CloudNinja.pro

PROFESSIONAL · DOP-C02

AWS Certified DevOps Engineer – Professional

DOP-C02 Cheat Sheet

CloudNinja · cloudninja.pro · Free — no signup required

Every service in scope, every automation pattern, every incident-response and resilience decision the DOP-C02 exam actually tests — organised around the six exam domains. Assumes ≥ 2 years of hands-on AWS provisioning, operations and automation.

EXAM SNAPSHOT

- **75 questions** — 65 scored + 10 unscored
- **180 minutes** total time
- **750 / 1000** to pass
- **$300 USD** exam fee

- **SDLC Automation** — 22%
- **Configuration Management & IaC** — 17%
- **Resilient Cloud Solutions** — 15%
- **Monitoring & Logging** — 15%
- **Incident & Event Response** — 14%
- **Security & Compliance** — 17%

FREE DOP-C02 practice exam →

1 · Foundations & Tooling

- **Region / AZ / Edge** — design every pipeline for multi-AZ first; multi-Region where RPO/RTO warrant it. Edge locations serve CloudFront, Route 53, Global Accelerator, Lambda@Edge.
- **AWS CLI, AWS SDKs and Tools, AWS CloudShell** — CLI for scripting, SDKs for apps (Python, Node, Java, .NET, Go, Ruby, PHP, C++, Rust). CloudShell is browser-based with 1 GB persistent home. Use for ad-hoc automation when laptops lack creds.
- **AWS Management Console** — fine for discovery, never for production change. DevOps answer is always pipeline or IaC.
- **IAM Identity Center** — federated SSO, Permission Sets per OU/account. Pipelines assume cross-account roles rather than using long-lived keys.
- **AWS Health** — Personal Health Dashboard + Organizational view. EventBridge integration fires automated runbooks on AWS-side issues before users notice.

2 · Organizations, Landing Zone & Governance

Canonical Multi-Account Layout

OU / Account	Purpose
Management	Organizations, billing, Control Tower only. No workloads.
Security OU	Log Archive + Audit accounts. Central CloudTrail, Config, GuardDuty, Security Hub admin.
Deployments OU	CI/CD tooling: CodePipeline, CodeBuild, CodeDeploy, CloudFormation StackSets.
Shared Services OU	Transit Gateway, IPAM, Directory Service, DNS, artefacts, container registries.
Workloads OU	Prod / staging / dev sub-OUs. SCPs tighten closer to prod.
Sandbox OU	Developer experimentation with budget caps + SCPs.

Governance Building Blocks

- **AWS Organizations + SCPs** — SCPs cap permissions (never grant). Classic patterns: deny region, deny root user, deny disabling CloudTrail, deny `iam:*` outside identity account.
- **AWS Control Tower** — managed landing zone: log-archive + audit accounts, IAM Identity Center, guardrails (preventive=SCP, detective=Config), Account Factory / AFT for IaC-driven provisioning.
- **AWS Service Catalog** — curated CloudFormation portfolios developers self-service. Launch Constraints + TagOptions enforce standards.
- **AWS RAM** — share Transit Gateways, subnets, Route 53 Resolver rules, License Manager configs, private CAs across accounts.
- **CloudFormation StackSets** — deploy stacks to many accounts/Regions from one template. Service-managed permissions with Organizations = auto target new accounts.
- **AWS Config Conformance Packs** — packaged rule sets deployed org-wide via delegated admin. Auto-remediation via SSM Automation documents.
- **Delegated administration** — move admin out of the management account for GuardDuty, Security Hub, Config, CloudTrail, IAM Access Analyzer, Macie, Inspector, Firewall Manager, Audit Manager, StackSets.

3 · Identity, Encryption & Secrets

Service	DevOps Role
AWS IAM	Users, groups, roles, policies. **Roles for EC2/Lambda/CodeBuild** — never embed keys. IAM Access Analyzer finds external / unused access.

Service	DevOps Role
AWS STS	Temporary creds. Cross-account pipeline steps assume role via `sts:AssumeRole`; session tags carry identity attributes.
AWS IAM Identity Center	Workforce SSO across accounts. Permission Sets become IAM roles; sessions last 1–12 h.
AWS Directory Service	Managed AD (full), AD Connector (proxy to on-prem), Simple AD. Used for Windows fleet joining and EC2 RDS domain auth.
Amazon Cognito	User pools (identity) + identity pools (AWS creds). Common in developer/self-service portals fronting pipelines.
AWS KMS	CMKs for everything encrypted: CodePipeline artefacts, Secrets Manager, Parameter Store SecureString, EBS, S3, RDS. **Cross-account pipelines need key policy granting the other account.**
AWS CloudHSM	FIPS 140-2 Level 3 HSM cluster when single-tenant HSM is mandatory.
AWS Certificate Manager (ACM)	Public + private certs. Free, auto-renewed on AWS integrated services (ALB/NLB/CloudFront/API GW). Private CA for internal PKI.
AWS Secrets Manager	Rotation via Lambda, cross-account sharing via resource policy, RDS/Redshift/DocumentDB native rotation.
AWS Systems Manager Parameter Store	Free (Standard), SecureString with KMS. Hierarchical paths; pipelines pull env config by path.

Secrets vs Parameter Store — Decision

Need	Pick
Automatic rotation (DB creds, API keys)	Secrets Manager
Free, config / feature flags, non-rotating	Parameter Store Standard
High throughput (> 40 TPS) + change notifications	Parameter Store Advanced or Secrets Manager
Cross-account, cross-Region replication	Secrets Manager (native replication)

4 · Threat Detection & Compliance Automation

Service	Automation Pattern
Amazon GuardDuty	Threat detection on VPC Flow, DNS, CloudTrail, EKS audit, S3, RDS, Lambda, Runtime Monitoring. **EventBridge** rule on findings → SSM Automation quarantine runbook.
Amazon Inspector	Continuous vulnerability scanning for EC2, ECR images, Lambda. Push CVE findings to Security Hub → ticket via Jira integration.
Amazon Detective	Graph-based incident investigation across GuardDuty findings, VPC Flow, CloudTrail.
Amazon Macie	PII/PHI discovery in S3. Alerts drive tagging + auto-remediation (block public, enforce encryption).
AWS Security Hub	Aggregator of findings (GuardDuty, Inspector, Macie, IAM Access Analyzer, 3rd-party). CIS, PCI, AWS FSBP standards. **Delegated admin** account per Org.
AWS Config	Resource inventory + compliance. Rules + auto-remediation via SSM Automation. Aggregator for multi-account dashboards.
AWS CloudTrail	Management + data events. Organization trail → S3 in Log Archive account with Object Lock for WORM retention.

