AWS CERTIFIED SOLUTIONS ARCHITECT - ASSOCIATE

AWS SAA-C03 문제은행 결정 키워드 정리

문제은행 684문항과 개인 오답을 한 흐름으로 읽는 공부용 요약본

> 핵심 서비스 설명뿐 아니라 선택 조건, 비교 기준, 오답 제거 이유, 공식 범위 보충까지 마크다운 원문 순서 그대로 수록

2026-08-15 · OsoriAndOmori

목차

AWS Solutions Architect Associate(SAA-C03) 문제은행 684문항과 정답 해설, 직접 틀린 문제 메모를 기준으로 다시 정리했다. 서비스 설명을 사전처럼 나열하기보다, 문제의 조건에서 정답을 결정하는 표현과 비슷한 오답을 제거하는 기준에 집중한다.

> 이 글의 숫자는 제공된 문제 PDF에서 서비스명이 등장한 문항 수를 단순 집계한 값이다. 한 문항에 정답과 오답 서비스가 함께 있으므로 출제 비중의 절대값이 아니라, 복습 우선순위를 잡는 지표로만 본다.

이 문제은행, SAA-C03 범위가 맞나?

결론부터 말하면 핵심 범위는 대부분 맞고, 실전형 서비스 선택 연습용으로 충분히 넓다. 공식 SAA-C03은 보안 30%, 복원력 26%, 고성능 24%, 비용 최적화 20%의 네 도메인으로 구성된다. 이 문제은행도 IAM/KMS/VPC 보안, Multi-AZ/Auto Scaling/DR, 스토리지·DB·네트워크 성능, 구매 옵션·스토리지 클래스 비용 문제가 반복된다.

공식 도메인	문제은행에서 반복되는 대표 주제	판단
Secure Architectures 30%	IAM Role, Organizations, KMS, S3 policy, SG/NACL, WAF/Shield	핵심 충분
Resilient Architectures 26%	Multi-AZ, Auto Scaling, SQS decoupling, backup, multi-Region DR	핵심 충분
High-Performing Architectures 24%	EBS/FSx, ElastiCache, DynamoDB, CloudFront, Kinesis, read scaling	핵심 충분
Cost-Optimized Architectures 20%	Spot/RI, S3 class, NAT 비용, serverless, Cost Explorer	핵심 충분

다만 유명한 문제은행 = 현재 시험 범위를 완벽하게 보장하는 공식 자료는 아니다.

- 원본 PDF 생성 시점이 2024년이라 **AWS SSO**, **Amazon Elasticsearch Service**, **Kinesis Data Firehose**처럼 옛 이름이 섞여 있다. 현재 이름은 IAM Identity Center, Amazon OpenSearch Service, Amazon Data Firehose다.
- 일부 해설은 지나치게 단정적이거나 부정확하다. 예를 들어 "SQS queue는 초당 3,000 messages가 최대"라는 설명은 Standard queue와 FIFO queue의 조건을 섞은 것이다.
- 공식 in-scope 목록에 있지만 이 문제은행에서 거의 보이지 않는 서비스도 있다. Application Migration Service, Compute Optimizer, X-Ray, Artifact/Audit Manager, Managed Grafana/Prometheus, Kendra, Keyspaces 등이 대표적이다.
- 반대로 EC2, S3, Lambda, RDS, ELB, VPC처럼 실제 설계 판단의 중심이 되는 서비스는 매우 많이 반복된다. 시험 대비 효율 면에서는 자연스러운 편중이다.

따라서 이 자료는 문제은행으로 패턴을 익히되, 정답 해설은 절대적인 사실로 외우지 않고 공식 기능과 비교하는 방식으로 사용한다. 아래 정리에는 문제은행에서 빠진 공식 범위 중 SAA 수준에서 알아둘 가치가 있는 항목도 보충했다.

먼저 볼 것: 반복 출제 지도

우선순위	문제은행 등장 문항 수	먼저 구분할 것
EC2	313	Auto Scaling, 구매 옵션, EBS, 배치 그룹
S3	273	스토리지 클래스, 전송, 복제, 암호화, 정적 웹
Lambda	179	비동기 처리, 동시성, SQS/API Gateway 연동
RDS	135	Multi-AZ, Read Replica, Aurora, RDS Proxy
Auto Scaling	115	ALB, 큐 깊이 기반 확장, 상태 확인
DynamoDB	74	On-demand, DAX, Global Tables, Streams
CloudFront	69	캐시·콘텐츠 배포와 Global Accelerator 구분
API Gateway	61	REST API, Lambda, Private API, 사용량 제한
EBS / SQS / Aurora	57 / 54 / 48	블록 스토리지, 버퍼링, DB 고가용성

우선순위	문제은행 등장 문항 수	먼저 구분할 것
KMS / SNS / ECS / Kinesis	44 / 41 / 41 / 40	암호화, fan-out, 컨테이너, 스트리밍

문장을 읽을 때는 **요구사항 -> 제약 -> 운영 부담 -> 비용** 순서로 표시한다.

