thời trên hàng trăm hoặc hàng ngàn instances.
* **Khả năng mở rộng**: Tự động hóa quy trình quản trị cho môi trường quy mô lớn.

### **Ví dụ sử dụng Run Command**

Bạn có thể dùng Run Command để:

* Cài đặt phần mềm trên 100 EC2 instances chỉ với một lệnh duy nhất.
* Khởi động lại các dịch vụ hệ thống mà không cần đăng nhập thủ công vào từng máy chủ.

======================================================================



| **Tiêu chí** | **AWS WAF** | **AWS Shield** |
| --- | --- | --- |
| **Mục đích chính** | Bảo vệ ứng dụng web khỏi các lỗ hổng bảo mật phổ biến ở **layer 7** (ứng dụng). | Bảo vệ chống lại các cuộc tấn công **DDoS** ở **layer 3 (network)**, **layer 4 (transport)**, và **layer 7 (application)**. |
| **Loại tấn công bảo vệ** | SQL Injection, Cross-Site Scripting (XSS), bad bots, và các lỗi bảo mật ứng dụng khác. | Distributed Denial of Service (DDoS). |
| **Phương pháp bảo vệ** | Áp dụng các quy tắc (rules) để cho phép, chặn hoặc giám sát lưu lượng dựa trên tiêu chí như: IP, HTTP headers, URI, body, etc. | Giảm thiểu các cuộc tấn công DDoS thông qua tự động phát hiện và phản ứng nhanh. |
| **Phiên bản miễn phí** | Không. Bạn chỉ trả phí cho số lượng quy tắc được sử dụng. | **Shield Standard**: Miễn phí, được kích hoạt mặc định cho tất cả tài nguyên AWS. |
| **Phiên bản nâng cao** | Không có phiên bản nâng cao, nhưng bạn có thể tuỳ chỉnh quy tắc cho các nhu cầu bảo mật phức tạp. | **Shield Advanced**: Có phí, cung cấp các tính năng bảo vệ DDoS nâng cao, hỗ trợ 24/7 từ AWS DDoS Response Team (DRT), và SLA tài chính. |
| **Ứng dụng chính** | - Ngăn chặn các hành vi độc hại ở tầng ứng dụng. <br> - Bảo vệ API Gateway, ALB (Application Load Balancer), CloudFront, AppSync. | - Bảo vệ toàn diện trước các cuộc tấn công DDoS. <br> - Bảo vệ tài nguyên như CloudFront, Global Accelerator, Elastic Load Balancer. |
| **Tính năng tự động** | Không tự động phát hiện tấn công; yêu cầu định nghĩa quy tắc thủ công. | Tự động phát hiện và phản ứng với tấn công DDoS. |
| **Tích hợp** | Tích hợp với Amazon CloudFront, API Gateway, AppSync, ALB. | Tích hợp tốt với CloudFront, Global Accelerator, và Elastic Load Balancer. |
| **Trường hợp sử dụng** | - Bảo vệ ứng dụng web khỏi các lỗi bảo mật phổ biến. <br> - Kiểm soát bot và lưu lượng bất thường ở layer 7. | - Ngăn chặn các cuộc tấn công DDoS lớn hoặc phức tạp. <br> - Đảm bảo tài nguyên luôn sẵn sàng. |

### **Tóm tắt**

* **AWS WAF**: Tập trung vào bảo vệ ứng dụng web (layer 7), giúp giảm thiểu các lỗi bảo mật phổ biến và kiểm soát lưu lượng truy cập đáng ngờ.
* **AWS Shield**: Cung cấp khả năng bảo vệ trước các cuộc tấn công DDoS, đặc biệt với **Shield Advanced** để xử lý các tấn công phức tạp trên nhiều layer mạng.

======================================================================

Lambda có thời gian thực thi tối đa là 900 giây và bộ nhớ có thể được phân bổ lên đến 3008 MB