Service	Automation Pattern
AWS Trusted Advisor	Cost, performance, security, fault tolerance, service limits. EventBridge events on check status change.
AWS Audit Manager	Automated evidence collection for SOC, PCI, HIPAA, GDPR.
AWS Firewall Manager	Central management of WAF rules, Shield Advanced, Security Groups, Network Firewall, Route 53 Resolver DNS Firewall across accounts.

5 · Networking for DevOps Workloads

Service	Role
Amazon VPC	Public, private, isolated subnets per AZ; NAT Gateway for egress-only workloads. Flow Logs to S3/CloudWatch for audit.
Elastic Load Balancing (ELB)	ALB (HTTP/HTTPS, path / host / header / weighted routing, blue/green via weighted target groups), NLB (TCP/UDP/TLS, static IPs, PrivateLink backend), GWLB (inline security appliances).
AWS PrivateLink	Expose a service to other VPCs/accounts without VPC peering or public IPs. Producer side = NLB; consumer = VPC interface endpoint.
AWS Transit Gateway	Hub for thousands of VPCs + Direct Connect + VPN. Route tables isolate segments. Share via RAM.
AWS Site-to-Site VPN	IPsec VPN over internet to VGW or TGW. Good for burst or failover DR.
AWS Client VPN	Managed OpenVPN for workforce access to VPC; integrates with IAM Identity Center / SAML.
Amazon Route 53	Simple / weighted / latency / geolocation / geoproximity / failover / multivalue routing. **Health checks** drive automated DNS failover. Private zones per VPC.
Amazon CloudFront	Global CDN + DDoS (Shield Standard free). OAC for private S3, signed URLs/cookies, Lambda@Edge + CloudFront Functions for request/response manipulation.
AWS Global Accelerator	Anycast IPs routed over AWS backbone; health-based regional failover in seconds. Ideal for non-HTTP.
AWS WAF	L7 rules on ALB/API GW/AppSync/CloudFront: managed rules, rate limits, CAPTCHA. Deploy org-wide via Firewall Manager.
AWS Shield (Standard / Advanced)	DDoS. Advanced adds SRT, cost protection, Global Threat Environment dashboard.
AWS Network Firewall	Stateful L3–L7 filtering at VPC edge. Deep-packet inspection, Suricata rules, TLS inspection.

6 · Compute

Amazon EC2 — Automation Essentials

- **Instance families** — C (compute), M (general), R (memory), T (burstable), I (NVMe), G/P (GPU), Graviton *g* suffix (ARM; cheapest $/perf for web, CI agents, caches).
- **Purchasing** — On-Demand · Reserved (1/3 y) · Savings Plans (Compute/EC2/SageMaker) · Spot (up to 90% off, 2-min interruption notice). Mix via Auto Scaling mixed-instances + allocation strategies (*capacity-optimized* for Spot, *price-capacity-optimized* default).
- **User data** — bootstrap at first boot. Idempotent scripts + `cloud-init` tag. Prefer **SSM State Manager** or **Image Builder** for repeatable config.
- **Instance metadata** — always use **IMDSv2** (token-based); block v1 via launch template.
- **Placement groups** — Cluster (low latency), Spread (max 7/AZ physical isolation), Partition (big data racks).

- **Nitro Enclaves** — isolated CPU/RAM slice with no persistent storage or networking — for keys / PII.

EC2 Image Builder

- Managed golden-AMI pipeline: source image + components + tests + distribution. Signs images, publishes SBOM, shares across accounts/Regions, triggers on component/image updates via EventBridge.
- Typical pattern: Image Builder → AMI → launch templates → Auto Scaling rolling refresh on new AMI.

Amazon EC2 Auto Scaling + AWS Auto Scaling

- **Scaling policies** — Target Tracking (preferred), Step Scaling, Simple, Scheduled, Predictive (ML uses 14-day history).
- **Health checks** — EC2 status + ELB health. *Instance refresh* replaces instances in-place when launch template version changes — honour minimum healthy % and warm-up.
- **Lifecycle hooks** — pause on launching/terminating to run custom actions (drain queues, take snapshots).
- **AWS Auto Scaling** — unified scaling plans across ASGs, ECS services, DynamoDB, Aurora replicas, Spot Fleet.
- **AWS Compute Optimizer** — ML-based right-sizing recommendations. Surface in Cost Explorer + feed CI to auto-PR IaC changes.

AWS Elastic Beanstalk

- PaaS for Java, .NET, Node, Python, Ruby, PHP, Go, Docker, multi-container. **.ebextensions** YAML/JSON config files stay in source.
- **Deployment policies**: All-at-once (fastest, downtime), Rolling, Rolling with additional batch, Immutable (new ASG, safest in-place), **Blue/Green** (swap URL via Route 53 / environment swap).
- Worker environments process SQS messages; web environments front-ended by ALB.

> **Exam cue:** "Zero-downtime, full rollback on failure" with Elastic Beanstalk → **Immutable** or **Blue/Green**. "Fastest, dev only" → All-at-once.

7 · Containers & Kubernetes

Service	Purpose
Amazon ECR	Private + public registry. Image scanning (Basic or Enhanced via Inspector), lifecycle policies, cross-Region/cross-account replication, signing via Signer.
Amazon ECS	Native orchestrator. **EC2** launch type (control over instances), **Fargate** (serverless). Task definitions = containers + IAM role + logging.
Amazon EKS	Managed Kubernetes control plane. Node groups: EC2 self-managed, Managed, Fargate profiles, Karpenter for just-in-time.
Amazon EKS Distro	Same Kubernetes distribution, bring-your-own-infra on prem / bare metal.
Red Hat OpenShift Service on AWS (ROSA)	Managed OpenShift. Jointly supported by AWS + Red Hat.
AWS Fargate	Serverless pods/tasks. Per-second billing, no patching, scales horizontally fast.
AWS Copilot	CLI that scaffolds ECS + Fargate apps, pipelines, environments. Great for greenfield.
AWS App2Container	Lift-and-shift Java, .NET, IIS apps into container images + ECS/EKS manifests.
AWS App Runner	Fully managed container or source-code web service; auto-builds from repo, autoscales, HTTPS. No VPC management unless you opt in.
AWS Batch	Managed batch on Fargate / EC2 / EKS. Queues, job definitions, compute environments, Array jobs for fan-out.