1. 무엇을 보장해야 하나? 고가용성, 내구성, 순서, 고정 IP, RPO/RTO, 암호화.
2. 무엇을 쓸 수 없나? 인터넷 없음, 코드 변경 최소, 온프레미스 유지, 특정 리전 사용 불가.
3. 운영 부담을 얼마나 줄여야 하나? **least operational overhead**이면 관리형·서버리스·기본 기능을 우선한다.
4. 비용 조건은 마지막에 적용한다. 요구사항을 못 맞추는 싼 선택지는 정답이 아니다.

S3와 데이터 이동

스토리지 클래스 결정표

문제의 결정 문구	정답 후보	오답 제거 포인트
접근 패턴을 예측할 수 없음	S3 Intelligent-Tiering	수명 주기로 시점을 정하려면 패턴을 알아야 함
자주 접근하지 않지만 즉시 조회	Standard-IA	최소 저장 기간·검색 비용 확인
한 AZ 손실을 감수할 수 있는 재생성 가능 데이터	One Zone-IA	고가용성 요구면 제외
보관, 분~시간 단위 복원 허용	Glacier Flexible Retrieval	즉시 조회 요구면 제외
장기 보관, 가장 저렴, 복원 지연 허용	Glacier Deep Archive	최소 저장 기간 확인
날짜별 접근 패턴이 명확	Lifecycle	**Standard -> IA -> Glacier -> Expiration**

시험식 암기: **unpredictable access pattern** -> Intelligent-Tiering.

전송·복제·잠금

상황	선택
전 세계 지점 + 고속 인터넷 + 한 버킷으로 빠른 업로드	S3 Transfer Acceleration + multipart upload
온라인 반복 전송, NFS/SMB/객체 스토리지	DataSync
수십 TB 이상, 네트워크 사용 최소, 일회성 오프라인 이동	Snowball Edge
SFTP/FTPS/FTP/AS2 인터페이스 유지	AWS Transfer Family
온프레미스에서 NFS/SMB로 보이되 S3에 저장	S3 File Gateway
새 객체를 다른 리전/계정으로 자동 복제	S3 Replication; 양쪽 버전 관리 필요
기존 객체까지 한 번에 복제	S3 Batch Replication
WORM, 보존 기간 중 삭제 방지	S3 Object Lock; Compliance는 root도 삭제 불가

가장 빠르게만 보고 Snowball을 고르지 않는다. 고속 인터넷이 이미 있고 매일 전송한다면 Transfer Acceleration, 네트워크 대역폭을 거의 쓰지 않는 대용량 일회성 이전이면 Snowball이다.

공유 파일 시스템

결정 문구	선택	핵심
Linux, NFS, 여러 AZ의 EC2가 동시 공유	EFS	리전 단위 관리형 공유 파일 시스템
Windows IIS, SMB, 기존 Windows NAS/AD	FSx for Windows File Server	Windows 네이티브 파일 공유
HPC, 머신러닝, 유전체, 렌더링, S3 연동 병렬 처리	FSx for Lustre	초고속 분산 파일 시스템

결정 문구	선택	핵심
NetApp, NFS+SMB+iSCSI, ONTAP 기능 유지	FSx for NetApp ONTAP	NetApp 워크로드 이전
단일 EC2 중심 블록 스토리지	EBS	AZ 종속; 스냅샷은 S3에 관리됨

오답 메모: Windows IIS + 기존 NAS 파일 공유 -> FSx for Windows, 여러 Linux EC2가 같은 파일을 봄 -> EFS.

EBS 성능

- **gp2**: 용량이 커지면 기준 IOPS도 함께 증가한다.
- **gp3**: 용량과 IOPS·처리량을 독립적으로 설정한다. 일반적인 비용 최적화 교체 후보.
- **io2**: 일관된 고 IOPS·낮은 지연이 중요한 데이터베이스.
- 스냅샷 복원 직후 모든 블록을 즉시 최고 성능으로 읽어야 하면 Fast Snapshot Restore.
- EBS는 기본적으로 한 AZ의 블록 스토리지다. 여러 AZ 공유 문제를 스냅샷 복제로 해결하려 하지 않는다.

S3 암호화 키 관리

방식	누가 키를 관리하나	문제의 신호
SSE-S3	S3가 전부 관리	별도 키 제어 없이 기본 서버측 암호화
SSE-KMS + AWS managed key	AWS KMS의 **aws/s3** 키	KMS 감사·권한 통제는 필요하나 별도 키 생성 없음
SSE-KMS + customer managed key	내가 KMS 키 정책·회전·권한 관리	키 회전과 접근을 직접 제어, 교차 계정 권한
SSE-C	고객이 키를 제공하고 보관	요청마다 키 제공; KMS 키와 다름

SSE-KMS와 **customer managed key**는 동의어가 아니다. SSE-KMS는 AWS managed key 또는 customer managed key를 모두 쓸 수 있다. 문제에 키 회전·키 정책을 직접 통제가 나오면 customer managed KMS key를 고른다.