======================================================================

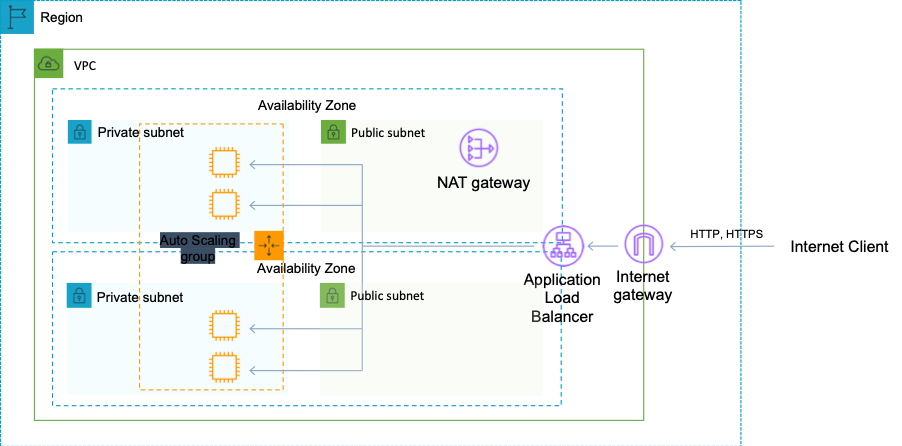
| **Tiêu chí** | **NAT Instance** | **NAT Gateway** |
| --- | --- | --- |
| **Loại dịch vụ** | EC2 instance do bạn quản lý | Dịch vụ được AWS quản lý (Managed service) |
| **Hiệu suất (throughput)** | Phụ thuộc vào loại EC2 instance | Tự động scale, hiệu suất cao |
| **High Availability (HA)** | Phải tự cấu hình với nhiều AZ | Tự động HA trong mỗi AZ |
| **Tự động scale** | ❌ Không, cần setup thủ công | ✅ Có sẵn, tự động mở rộng |
| **Chi phí** | Rẻ hơn nếu ít traffic (~ EC2 cost + data transfer) | Tốn kém hơn (theo giờ và theo GB traffic) |
| **Tùy chỉnh hệ thống (customizable)** | ✅ Cài phần mềm, thay đổi cấu hình tùy ý | ❌ Không thể tùy chỉnh |
| **Logging chi tiết** | Tùy bạn tự cấu hình (CloudWatch, syslog...) | Có tích hợp CloudWatch Logs |
| **Security Groups hỗ trợ** | ✅ Có thể gán Security Group | ❌ Không hỗ trợ Security Group (chỉ dùng Route Table) |
| **Yêu cầu bảo trì** | ✅ Có: cập nhật, patch, reboot khi cần | ❌ Không cần bảo trì |
| **Dễ triển khai** | ❌ Phức tạp hơn, cần cấu hình thủ công | ✅ Rất đơn giản, vài cú click |
| **Sử dụng phổ biến trong production** | Không thường dùng trừ khi cần tùy chỉnh sâu | ✅ Là lựa chọn mặc định và khuyến nghị của AWS |
| **Thích hợp cho** | Workload nhỏ, cần tùy chỉnh, chi phí thấp | Production, traffic lớn, cần hiệu suất và HA |

======================================================================

AT Gateway chỉ cung cấp kết nối outbound từ EC2 instances đến Internet, không nhận lưu lượng inbound từ người dùng.

Không đáp ứng yêu cầu xử lý lưu lượng từ người dùng Internet vào ứng dụng.

======================================================================



====================================================================

| **Loại Volume** | **Loại lưu trữ** | **Mô tả** | **Đặc điểm chính** |
| --- | --- | --- | --- |
| **gp2** (General Purpose SSD) | SSD | Dùng chung, cân bằng giữa cost và hiệu suất | Tối đa 16K IOPS, hiệu suất phụ thuộc vào dung lượng |
| **io1** (Provisioned IOPS SSD) | SSD | Thiết kế cho workload I/O cao | ✅ Hiệu năng cao, có thể provision IOPS riêng biệt, latency thấp |
| **st1** (Throughput Optimized HDD) | HDD | Dành cho truy cập theo block lớn (sequential), như big data | ❌ Không phù hợp cho random I/O |
| **sc1** (Cold HDD) | HDD | Giá rẻ, hiệu năng thấp, dùng cho dữ liệu ít truy cập | ❌ Không dùng cho cơ sở dữ liệu |

======================================================================

#### **Utilize Lake Formation tag-based access control for cross-account permissions**

* **Cách hoạt động**:
  + Sử dụng **Lake Formation Tag-Based Access Control** (LF-TBAC) để quản lý quyền truy cập dữ liệu dựa trên **tags**.
  + Tags được gán cho dữ liệu trong các tài khoản khác nhau, và marketing team được cấp quyền dựa trên tags.
* **Ưu điểm**:
  + Cấp quyền truy cập **cross-account** dễ dàng và bảo mật.
  + Tự động hóa quy trình quản lý quyền với ít thao tác thủ công.
  + Tích hợp chặt chẽ với Lake Formation.

======================================================================

### **AWS AppSync là gì?**

AWS AppSync là một dịch vụ được quản lý bởi AWS giúp bạn dễ dàng xây dựng và triển khai **API GraphQL**. AppSync cho phép các ứng dụng giao tiếp với nhau và với các nguồn dữ liệu một cách hiệu quả, đồng thời cung cấp khả năng đồng bộ hóa dữ liệu trong thời gian thực.

### **Tính năng chính**

1. **API GraphQL**:
   * AppSync hỗ trợ **GraphQL API**, giúp các ứng dụng có thể yêu cầu dữ liệu chính xác mà chúng cần, thay vì tải toàn bộ.
   * Cung cấp một endpoint duy nhất để truy cập nhiều nguồn dữ liệu.
2. **Thời gian thực (Real-time)**:
   * Hỗ trợ các tính năng như **subscriptions** và **websockets** để cập nhật dữ liệu thời gian thực, ví dụ: thông báo, trò chuyện.
3. **Tích hợp với nhiều nguồn dữ liệu**:
   * Tương thích với **Amazon DynamoDB**, **AWS Lambda**, **Amazon Aurora**, **Amazon Elasticsearch**, và **Amazon S3**.
   * Có thể tích hợp cả cơ sở dữ liệu tùy chỉnh hoặc API bên ngoài thông qua resolvers.
4. **Quản lý bảo mật**:
   * Tích hợp với **AWS Identity and Access Management (IAM)**, **Amazon Cognito**, và các cơ chế bảo mật như API keys, OIDC providers để kiểm soát quyền truy cập.
5. **Đồng bộ hóa ngoại tuyến (Offline)**:
   * Hỗ trợ ứng dụng ngoại tuyến: tự động đồng bộ hóa dữ liệu khi thiết bị kết nối lại internet.
6. **Tự động mở rộng**:
   * Dịch vụ tự động điều chỉnh để xử lý lưu lượng lớn hoặc biến động.