Container Deployment Strategies

- **ECS rolling update** — default; controlled by *min/max healthy %*. Fast but partial bad versions can serve traffic.
- **ECS + CodeDeploy Blue/Green** — CodeDeploy shifts ALB target group traffic from blue to green. Canary / linear / all-at-once shift; CloudWatch alarm rollback; lifecycle hooks.
- **ECS external deployment** — controller you run (Argo, Flux). Useful when GitOps is the system of record.
- **EKS** — kubectl rolling update, Argo Rollouts / Flagger for canary, service mesh (App Mesh, Istio) for traffic shifting.

8 · Serverless

AWS Lambda

- Runtimes: Node, Python, Java, Go, .NET, Ruby, custom (Rust, Swift) via Lambda runtime interface.
- Memory 128 MB–10 GB (CPU scales linearly), timeout 15 min max, payload 6 MB sync / 256 KB async, ephemeral `/tmp` 512 MB–10 GB.
- **Versions + aliases** — immutable versions, aliases point to one version with weighted traffic for canary.
- **Provisioned concurrency** — pre-warm to kill cold starts for spiky, latency-sensitive APIs.
- **SnapStart** (Java) — init snapshot restored on invoke = sub-second cold start.
- **Lambda Extensions** — side-car processes for observability (Datadog, Dynatrace).
- **Function URLs** — direct HTTPS, IAM or public. Good for internal tools, webhook sinks.
- **Event sources** — API Gateway, ALB, EventBridge, SNS, SQS, Kinesis, DynamoDB Streams, S3, CodeCommit, CodePipeline.
- **DLQ / on-failure destinations** — async failures routed to SQS/SNS/EventBridge/Lambda.

Other Serverless Primitives

- **Amazon API Gateway** — REST, HTTP (lighter/cheaper), WebSocket. Auth: IAM, Cognito, Lambda authorizer. Usage plans + API keys. Stage variables for pipeline promotion.
- **AWS Step Functions** — Standard (durable, >1 y) + Express (event streaming, < 5 min). Retry / Catch / parallel / map. Integrates natively with 200+ services. **First answer for long-running, stateful workflows.**
- **AWS Serverless Application Model (SAM)** — CloudFormation transform, `sam local` for local Lambda/API GW emulation, `sam deploy` wraps CFN.
- **AWS Serverless Application Repository (SAR)** — share private or public SAM apps across accounts.

9 · Storage & Databases

Service	DevOps Notes
Amazon S3	Versioning, Object Lock (WORM), lifecycle → IA / Glacier Instant / Flexible / Deep Archive. Replication (SRR/CRR, RTC 15 min SLA). Event Notifications → SQS/SNS/Lambda/EventBridge. Access Points, Multi-Region Access Points, Batch Operations.
Amazon S3 Glacier	Flexible / Deep Archive; vault lock policies for compliance retention.
Amazon EBS	gp3 (default, independently scale IOPS/throughput), io2 Block Express (sub-ms, 256K IOPS, Nitro), st1 / sc1 (throughput). DLM for lifecycle snapshots. Multi-Attach on io1/io2 for cluster software.
Amazon EFS	NFS multi-AZ. Standard / IA tiers, lifecycle auto-tiering. Good for shared CI scratch, Jenkins masters.
Amazon FSx for Windows File Server	SMB + AD integration, backups via AWS Backup.
Amazon FSx for Lustre	HPC, ML training, genomics. Link to S3 for dataset repo.
Amazon FSx for NetApp ONTAP	SnapMirror, SnapVault, multi-protocol NFS/SMB/iSCSI — enterprise NAS migrations.

Service	DevOps Notes
Amazon FSx for OpenZFS	POSIX with ZFS snapshots/clones for DB dev-copy workflows.
AWS Storage Gateway	File / Volume / Tape gateways — hybrid cache/backup bridge.
AWS Backup	Central backup: EBS, EFS, RDS, Aurora, DynamoDB, FSx, EC2, Storage Gateway, S3, VMware. Backup plans via Organizations. Cross-account + cross-Region copy, vault lock.
AWS Elastic Disaster Recovery (DRS)	Continuous block-level replication; sub-second RPO. Launch drill instances on demand.

Databases

Service	When to use
Amazon RDS	Managed MySQL / PostgreSQL / MariaDB / Oracle / SQL Server / Db2. Multi-AZ (sync) + read replicas. Blue/Green deployments for low-risk upgrades. Performance Insights.
Amazon Aurora	MySQL / PG compatible. Storage auto-grow to 128 TiB. Up to 15 read replicas, Global Database (<1 s cross-Region). Aurora Serverless v2 for variable workload.
Amazon DynamoDB	Single-digit ms NoSQL; on-demand or provisioned with auto-scaling. Global Tables multi-Region active-active. Streams → Lambda. PITR + on-demand backup.
Amazon ElastiCache	Redis OSS / Valkey / Memcached. Redis supports replication + Multi-AZ + Global Datastore.
Amazon MemoryDB for Redis	Durable Redis-compatible primary DB (multi-AZ transaction log).
Amazon DocumentDB	MongoDB-compatible managed doc store.
Amazon Redshift	Columnar DW, RA3 decoupled compute/storage, Redshift Serverless, data sharing.
AWS DMS	Homogeneous + heterogeneous migration, CDC streaming. Pair with SCT for schema.