정적 웹사이트

HTML/CSS/JS/이미지만 있는 사이트는 S3 static website hosting으로 웹 서버 없이 제공할 수 있다. 글로벌 HTTPS, 캐시, 오리진 보호까지 요구하면 앞에 CloudFront를 둔다. 동적 서버 코드가 필요한데 S3만 고르는 것은 오답이다.

S3 접근·복제 세부 함정

- S3 버킷은 기본적으로 private이다. 퍼블릭 정적 사이트와 private S3 origin + CloudFront OAC를 구분한다.
- 객체 단위 임시 다운로드/업로드 권한: presigned URL. AWS 자격 증명을 사용자에게 주지 않는다.
- 소유권이 다른 업로더 문제: S3 Object Ownership - Bucket owner enforced와 ACL 비활성화를 먼저 본다.
- Versioning은 덮어쓰기·삭제로부터 복구할 수 있게 하지만 비용이 계속 쌓인다. noncurrent version Lifecycle도 함께 본다.
- CRR/SRR은 source와 destination 모두 Versioning이 필요하다. 기존 객체는 일반 replication rule만으로 자동 복제되지 않으므로 Batch Replication을 본다.
- SSE-KMS 복제는 destination KMS key 권한과 replication role 권한까지 필요하다.
- 수백만 객체에 태그·복사·복원 작업: S3 Batch Operations. 액세스 패턴과 비용 가시화는 S3 Storage Lens.

컴퓨팅, 확장, 구매 옵션

EC2 구매 옵션

워크로드	선택
24시간 켜지는 예측 가능한 baseline	Reserved Instances 또는 Savings Plans
짧고 수량 변화가 크며 중단 가능	Spot Instances
시작·종료가 자유롭고 약정 없음	On-Demand
물리 서버 단독 사용, 라이선스/규정	Dedicated Host
특정 AZ의 용량을 반드시 확보	Capacity Reservation

오답 메모의 전형: 항상 실행하는 Frontend -> Reserved, 짧게 실행되고 탄력적이며 중단 가능한 Backend -> Spot. 단, 중단 불가·상태 저장 작업이면 Spot을 비용만 보고 고르지 않는다.

고가용성 웹 계층

Route 53/CloudFront -> ALB -> 여러 AZ의 Auto Scaling group -> 관리형 DB/공유 스토리지가 기본 뼈대다.

- ALB: HTTP/HTTPS, 경로·호스트 기반 라우팅, Lambda도 대상 가능.
- NLB: TCP/UDP/TLS, 초고성능, 고정 IP 요구.
- Gateway Load Balancer: 방화벽·IDS/IPS 같은 가상 네트워크 어플라이언스를 투명하게 삽입.
- Auto Scaling은 여러 AZ에 배치하고 ALB health check를 함께 사용한다.
- 작업 큐가 있으면 CPU보다 SQS ApproximateNumberOfMessagesVisible(큐 깊이) 기반 확장이 자연스럽다.
- 세션/업로드 파일을 로컬 EBS에 두면 확장 시 데이터가 갈라진다. 세션은 ElastiCache/DynamoDB, 공유 파일은 EFS/FSx/S3로 외부화한다.

배치 그룹

- Cluster: 한 AZ, 가까운 하드웨어, 낮은 지연·높은 처리량. 함께 시작하기 어렵고 장애 범위가 크다.
- Spread: 소수의 중요한 인스턴스를 서로 다른 하드웨어로 최대 분산.
- Partition: HDFS/Kafka/Cassandra 같은 대규모 분산 시스템의 파티션별 장애 격리.

Lambda 함정

- Reserved Concurrency: 함수가 사용할 수 있는 동시 실행 수를 예약하면서 상한도 설정.
- Provisioned Concurrency: 실행 환경을 미리 준비해 cold start 감소.
- SQS 트리거의 visibility timeout은 함수 처리·재시도 시간을 감당하도록 잡는다.
- DB 연결 폭증은 RDS Proxy로 연결 풀링한다.
- 장시간·특수 런타임·서버 제어가 필요하면 Lambda만 고집하지 말고 ECS/Fargate 또는 EC2를 본다.

컨테이너와 배포

결정 문구	선택
컨테이너 실행, 서버 관리 원치 않음	ECS/EKS + Fargate
AWS 네이티브 오케스트레이션, 운영 단순	ECS
Kubernetes API·생태계·이식성	EKS
컨테이너 이미지 저장·취약점 스캔	ECR

결정 문구	선택
큐 기반 대규모 배치 작업	AWS Batch
코드 배포와 인프라 운영을 더 추상화	Elastic Beanstalk

Fargate는 오케스트레이터가 아니라 ECS 또는 EKS에서 사용하는 serverless compute option이다. 이미지 저장소는 ECR, 실행·스케줄링은 ECS/EKS로 역할을 나눈다.