### **Lợi ích**

* **Hiệu quả**: Hỗ trợ GraphQL giúp giảm thiểu tải dữ liệu không cần thiết.
* **Thời gian thực**: Dễ dàng tích hợp các tính năng đồng bộ hóa thời gian thực.
* **Bảo mật cao**: Quản lý quyền truy cập dữ liệu linh hoạt.
* **Linh hoạt**: Có thể hoạt động với nhiều loại nguồn dữ liệu khác nhau.

### **Ứng dụng**

* Ứng dụng thời gian thực như trò chuyện, thông báo.
* Ứng dụng tương tác đa nền tảng (web, mobile).
* Quản lý và truy xuất dữ liệu từ nhiều nguồn.

======================================================================

Với EFS, bạn có thể chuyển đổi các tệp sang EFS IA sau khi tệp không được truy cập trong một khoảng thời gian nhất định với các tùy chọn lên đến 90 ngày. Bạn không thể chuyển đổi dựa trên độ tuổi là 2 năm.

======================================================================

#### **. Amazon Inspector**

* **Cách hoạt động**:
  + Amazon Inspector giúp đánh giá lỗ hổng bảo mật trong ứng dụng và cơ sở hạ tầng.
  + Kiểm tra tự động lỗ hổng của EC2 và container.
* **Hạn chế**:
  + Tập trung vào **đánh giá lỗ hổng bảo mật**, không phải phát hiện tấn công hoặc hành vi độc hại.

#### **Amazon Macie**

* **Cách hoạt động**:
  + Amazon Macie là công cụ phát hiện dữ liệu nhạy cảm và giám sát quyền truy cập S3.
* **Hạn chế**:
  + Tập trung vào việc bảo vệ và phân loại dữ liệu trong S3, không xử lý xâm phạm tổng quát hoặc các mối đe dọa.

#### **Amazon GuardDuty**

* **Cách hoạt động**:
  + GuardDuty là dịch vụ bảo mật được quản lý hoàn toàn, giúp phát hiện các hành vi độc hại và mối đe dọa trong tài khoản AWS.
  + Theo dõi **VPC Flow Logs**, **AWS CloudTrail**, và **DNS logs** để phát hiện xâm phạm.
  + Hoạt động liên tục, tự động giám sát và cảnh báo các hành vi đáng ngờ.
* **Ưu điểm**:
  + Bảo vệ toàn diện cho EC2, S3, và nhiều dịch vụ AWS khác.
  + Không yêu cầu cấu hình phức tạp.

======================================================================

RDS Multi-AZ is across availability zones, not across Region

| **Tiêu chí** | **Amazon RDS Multi-AZ** | **Amazon DynamoDB Global Tables** | **Amazon Aurora Global Database** | **Amazon RDS Cross-Region Replica** |
| --- | --- | --- | --- | --- |
| **Loại cơ sở dữ liệu** | **Quan hệ (relational)** | **Phi quan hệ (NoSQL)** | **Quan hệ (relational)** | **Quan hệ (relational)** |
| **Hỗ trợ Multi-Region** | **Không, hoạt động trong một Region duy nhất** | **Có, dữ liệu được tự động sao chép cross-Region** | **Có, hỗ trợ sao chép cross-Region** | **Có, nhưng là replica thủ công** |
| **RPO (Recovery Point Objective)** | **Không đảm bảo dưới 1 giây** | **< 1 giây (cập nhật đồng bộ)** | **< 1 giây** | **Phụ thuộc vào độ trễ giữa các Region** |
| **RTO (Recovery Time Objective)** | **Vài phút (tự động failover trong Region)** | **Không liên quan (NoSQL không có failover)** | **~1 phút (tự động failover cross-Region)** | **Vài phút (failover thủ công)** |
| **Quản lý tự động** | **Có, nhưng chỉ trong cùng một Region** | **Có, tự động sao chép và đồng bộ dữ liệu** | **Có, tự động failover và sao chép cross-Region** | **Không tự động, failover và quản lý thủ công** |
| **Tính phù hợp với yêu cầu** | **Không đáp ứng (không hỗ trợ Multi-Region)** | **Không đáp ứng (NoSQL, không phải relational)** | **Đáp ứng yêu cầu RPO/RTO** | **Không đáp ứng (failover thủ công, RTO cao)** |

======================================================================

======================================================================

======================================================================

============================================================================================================================================

======================================================================

======================================================================

======================================================================

======================================================================

============================================================================================================================================

======================================================================

======================================================================

======================================================================

======================================================================

======================================================================