10 · Application Integration

Service	Purpose
Amazon SNS	Pub/sub — fanout to SQS, Lambda, HTTP(S), email, SMS, mobile push, Kinesis Firehose. FIFO topic for ordered/dedupe. Message filtering, DLQ.
Amazon SQS	Standard (at-least-once) + FIFO (exactly-once, ≤ 3000 msg/s with batching). VisibilityTimeout, DLQ, long-polling.
Amazon EventBridge	Event bus (default, custom, partner), Pipes (source → filter → enrich → target), Scheduler (successor to CloudWatch cron). Schema registry, archive + replay — the DevOps automation backbone.
Amazon AppFlow	Managed connectors for SaaS ⇄ AWS (Salesforce, Zendesk, Slack, Marketo, SAP).
AWS Step Functions	See §8. Orchestrates Lambda + ECS + Batch + EMR + human approval.

Exam cue: "Route events to different targets based on content" → EventBridge rules. "Buffer, decouple, retry failed worker" → SQS + DLQ. "Fanout to many subscribers" → SNS. "Saga / long-running workflow" → Step Functions.

11 · CI/CD Pipeline Services

Service	Role
AWS CodePipeline	Pipeline orchestrator. V2 pipelines add branch filters, triggers on file paths, manual approval, parallel stages, git tag triggers.
AWS CodeBuild	Managed build. `buildspec.yml` phases (install / pre-build / build / post-build). **Reports** surface JUnit/Cucumber. Batch builds + matrix builds. VPC mode for private resources.
AWS CodeDeploy	Deploy to EC2/ASG (in-place or blue/green), Lambda (canary/linear/all-at-once), ECS (blue/green via ALB/NLB target groups). AppSpec defines hooks: BeforeInstall, AfterInstall, ApplicationStart, ValidateService.
AWS CodeArtifact	Managed artefact repository for npm, PyPI, Maven, NuGet, Gradle, Generic. Upstream public repos, cross-account sharing, domain encryption.
Amazon CodeGuru	Reviewer (code quality / security suggestions in PRs; deprecated → Amazon Q Developer), Profiler (runtime performance on EC2/ECS/Lambda).
AWS CodeStar DEPRECATED	End-of-life July 2024 for new projects — consuming projects should migrate to CodeCatalyst or bespoke CodePipeline + CloudFormation. Existing CodeStar projects keep running until sunset.

CodePipeline Action Providers Worth Memorising

- **Source** — S3, CodeStar Connections (GitHub, Bitbucket, GitLab), ECR, CodeCommit.
- **Build / Test** — CodeBuild, Jenkins, third-party.
- **Deploy** — CodeDeploy, CloudFormation (stack / change set / stack set), Elastic Beanstalk, ECS (standard + CodeDeploy), Service Catalog, S3, AppConfig.
- **Approval** — Manual approval action (pauses for human, notifies via SNS).
- **Invoke** — Lambda, Step Functions — for custom gates, integration tests, Slack approvals.

Source-Control Options

- **AWS CodeCommit** — Git-compatible. Was closed to new customers in 2024 then **returned to GA on 2025-11-24**; IAM-based access, CloudWatch Events on repo changes. Existing and new customers supported.
- **CodeStar Connections** — OAuth/IAM link to GitHub, Bitbucket, GitLab. Preferred for teams already on those platforms.

12 · Deployment Strategies Decision Matrix

Strategy	Downtime	Cost	Rollback	Use case
In-place (rolling)	Low	Low	Redeploy previous	Stateless, non-critical, dev/test.
Rolling with additional batch	None	Low-Med	Redeploy previous	EB apps where capacity must not dip.
Immutable	None	Med	Terminate new ASG	Elastic Beanstalk safety-first.
Blue/Green	None	High (2× temp)	Swap back	Zero-downtime, instant rollback. ECS/EC2/Lambda via CodeDeploy; Route 53 DNS flip.
Canary	None	Low	Shift traffic back	Lambda/API GW/ECS — 10% / 50% / 100% gates on CloudWatch alarms.
Linear	None	Low	Shift traffic back	Lambda — +10% every N minutes.

Strategy	Downtime	Cost	Rollback	Use case
All-at-once	Yes	Low	Manual	Hotfix, small blast radius.

Lambda deployment preferences: *Canary10Percent5Minutes* for latency-sensitive APIs; *Linear10PercentEvery1Minute* for steady rollout; *AllAtOnce* only for dev. Always attach a CloudWatch alarm to the alias → auto rollback.

13 · Infrastructure as Code

Tool	When to pick
AWS CloudFormation	Native IaC. *Change Sets* preview, *Drift Detection*, *Nested Stacks*, *Cross-Stack Refs* (Outputs + ImportValue), *StackSets* org-wide. Rollback triggers on CloudWatch alarms. Hooks for pre-deploy validation.
AWS SAM	CFN transform for serverless: compact Lambda + API GW + DynamoDB templates. `sam local invoke` for local test, SAM Pipelines for CI.
AWS CDK	Imperative IaC (TS / JS / Python / Java / C# / Go). Synthesises to CFN. Constructs + Aspects + custom L3 libraries.
AWS Proton	Platform engineering — curated stacks (environment templates + service templates) developers consume. Lifecycle updates on existing deployments.
AWS Service Catalog	Governed self-service — wraps CFN as approved Products with TagOptions + Constraints.
AWS Cloud Development Kit (AWS CDK)	Same as CDK — listed twice in exam guide. Prefer for complex, reusable abstractions.

CloudFormation Patterns Tested Constantly

- **Rollback** — default on failure. Disable with `DisableRollback` for debugging. Policy: `DeletionPolicy: Retain / Snapshot` on stateful (RDS, S3, DynamoDB).
- **UpdatePolicy** — ASG (rolling/replacing update), Lambda alias (CodeDeploy). **CreationPolicy** + `cfn-signal` — wait for success signal from instances.
- **Parameters + conditions** — one template, many environments. Use SSM parameter types (`AWS::SSM::Parameter::Value<String>`) to resolve config at deploy.
- **Custom Resources** (Lambda-backed) — extend CFN for anything not natively supported. Return `PhysicalResourceId`; handle Create/Update/Delete cleanly.
- **Drift Detection** — detect manual console changes. Org-wide pattern: Config rule + EventBridge → auto-remediate by re-applying the stack.
- **StackSets** — service-managed with Organizations auto-propagates to new accounts in target OUs. Deploy baselines (CloudTrail, Config, IAM roles).