Auto Scaling 정책과 배포 안전장치

- Target tracking: 평균 CPU 50%처럼 목표값을 유지. 가장 단순한 기본 선택.
- Step scaling: 알람 크기에 따라 단계별 증감. 급격한 변화에 세밀한 조절.
- Scheduled scaling: 예측 가능한 시간대 트래픽.
- Predictive scaling: 과거 패턴을 예측해 미리 용량 확보.
- Lifecycle hook: launch/terminate를 잠시 멈추고 초기화·로그 수집.
- Warm pool: 초기화된 인스턴스를 대기시켜 긴 부팅 시간을 줄임.
- Launch configuration보다 Launch Template을 우선한다. 여러 instance type·Spot 혼합 정책에도 필요하다.

데이터베이스와 캐시

RDS Multi-AZ를 먼저 구분

배포	목적	읽기 분산
Multi-AZ DB instance deployment	고가용성·자동 장애 조치	standby는 읽기용으로 사용하지 않음
Multi-AZ DB cluster deployment	고가용성 + 빠른 장애 조치	reader 인스턴스와 reader endpoint 제공
Read Replica	읽기 확장, 리전 간 복제 가능	비동기 복제; DR 시 승격 가능

문제에 읽기 부하를 primary에서 분리 + reader endpoint + 빠른 failover가 함께 나오면 Multi-AZ DB cluster다. 단순히 **Multi-AZ**라는 단어만 보고 모든 standby를 읽기용이라고 생각하지 않는다.

Aurora·RDS 선택

- Aurora cluster endpoint: writer 연결.
- Aurora reader endpoint: replica에 읽기 연결 분산.
- Aurora Global Database: 한 primary 리전, 여러 secondary 리전의 낮은 지연 글로벌 읽기·재해 복구.
- RDS Proxy: Lambda/짧은 연결이 많은 앱의 DB 연결 풀 관리, 장애 조치 영향 감소.
- 엔진 변경 최소 + 관리 부담 감소: 온프레미스 MySQL/PostgreSQL을 RDS/Aurora로.
- 이기종 DB 마이그레이션: SCT로 스키마 변환 + DMS로 데이터 이동.

DynamoDB·캐시

신호	선택
트래픽 예측 불가, 운영 최소	DynamoDB On-demand
예측 가능한 트래픽, 용량 계획 가능	Provisioned + Auto Scaling
마이크로초 읽기 캐시	DAX
다중 리전 active-active	Global Tables
항목 변경 이벤트 처리	DynamoDB Streams + Lambda

신호	선택
관계형 DB 세션·캐시·순위·분산 잠금	ElastiCache for Redis
단순 멀티스레드 캐시, 샤딩	Memcached

읽기 확장을 위해 RDS standby를 고르지 않는다. 관계형 읽기 확장은 Read Replica/Aurora reader, 키-값 초저지연은 DynamoDB/DAX, 반복 조회 캐시는 ElastiCache를 본다.

DynamoDB 세부 결정

- Partition key는 트래픽이 고르게 분산되도록 high-cardinality 값을 고른다. 특정 key에 요청이 몰리면 hot partition이 된다.
- GSI는 새로운 partition/sort key로 전체 테이블을 조회한다. LSI는 같은 partition key 안에서 다른 sort key를 사용하며 테이블 생성 시 정의한다.
- TTL은 만료 항목을 자동 삭제하는 비용 효율적인 방법이지만 즉시 삭제 보장은 아니다.
- 조건부 쓰기와 transaction은 동시성 충돌·원자성 요구에 사용한다.
- Point-in-time recovery는 연속 백업, on-demand backup은 특정 시점 장기 보존에 적합하다.

분석 서비스 선택

요구	선택
S3 파일을 서버 없이 SQL로 질의	Athena
데이터 카탈로그·ETL·schema discovery	AWS Glue
대규모 Spark/Hadoop 처리	Amazon EMR
컬럼형 데이터 웨어하우스	Redshift
로그·검색·대시보드	OpenSearch Service
BI 시각화·대시보드	QuickSight
데이터 레이크 권한 중앙 관리	Lake Formation

Athena는 저장소가 아니라 S3 데이터를 질의하는 엔진이다. Redshift Spectrum도 S3를 읽을 수 있지만, 이미 Redshift 분석 환경이 있거나 웨어하우스 조인이 핵심인지 확인한다.

네트워크와 글로벌 전송

CloudFront vs Global Accelerator vs Route 53

결정 문구	정답
정적·동적 웹 콘텐츠, HTTP/HTTPS, edge cache, 오리진 부하 감소	CloudFront
고정 글로벌 Anycast IP, TCP/UDP, AWS 글로벌 네트워크, 빠른 endpoint failover	Global Accelerator
DNS 응답으로 리전 선택, latency/weighted/failover/geolocation 정책	Route 53

HTTP라고 무조건 CloudFront가 아니다. 캐싱이 핵심이면 CloudFront, HTTP/HTTPS라도 고정 글로벌 IP + 멀티리전 장애 조치 + 캐시 불필요가 핵심이면 Global Accelerator가 더 맞다. 일반 TCP/UDP 애플리케이션에 CloudFront를 고르지 않는다.

시험 암기: **Global static IP + TCP/UDP + 글로벌 최적 경로** -> Global Accelerator.

Route 53 라우팅 정책

정책	결정 문구
Simple	특별한 정책 없이 하나의 리소스
Weighted	blue/green, A/B, 비율별 분산
Latency-based	사용자에게 지연이 가장 낮은 리전
Failover	active-passive + health check
Geolocation	사용자의 국가·대륙 위치에 따른 규칙
Geoproximity	리소스 위치와 bias로 지리적 분산
Multi-value answer	여러 healthy record를 DNS 응답; load balancer 대체는 아님

Alias record는 AWS 리소스(ALB, CloudFront, S3 website 등)를 루트 도메인에도 연결할 수 있고 Route 53 query 요금 측면의 장점이 있다. CNAME은 zone apex에 사용할 수 없다.

VPC 사설 연결

- S3/DynamoDB를 인터넷·NAT 없이 접근: Gateway VPC endpoint. 라우팅 테이블에 경로 추가, 별도 시간 요금 없음.
- 대부분의 AWS 서비스를 ENI의 사설 IP로 접근: Interface endpoint(PrivateLink). Security Group 적용, 시간·데이터 요금.
- 특정 서비스만 다른 VPC에 사설 노출, CIDR 중복 가능: PrivateLink, 제공 측은 보통 NLB.
- VPC 전체를 1:1 연결: VPC Peering. 비전이성이라 A-B, B-C가 있어도 A-C 통신 불가.
- 다수 VPC/VPN 중앙 허브: Transit Gateway.

온프레미스 연결·DNS

- Site-to-Site VPN: 빠르게 구축, 인터넷 기반 암호화 터널.
- Direct Connect: 전용 회선, 일관된 대역폭·지연. 자체적으로 암호화를 제공한다는 뜻은 아니므로 필요하면 VPN과 결합.
- 한 DX로 여러 리전 VPC/TGW 접근: Direct Connect Gateway.
- 온프레미스가 Route 53 Private Hosted Zone을 질의: Resolver Inbound endpoint.
- VPC가 온프레미스 도메인을 질의: Resolver Outbound endpoint + forwarding rule.
- 특정 대도시 근처에서 한 자릿수 ms 지연으로 워크로드 실행: AWS Local Zones. CloudFront처럼 캐시만 두는 서비스가 아니다.

보안 경계

- Security Group: ENI/인스턴스 수준, stateful, allow 규칙만.
- Network ACL: subnet 수준, stateless, allow/deny, 요청·응답 양방향 규칙.
- NAT Gateway는 private subnet의 아웃바운드 인터넷용이지 외부에서 들어오는 연결을 허용하지 않는다.

Load Balancer 세부 구분

- ALB: Layer 7, HTTP/HTTPS, host/path/header/query 기반 라우팅, WAF 연동.
- NLB: Layer 4, TCP/UDP/TLS, source IP 보존, 고정 IP/EIP, 갑작스러운 대규모 트래픽.
- GWLB: GENEVE 기반으로 방화벽·IDS/IPS fleet을 경로에 삽입.
- Cross-zone load balancing, deregistration delay, health check가 선택지에 나오면 "부하 분산"과 "장애 조치"를 따로 본다.
- ALB access log는 S3, 애플리케이션 지표는 CloudWatch, API 변경 감사는 CloudTrail이다.

메시징, 이벤트, 스트리밍

SNS와 SQS를 분리해서 생각

- SNS: push pub/sub, 한 이벤트를 여러 구독자에게 fan-out.
- SQS: 큐에 쌓아 소비 속도를 조절, 생산자와 소비자를 decouple, 재시도·DLQ.
- 여러 서비스가 같은 이벤트를 각자 안정적으로 처리: SNS topic -> 서비스별 SQS queue -> consumer.
- DB가 순간 폭주를 감당하지 못함: SNS에서 Lambda를 바로 폭증시키기보다 SQS로 버퍼링하고 소비 동시성을 제한.
- **Standard queue**는 매우 높은 처리량, at-least-once, best-effort ordering이므로 소비자를 멱등하게 만든다.
- 순서와 중복 제거가 필수: FIFO queue. **MessageGroupId**가 같은 메시지 안에서 순서가 보장된다.