14 · Configuration Management & Fleet Ops

AWS Systems Manager — The DevOps Toolbox

Capability	What it does
Fleet Manager	Unified view of managed nodes (EC2, on-prem, edge).
Run Command	Agentless (SSM Agent) remote shell on fleets. IAM-gated, audited via CloudTrail.
State Manager	Declarative desired-state config (Associations) — e.g., install agent on every new instance.

Capability	What it does
Patch Manager	Patch baselines per OS. Maintenance Windows schedule safe patch cycles. Compliance reporting.
Automation	Runbooks (SSM documents) — triage, remediation, chaos. Called from EventBridge, Config rules, Incident Manager.
Session Manager	Shell into instances without SSH / bastion / open ports. Logs to S3 + CloudWatch.
Parameter Store	Config + SecureString secrets (see §3).
AppConfig	Runtime feature flags / config deployment with validators + CloudWatch alarm rollback.
Change Manager	Approval workflow + calendar-aware windows for ops actions.
Incident Manager	On-call rotas, escalation plans, runbook auto-invocation, postmortems.
Inventory	Software + OS inventory for managed nodes; feeds Config.
OpsCenter	OpsItems aggregate issues; integrates with Incident Manager + Security Hub.
Quick Setup	One-click baseline (SSM Agent, Inventory, Patch baseline) across Organizations.

AWS OpsWorks DEPRECATED

- OpsWorks Stacks reached end-of-life **2024-05-26**. No new customers and AWS-operated stacks were terminated.
- OpsWorks for Chef Automate and Puppet Enterprise likewise EOL'd in 2024.
- **Exam answer for "replace OpsWorks":** Systems Manager + custom config mgmt (Chef/Puppet on EC2) or move to containers + ECS/EKS. The exam still tests OpsWorks vocabulary — know the migration direction.

15 · Monitoring, Logging & Observability

Service	Role
Amazon CloudWatch	Metrics (standard + custom, high-res 1s), Alarms (standard, composite, anomaly detection), Dashboards, Metric Streams, Contributor Insights, Synthetics canaries, Evidently feature flags + experiments, ServiceLens + X-Ray maps.
Amazon CloudWatch Logs	Log groups, retention per group, subscription filters → Firehose/Kinesis/Lambda/OpenSearch, Logs Insights (query DSL), Live Tail, cross-account sharing.
Amazon Managed Service for Prometheus	PromQL at scale; scrape targets via ADOT Collector. Plug into Managed Grafana.
Amazon Managed Grafana	Dashboards across CloudWatch, Prometheus, X-Ray, Timestream, Athena, 3rd-party.
AWS X-Ray	Distributed tracing for microservices; sampling rules; service maps. ADOT Collector preferred for OpenTelemetry.
AWS CloudTrail	Management + data events + Insights (anomaly detection). Lake (SQL queries over event history).
Amazon Kinesis Data Streams	Durable event bus, shard-based throughput, < 1 s latency. Consumers via Enhanced Fan-Out (2 MB/s dedicated per consumer).
Amazon Kinesis Data Firehose	Managed delivery to S3 / Redshift / OpenSearch / Splunk / HTTP with buffering + Lambda transform + Parquet/ORC conversion.
Amazon OpenSearch Service	Search + log analytics. UltraWarm + cold tiers, anomaly detection, alerting, SAML SSO.
Amazon Athena	SQL over S3 (CloudTrail, VPC Flow, ALB / CloudFront / WAF logs). Partitioned + compressed for speed and cost.

Service	Role
Amazon QuickSight	BI dashboards on monitoring data; Q for natural-language.
Amazon EMR	Spark / Hive / Presto / Trino on Hadoop. EMR Serverless for on-demand analytics over log lakes.

Alarm Patterns

- **Composite alarms** — combine alarms (AND/OR/NOT) to suppress noise (e.g., only alert when CPU high AND throttles).
- **Anomaly detection** — ML band around expected value. Great for seasonal workloads.
- **Metric math** — derive from multiple metrics in one alarm.
- **Cross-account / cross-Region** dashboards; Metric Streams deliver to Kinesis Firehose → data lake.

16 · Incident & Event Response

Event-Driven Auto-Remediation Pattern

The single pattern that shows up in nearly every incident-response question:

1. **Source** — GuardDuty finding / Config rule NON_COMPLIANT / CloudWatch Alarm / CloudTrail event / Health event.
2. **Router** — EventBridge rule matches on source + detail-type + event pattern.
3. **Action** — SSM Automation document (runbook), Lambda function, Step Functions state machine, Incident Manager response plan.
4. **Notification** — SNS / Chatbot (Slack, Microsoft Teams) with context + runbook link.
5. **Evidence** — OpsCenter OpsItem, Security Hub finding, Incident Manager timeline, Systems Manager Change Calendar.

Additional Services

- **AWS Systems Manager Incident Manager** — on-call schedules, escalation plans, chat channel auto-populated, response plan invokes SSM Automation.
- **AWS Fault Injection Simulator (AWS FIS)** — managed chaos experiments (stop instances, throttle network, add latency, kill pods). Stop conditions on CloudWatch alarms. Integrates with Resilience Hub.
- **AWS Resilience Hub** — score apps against RTO/RPO targets, recommends fixes (backup policies, Multi-AZ, replica counts). Drift detection on resilience posture.
- **AWS Health + EventBridge** — AWS operational events (maintenance, outages). Org-wide Health propagates to Delegated admin account.
- **AWS Backup** — restore testing plans (run scheduled automated restores), vault lock for ransomware protection.
- **AWS Elastic Disaster Recovery (DRS)** — push-button failover / failback for EC2 + on-prem.