SQS 운영 키워드

- Visibility timeout: 처리 중 메시지를 다른 consumer에게 숨기는 시간. 처리보다 짧으면 중복 실행이 늘어난다.
- Long polling: 빈 응답과 API 호출 비용을 줄인다.
- Dead-letter queue: 반복 실패 메시지 격리. maxReceiveCount와 redrive를 함께 본다.
- Delay queue/message timer: 일정 시간 뒤 메시지를 보이게 한다.
- Standard는 at-least-once이므로 exactly-once processing을 보장한다고 외우지 않는다. 애플리케이션 멱등성이 필요하다.
- FIFO의 deduplication ID는 중복 전송 억제, message group ID는 그룹 내 순서·병렬성 단위다.

EventBridge·Step Functions

- AWS 서비스/SaaS/커스텀 이벤트를 규칙으로 라우팅: EventBridge.
- 과거 이벤트를 다시 처리: EventBridge Archive and Replay.
- 여러 단계, 분기, 재시도, 병렬 처리, 사람 승인 등의 워크플로: Step Functions.
- 단순 fan-out이면 SNS, 소비 속도 조절이면 SQS, 이벤트 규칙 기반 라우팅이면 EventBridge다.

Kinesis Data Streams vs Data Firehose

요구	선택
소비자가 직접 읽고 재처리, shard·순서·체크포인트, 실시간 앱	Kinesis Data Streams
S3/Redshift/OpenSearch 등에 관리형으로 버퍼링·변환·배달	Amazon Data Firehose
Kafka API·생태계가 필요	Amazon MSK

암기: Data Streams = Kafka처럼 스트림을 보관하고 소비자가 읽음, Firehose = 목적지로 알아서 적재하는 파이프라인. Firehose는 일반적인 메시지 큐 대체가 아니다.

Kinesis Data Streams의 처리량은 shard와 on-demand/provisioned mode를 본다. 여러 consumer가 독립적으로 높은 처리량을 요구하면 enhanced fan-out이 후보이고, 스트림 SQL/Apache Flink 처리는 Amazon Managed Service for Apache Flink를 떠올린다. Kafka 호환 API와 기존 Kafka 도구가 결정타면 MSK다.

보안, 계정, 규정 준수

교차 계정과 Organizations

- 개발 계정 사용자가 운영 계정 S3에 접근: 운영 계정에 IAM user를 새로 만들지 말고 운영 계정 IAM Role의 trust policy가 개발 계정을 신뢰하게 한다. 개발 사용자는 **AssumeRole**로 최소 권한 임시 자격 증명을 얻는다.
- AWS 계정 ID를 수십 개 나열하지 않고 조직 전체만 허용: 리소스 정책의 **aws:PrincipalOrgID**.
- 특정 OU 경로만 제한: **aws:PrincipalOrgPaths**.
- SCP는 Organizations 계정의 최대 권한 경계이며 권한을 직접 부여하지 않는다.
- Permissions boundary는 특정 IAM principal이 가질 수 있는 최대 권한을 제한하며 역시 권한을 직접 부여하지 않는다.

SSO와 Microsoft AD

온프레미스 Microsoft AD의 사용자·그룹을 계속 관리 + 여러 AWS 계정 SSO 조건이면:

IAM Identity Center + AWS Managed Microsoft AD + self-managed AD와 two-way forest trust

계정마다 IAM user를 복제하는 선택지는 운영 부담과 보안 측면에서 탈락한다.

자격 증명과 비밀 관리

- 애플리케이션에 access key를 하드코딩하지 않는다. EC2 instance profile, ECS task role, Lambda execution role을 사용한다.
- DB password 자동 회전이 핵심이면 Secrets Manager.
- 구성값·간단한 secret을 계층적으로 저장하고 비용을 줄이면 Systems Manager Parameter Store. 회전 기능 요구를 확인한다.
- 웹·모바일 최종 사용자 가입/로그인과 token은 Amazon Cognito. 직원의 여러 AWS 계정 SSO는 IAM Identity Center다.
- 퍼블릭 TLS 인증서 발급·자동 갱신은 ACM. CloudFront 인증서는 us-east-1 요구를 기억한다.
- 전용 HSM과 직접 키 제어 요구는 CloudHSM, 일반적인 관리형 키·서비스 통합은 KMS다.

Config vs CloudTrail vs CloudWatch

질문	서비스
누가 언제 어떤 API로 설정을 바꿨나	CloudTrail
리소스 구성이 원하는 상태인가, 변경 이력·규칙 준수 여부	AWS Config
비준수 리소스를 자동 수정	AWS Config rule + Systems Manager Automation
지표·로그·알람·대시보드	CloudWatch
AWS 계정 없는 외부인에게 대시보드만 공유	CloudWatch Dashboard Sharing

오답 메모: "누가 바꿨나" -> CloudTrail, "올바른 설정인가를 지속 평가" -> Config. 암호화되지 않은 볼륨을 자동으로 찾아 고치는 문제는 Config가 탐지하고 SSM Automation이 수정하며, 실행 역할에는 필요한 IAM 권한만 준다.

보안 서비스 한 줄 판별

- GuardDuty: CloudTrail/VPC Flow Logs/DNS 등의 신호로 위협 탐지.
- Inspector: EC2·컨테이너 이미지·Lambda의 취약점 관리.
- Macie: S3의 민감 데이터 발견·분류.
- Rekognition: 이미지·동영상 분석, 부적절 콘텐츠 탐지.

- Comprehend: 텍스트의 감정·엔터티·주제·언어 분석.
- WAF: HTTP(S) Layer 7 규칙, SQL injection/XSS/IP 제어.
- Shield Advanced: DDoS 보호 강화와 비용 보호·지원.
- Firewall Manager: Organizations 전반의 WAF/Shield/보안 정책 중앙 관리.

공식 범위 보충: 적게 나온 관리·보안 서비스

- AWS Artifact: AWS 규정 준수 보고서와 계약 문서 다운로드.
- Audit Manager: 감사 증거를 지속 수집하고 평가 보고서 구성.
- Security Hub: 여러 보안 서비스 finding과 표준 준수 상태를 중앙 집계.
- Detective: 보안 finding 이후 관계와 활동을 시각화해 조사.
- Compute Optimizer: 사용량 기반 EC2/EBS/Lambda 등의 right-sizing 추천.
- X-Ray: 분산 요청 trace, service map, latency 원인 분석.
- Application Migration Service(MGN): 서버를 block-level replication으로 lift-and-shift.
- Managed Grafana/Prometheus: Kubernetes·애플리케이션 메트릭 시각화와 Prometheus 호환 모니터링. 핵심 출제 빈도는 낮다.

비용, 모니터링, 재해 복구

비용 도구

요구	선택
서비스·리전·인스턴스 타입 등으로 비용 추세 분석	Cost Explorer
예산/사용량 임계치 초과 알림	AWS Budgets
가장 상세한 원천 청구·사용 데이터	Cost and Usage Report(CUR)
전체 청구 현황을 간단히 확인	Billing Dashboard

EC2 비용이 왜 늘었는지, 최근 2개월을 instance type별로 분석, least operational overhead -> Cost Explorer.
CUR + S3 + Athena/QuickSight는 더 세밀하지만 구축이 필요하므로 "최소 운영 부담"에서는 밀릴 수 있다.

모니터링·로그

- CloudWatch metric/alarm: 수치 임계치와 알림.
- CloudWatch Logs: 애플리케이션·시스템 로그 수집 및 Logs Insights 쿼리.
- S3의 JSON/CSV 로그를 간단한 SQL로 즉시 분석: Athena.
- X-Ray: 분산 애플리케이션 요청 추적과 병목 분석.
- CloudTrail: AWS API 감사. 조직 전체는 organization trail.

비용 최적화 추가 포인트

- Compute Savings Plans: instance family/리전/OS 변화에 더 유연. EC2 Instance Savings Plans보다 할인 유연성이 크다.
- Reserved Instance: EC2/RDS 등 서비스별 약정 할인. Standard와 Convertible의 교환 유연성을 구분한다.
- NAT Gateway 비용: 같은 AZ 경로, S3/DynamoDB gateway endpoint, interface endpoint의 시간·처리 비용을 함께 비교한다.
- CloudFront cache는 사용자 지연뿐 아니라 origin data transfer와 처리 부하도 줄일 수 있다.
- Compute Optimizer는 right-sizing 추천, Cost Explorer는 실제 비용 추세·필터 분석, Budgets는 임계치 알림이다.

- Trusted Advisor는 비용·보안·성능·내결함성 등의 권장 사항, CUR은 가장 상세한 원천 비용 데이터다.

DR 전략

전략	비용	RTO/RPO	결정 문구
Backup & Restore	최저	가장 김	평소 컴퓨팅 없음, 백업에서 복원
Pilot Light	낮음	김	핵심 DB/데이터만 항상 실행
Warm Standby	중간	짧음	축소 환경이 실행 중, 장애 시 scale out
Multi-Site Active/Active	최고	가장 짧음	양쪽이 실제 트래픽 처리

- RPO: 얼마나 많은 데이터 손실을 허용하는가.
- RTO: 얼마나 오래 중단을 허용하는가.
- 백업 주기·복제 방식은 RPO를, 프로비저닝·전환 시간은 RTO를 결정한다.
- Route 53 failover, Global Accelerator endpoint health, Aurora Global Database 같은 선택지는 요구 프로토콜·고정 IP·DB 엔진까지 함께 본다.