17 · Resilient Cloud Solutions

Resilience Choices by RPO / RTO

Pattern	RPO	RTO	Cost	Build-blocks
Backup & Restore	Hours	Hours	$	AWS Backup + CloudFormation to re-launch
Pilot Light	Minutes	10s of min	$$	Core DB replicated, minimal compute; Route 53 + ASG scales out on failover
Warm Standby	Seconds	Minutes	$$$	Full stack running at reduced capacity; DNS failover + scale-up
Multi-site Active/Active	~ 0	~ 0	$$$$	Aurora Global DB + DynamoDB Global Tables + Route 53 / Global Accelerator

Resilience Toolbox

- **Multi-AZ** — default for RDS, Aurora, ElastiCache, EFS, NAT Gateway. Spread ASG min across ≥ 2 AZs.
- **Multi-Region** — Aurora Global DB, DynamoDB Global Tables, S3 CRR with Replication Time Control, Route 53 failover or ARC routing controls, Global Accelerator.
- **Application Recovery Controller (ARC)** — routing controls + readiness checks to safely flip traffic across Regions.
- **AWS Backup + AWS Elastic DR** — cross-Region vaults + continuous replication.
- **Auto Scaling + Health checks** — self-healing compute. Target tracking + predictive scaling.
- **Circuit breakers** — API GW throttles, SQS + DLQ, Step Functions retries with exponential back-off.
- **Chaos testing** — FIS experiments + Resilience Hub recommendations.

18 · Security & Compliance for DevOps

Pipeline Security Controls

- **No long-lived creds** — pipelines use roles assumed via OIDC (GitHub Actions → IAM), or CodePipeline's native role with cross-account role chaining.
- **Signed artefacts** — CodeArtifact + Signer (Lambda, container images). Notation + AWS Signer verification in deployment stage.
- **Secret scanning** — pre-commit + CodeGuru / Amazon Q + third-party. Block merges on findings.
- **SAST / SCA** — CodeBuild step invoking scanners (Inspector for ECR, CodeGuru reviewer, bandit / eslint / semgrep). Fail build on high severity.
- **Encrypt artefacts at rest + in transit** — KMS CMKs for S3 artefact buckets + ECR + CodeArtifact domain.
- **Manual approval gates** — pre-prod SNS approval linked to Change Manager / Incident calendar.
- **Policy as code** — CloudFormation Guard, cfn-lint, OPA. Config Conformance Packs enforce post-deploy.

Key Services Checklist

- IAM + IAM Identity Center, STS, Directory Service, Cognito (see §3).
- KMS, CloudHSM, ACM, Secrets Manager, Parameter Store SecureString (see §3).
- WAF, Shield, Network Firewall, Firewall Manager (see §5).
- GuardDuty, Inspector, Detective, Macie, Security Hub, Audit Manager (see §4).
- **AWS CloudTrail Lake + CloudTrail Insights** for audit SQL + anomaly detection.
- **AWS Config + Conformance Packs** for continuous compliance.

19 · Cost Management for DevOps

- **Cost Explorer** — forecasts, RI/SP utilisation, rightsizing recommendations.
- **AWS Budgets** — cost + usage + RI/SP utilisation + coverage. Actions: stop instances, detach IAM policies, apply SCP when a threshold breaches.
- **AWS Cost Anomaly Detection** — ML alerts on unusual spend. Subscribe via SNS/email.
- **AWS Compute Optimizer** — ML rightsizing for EC2, EBS, Lambda, ECS on Fargate, ASG.
- **Trusted Advisor** — idle load balancers, under-utilised EC2, unassociated EIPs, service limits.
- **Savings Plans / Reserved Instances** via Organizations linked-account sharing.
- **Tagging + AWS Billing Conductor** — chargeback/showback across teams.

20 · CI/CD Reference Architectures

A · Serverless API pipeline (Lambda + API Gateway + DynamoDB)

- **Source** CodeStar Connections → GitHub → main branch trigger.
- **Build** CodeBuild runs `sam build` + unit tests + security scans (bandit, cfn-guard).

- **Package** `sam package` uploads to S3 artefact bucket (KMS-encrypted).
- **Deploy dev** CloudFormation change set → execute. Integration tests in CodeBuild against dev stage URL.
- **Manual approval** via SNS / Slack Chatbot.
- **Deploy prod** Lambda alias + CodeDeploy *Canary10Percent5Minutes* with CloudWatch alarm on 5xx + latency.
- **Observability** X-Ray, CloudWatch dashboard, Synthetics canary smoke test post-deploy.

B · ECS Fargate Blue/Green

- CodePipeline triggers on ECR image push.
- CodeBuild runs `trivy/grype` on image, fails on critical CVE; pushes to ECR prod repo.
- CodeDeploy blue/green: two ALB target groups, traffic shift 10% / 1 min ramp, CloudWatch alarm on 5xx rollback.
- `BeforeAllowTraffic` Lambda hook runs smoke tests; `AfterAllowTraffic` publishes CloudWatch custom metric for deploy success.

C · Multi-account IaC pipeline (CloudFormation StackSets)

- Deployments account hosts CodePipeline + KMS key for artefact encryption.
- Each workload account owns a cross-account role trusted by the deployments account.
- StackSet *service-managed* with Organizations auto-propagates baseline (CloudTrail, Config, GuardDuty, IAM roles).
- Drift detection Config rule + EventBridge → re-apply stack via SSM Automation.

D · EKS GitOps pipeline

- Git repo of manifests; Flux/Argo CD reconciles cluster state.
- CodeBuild builds container + pushes to ECR + updates manifest tag via `git commit` through CodeStar Connection.
- Argo Rollouts performs canary with Prometheus metric analysis; Karpenter provisions nodes on demand.

E · Data pipeline (ingest + analytics)

- Kinesis Data Streams → Firehose → S3 (Parquet, partitioned by date) → Athena / Redshift Spectrum / QuickSight.
- Glue crawler populates catalog; Step Functions orchestrates ETL; Lake Formation governs fine-grained access.
- Athena CTAS + Glue Jobs scheduled by EventBridge Scheduler.

21 · DOP-C02 Scenario → Answer Patterns

Scenario	Answer
Single-command centralised logging across many accounts	Org Trail → S3 in Log Archive + CloudWatch Logs cross-account subscription / Kinesis Data Streams destinations.
Zero-downtime Lambda deploys with automatic rollback	Lambda alias + CodeDeploy *CanaryXPercentYMinutes* + CloudWatch alarm attached to alias.
Auto-remediate public S3 bucket	Config managed rule *s3-bucket-public-read-prohibited* → EventBridge → SSM Automation document blocks public access.
Remove embedded EC2 long-lived keys	Replace with IAM instance profile + Secrets Manager / Parameter Store for app-level secrets.
Patch thousands of Linux + Windows instances monthly with compliance reporting	Systems Manager Patch Manager + Maintenance Windows + patch baselines; compliance dashboard in Systems Manager.
Cross-account CodePipeline deploys to prod	Encrypt artefact bucket with KMS; key policy + S3 bucket policy grant target account. CodePipeline assumes cross-account role to run CloudFormation.
Developers can launch only approved stacks	Service Catalog portfolios with Launch Constraints + TagOptions, granted via IAM Identity Center.