다중 선택 문제의 조합 패턴

요구	자주 맞물리는 조합
웹 계층 HA + DB HA	ALB + multi-AZ ASG + RDS Multi-AZ
서버리스 API + 관계형 DB 보호	API Gateway + Lambda + RDS Proxy
다수 소비자 fan-out + burst 완충	SNS + consumer별 SQS
정적 웹 글로벌 보안 배포	private S3 + CloudFront OAC + ACM/WAF
온프레미스 대용량 반복 동기화	Direct Connect/VPN + DataSync
조직 전체 보안 규칙	Organizations + Firewall Manager/Config aggregator
암호화되지 않은 리소스 자동 수정	Config rule + SSM Automation + execution role
S3 데이터 레이크 분석	S3 + Glue Data Catalog + Athena + Lake Formation

"Choose two/three"에서는 각 선택지가 같은 계층을 중복 해결하는지 확인한다. 예를 들어 ALB와 Route 53은 서로 완전 대체재가 아니라 DNS 진입과 HTTP 부하 분산으로 함께 쓰일 수 있다.

자주 틀리는 결정타 25개

1. **Windows + SMB + IIS/NAS** -> FSx for Windows File Server.
2. **Linux + NFS + 여러 AZ 공유** -> EFS.
3. **HPC/ML + S3 병렬 처리** -> FSx for Lustre.
4. **global static IP + TCP/UDP** -> Global Accelerator.
5. **웹 콘텐츠 캐시 + HTTP/HTTPS** -> CloudFront.
6. **DNS 기반 가까운 리전/가중치/장애 조치** -> Route 53 routing policy.
7. **S3/DynamoDB + private + NAT 없음** -> Gateway endpoint.
8. **특정 서비스만 사설 노출 + CIDR 중복** -> PrivateLink.
9. **조직 전체 계정만 허용** -> **aws:PrincipalOrgID**.
10. **교차 계정 임시 권한** -> trust policy + AssumeRole.
11. **Multi-AZ DB instance** -> HA, standby 읽기 불가.
12. **Multi-AZ DB cluster + reader endpoint** -> HA + 읽기 분산.

13. **항상 켜지는 baseline** -> RI/Savings Plans.
14. **중단 가능 + 짧고 탄력적** -> Spot.
15. **gp3** -> 용량과 IOPS/처리량을 독립 조정.
16. **unpredictable S3 access** -> Intelligent-Tiering.
17. **부적절한 이미지** -> Rekognition; 텍스트 분석은 Comprehend.
18. **비준수 탐지 + 자동 수정** -> Config + SSM Automation.
19. **누가 API 변경** -> CloudTrail; **현재 구성이 준수** -> Config.
20. **비용 원인 필터 분석** -> Cost Explorer; **초과 알림** -> Budgets.
21. **burst를 DB 앞에서 완충** -> SQS.
22. **한 이벤트를 여러 서비스로** -> SNS + 서비스별 SQS.
23. **스트림 직접 소비·재처리** -> Kinesis Data Streams.
24. **스트림을 S3/Redshift/OpenSearch로 배달** -> Data Firehose.
25. **AWS 계정 없는 사람에게 대시보드만** -> CloudWatch Dashboard Sharing.

마지막 30초 오답 제거

- 문제의 형용사 하나를 놓치지 않는다: **least**, **most**, **immediately**, **unpredictable**, **without internet**, **static IP**, **no code changes**.
- 요구사항을 하나라도 못 맞추면 익숙한 서비스여도 제거한다.
- 관리형 기능이 있는데 EC2에 직접 설치·cron·사용자 코드를 만드는 답은 **least operational overhead**에서 의심한다.
- 고가용성은 백업만으로 해결되지 않고, 읽기 확장은 standby만으로 해결되지 않는다.
- DNS, 캐시, 네트워크 가속을 섞지 않는다: Route 53 / CloudFront / Global Accelerator.
- 인증과 권한을 섞지 않는다: Identity Center는 SSO, IAM Role은 임시 권한, KMS key policy는 암호화 키 사용 권한.
- 데이터 형식을 본다: 객체는 S3, 블록은 EBS, Linux 공유 파일은 EFS, Windows SMB는 FSx for Windows.
- 실시간이라는 말만 보고 Kinesis를 고르지 않는다. fan-out, 버퍼링, 재처리, 목적지 적재 중 무엇이 필요한지 먼저 정한다.

참고한 공식 문서

- AWS SAA-C03 시험 가이드와 도메인 비중
- SAA-C03 공식 in-scope AWS 서비스
- Amazon S3 서버 측 암호화
- RDS Multi-AZ DB cluster 연결과 reader endpoint
- AWS Global Accelerator 동작 방식
- Amazon SQS queue types
- AWS Config 비준수 리소스 수정
- Amazon FSx for Windows File Server
- Amazon Data Firehose 데이터 전송