Scenario	Answer
Must detect drift from baseline across all accounts	Config Aggregator + Conformance Packs + CloudFormation StackSets drift detection → SNS.
Disaster recovery for stateful EC2 app with minutes RPO	AWS Elastic Disaster Recovery (DRS) — continuous block-level replication.
Reduce cold-start for latency-critical Lambda	Provisioned concurrency + SnapStart (Java); alias warm-up via Application Auto Scaling.
Detect unusual API calls pointing to compromise	GuardDuty + CloudTrail Insights + Security Hub. Auto-quarantine via SSM Automation.
Route traffic between two Regions with < 30 s failover	Route 53 health checks + failover records; or ARC routing controls; or Global Accelerator for non-HTTP.
Manage feature flags with gradual rollout	AWS AppConfig with deployment strategies + validators + CloudWatch alarm auto-rollback; or CloudWatch Evidently for A/B experiments.
Deploy hundreds of micro-services without long pipelines each	AWS Proton service + environment templates; Copilot for ECS/Fargate; or EKS + GitOps with Flux/Argo.
Company standardises on Kubernetes but wants less ops	EKS with Fargate profiles or Karpenter + managed node groups.
Chaos-test resilience against AZ failure	AWS FIS experiment "stop instances" filtered by AZ + CloudWatch alarm stop condition.
Enforce immutable logs for 7-year retention	S3 Object Lock (Compliance) + SSE-KMS + Glacier Deep Archive lifecycle.
SNS/SQS delivery failed, mustn't drop messages	Redrive policy to DLQ + CloudWatch alarm on *ApproximateNumberOfMessagesVisible* in DLQ.
Single sign-on for engineers across 50 accounts + GitHub	IAM Identity Center with Permission Sets + GitHub via SAML / OIDC from Identity Center.
Central WAF rules to every ALB in every account	AWS Firewall Manager policy targeting Organization scope; or StackSet deploying WAF web ACLs.
Version every configuration change	CloudFormation / CDK with Git source of truth; Config Recorder + CloudTrail for audit.
CodeCommit was deprecated — we need managed Git	**AWS CodeCommit returned to GA 2025-11-24.** Existing customers continue; new customers may now onboard. If the scenario pre-dates the reversal, migrate to GitHub/Bitbucket/GitLab via CodeStar Connections.

22 · High-Signal Service Detail Sheets

A · CloudFormation — Advanced

- **Intrinsic functions** — `!Ref`, `!GetAtt`, `!Sub`, `!Join`, `!ImportValue`, `!If`, `!FindInMap`, `!GetAZs`, `!Cidr`, `!Transform`.
- **Pseudo parameters** — `AWS::AccountId`, `AWS::Region`, `AWS::StackName`, `AWS::Partition`, `AWS::NoValue`.
- **Helper scripts** (on EC2): `cfn-init` (parses Metadata), `cfn-signal` (sends CreationPolicy signal), `cfn-hup` (watches for stack updates), `cfn-get-metadata`.
- **Modules** — packaged resource groupings, registered in the CloudFormation registry.
- **Hooks** — proactive guardrails (cfn-guard-like) evaluated before provisioning; fail the change set if non-compliant.
- **Change sets** — preview resource additions / modifications / replacements / deletions before executing.
- **Stack policies** — JSON blocking updates on specific resources; prevents accidental replacement of stateful resources.

- **Rollback configuration** — auto-rollback on CloudWatch alarm breach during stack update.

B · CodeBuild — Advanced buildspec

- `version: 0.2`, phases *install, pre_build, build, post_build*, *reports*, *artifacts*, *cache* (local + S3).
- **Environment variables** — plaintext, Parameter Store (`parameter-store`), Secrets Manager (`secrets-manager`). Never hard-code.
- **Batch builds** — *build-list*, *build-matrix*, *build-graph* (DAG of dependent builds).
- **Reports** — JUnit, TestNG, Cucumber, Visual Studio test trx, code coverage (JaCoCo, Cobertura, SimpleCov, Clover).
- **Compute types** — BUILD_GENERAL1_SMALL / MEDIUM / LARGE / 2XLARGE; ARM (Graviton); Lambda compute for fast, cold-start-free short builds; GPU for ML.
- **VPC mode** — run builds inside a VPC to access RDS, ElastiCache, private PyPI/npm registries.

C · CodeDeploy — AppSpec lifecycle hooks

Hook	When (EC2/ASG)	Typical use
ApplicationStop	Before download	Graceful shutdown of previous version.
BeforeInstall	Before copying files	Backup, pre-clean, prepare directories.
AfterInstall	After copying files	Compile, configure, migrate DB.
ApplicationStart	Before ELB health check	Start service.
ValidateService	After start, before traffic	Smoke tests, exit non-zero to rollback.
BeforeBlockTraffic	Before deregister	Drain session state.
BeforeAllowTraffic (Lambda/ECS)	After green is ready	Integration tests against green.
AfterAllowTraffic (Lambda/ECS)	After traffic shifted	Smoke tests + notify.

D · Lambda Power Tuning & Cost Patterns

- **Memory = CPU** — increasing memory linearly increases CPU. Use *aws-lambda-power-tuning* (Step Functions) to find the sweet spot.
- **Graviton (arm64)** — ~20% cheaper / ~20% faster than x86 for most Python/Node.
- **Billing** — 1 ms granularity; minimum 1 ms. Provisioned concurrency billed for provisioned time + invocation.
- **Cold-start culprits** — VPC ENI attach (largely mitigated by Hyperplane), large deployment packages, init-time work. Mitigate with provisioned concurrency, SnapStart (Java), container images with lazy init, Lambda layers for shared deps.
- **Concurrency limits** — account = 1000 default; reserved (isolates a function) vs provisioned (pre-warmed).

E · Step Functions — Patterns

- **Task token** — pause state machine waiting for external callback (`SendTaskSuccess`/`SendTaskFailure`).
- **Map state** — fan-out over array; distributed Map processes millions of items with inline or child workflows.
- **Parallel state** — branches run concurrently.
- **Retry + Catch** — exponential back-off with jitter; catch routes to compensating transaction.
- **Express vs Standard** — Express: < 5 min, at-least-once, event-driven streaming; Standard: up to 1 year, exactly-once, durable history.

F · Systems Manager — Managed Node Eligibility

- SSM Agent installed (pre-installed on Amazon Linux 2 / 2023, Ubuntu 18.04+, Windows 2012+).
- IAM instance profile with `AmazonSSMManagedInstanceCore`.
- Network path to SSM endpoints: NAT + public, or VPC endpoints (*ssm*, *ec2messages*, *ssmmessages*).
- Hybrid activation for on-prem / edge nodes — activation code + ID register the node.

G · Testing in a Pipeline

- **Unit** — in CodeBuild; fails build immediately.
- **Integration** — dedicated stage after dev deploy; Newman / Postman, pytest, *sam local invoke*, or ephemeral preview environments (Proton, Copilot).
- **Load** — Locust / k6 in CodeBuild Large; Artillery; target pre-prod.
- **Synthetics** — CloudWatch Synthetics canaries ping after deploy, alarm drives rollback.
- **Chaos** — FIS experiment as a pipeline gate (non-prod) or scheduled in prod with strict stop conditions.
- **Security** — Inspector ECR scan, CodeGuru Reviewer, Amazon Q Developer code suggestions, static scanners. Block on severity.

23 · Decision Trees That Save Exam Points

Event Broker — SNS vs SQS vs EventBridge vs Kinesis

Requirement	Pick
Fan-out one message to many subscribers	SNS (or EventBridge with multi-target rule)
Decouple producer and worker, retry failed work	SQS + DLQ
Route based on content / schema, add partners, cross-account	EventBridge
Ordered, high-throughput stream, multi-consumer replay	Kinesis Data Streams
Managed delivery to S3 / OpenSearch / Redshift / Splunk	Kinesis Data Firehose
Schedule tasks (cron / one-off at scale)	EventBridge Scheduler (CloudWatch Events cron is legacy)

Compute — When to pick what

Workload	Pick
Stateless HTTP API, bursty, short request	Lambda behind API GW / ALB; or App Runner
Long-running HTTP, containerised, AWS-native	ECS + Fargate
Kubernetes mandate, portability	EKS (Fargate or Karpenter-managed EC2)
Full stack auto-provisioned incl. DB, logs	Elastic Beanstalk
Windows / legacy / specialised AMI	EC2 Auto Scaling (with Image Builder)
Batch / big data / fan-out jobs	AWS Batch or Step Functions + Lambda Map
Red Hat OpenShift mandate	ROSA

Secrets — Secrets Manager vs Parameter Store

Attribute	Secrets Manager	Parameter Store
Price per secret/param	$0.40/mo + API calls	Free (Standard)

Attribute	Secrets Manager	Parameter Store
Automatic rotation	Yes (Lambda)	No native rotation
Cross-account	Resource policy	Requires sharing via IAM + KMS
Cross-Region replication	Native	No
Max value size	64 KB	4 KB (Standard) / 8 KB (Advanced)

DR Pattern vs Cost vs RPO/RTO

- If RPO ≤ seconds and RTO ≤ minutes: Warm Standby (full stack + scale up) or Active/Active with Aurora Global DB / DynamoDB Global Tables.
- If RPO is minutes and RTO 10 s of minutes: Pilot Light (DB replicated, compute scaled to zero).
- If RPO/RTO measured in hours and cost is dominant: Backup & Restore with AWS Backup cross-Region copy.
- For on-prem / EC2 workloads: AWS Elastic Disaster Recovery (DRS) gives sub-second RPO from continuous block-level replication.

24 · Multi-Region CI/CD Walkthrough

The reference architecture that pops up under "global SaaS deployment, zero-downtime, Region failover": a deployments account drives a single pipeline that deploys identical stacks in two (or more) Regions, with a gated promotion between them.

1. **Source** — GitHub via CodeStar Connections. On push to `main`, CodePipeline runs.
2. **Build + security scan** — CodeBuild (or Lambda compute type for speed) produces container image + signs with Signer; pushes to ECR in the primary Region with cross-Region replication enabled.
3. **Deploy Region A (primary)** — CodeDeploy blue/green on ECS Fargate with CloudWatch alarm rollback. Synthetics canary smoke tests post-deploy.
4. **Integration / performance gate** — CodeBuild runs k6 load test against A's public endpoint; metric threshold fails the pipeline if breached.
5. **Manual approval** — SNS → Slack Chatbot; reviewer clicks approve in Slack.
6. **Deploy Region B (secondary)** — identical CodeDeploy blue/green; CloudFormation StackSet if infrastructure also needs updating.
7. **Traffic management** — Route 53 failover records (primary+secondary health checks) or Application Recovery Controller routing controls allow an operator to flip traffic in seconds, audited via CloudTrail.
8. **Post-deploy monitoring** — composite CloudWatch alarm (5xx + latency + error rate) attached to ECS service; breach triggers CodeDeploy rollback + EventBridge → SSM runbook to alert on-call via Incident Manager.

25 · Exam Pitfalls & Gotchas

- **"Real-time" log aggregation** — CloudWatch Logs subscription filters → Kinesis Data Streams / Firehose → OpenSearch. Do not answer "CloudWatch agent" when the scenario says "aggregate into a search index."
- **Cross-account CodePipeline** — remember the three things that must be shared: artefact S3 bucket policy, KMS key policy (grant the target account's role), and a role in the target account trusted by the pipeline account.
- **CloudFormation custom resource stuck** — if a Lambda-backed custom resource never calls `SUCCESS`/`FAILED`, the stack hangs an hour before timing out. Always add try/catch that sends `FAILED`.
- **Systems Manager doesn't see my instance** — check: SSM Agent running, IAM profile `AmazonSSMManagedInstanceCore`, network reach to *ssm/ec2messages/ssmmessages* endpoints (VPC endpoints or NAT).
- **CodeDeploy ECS blue/green requires two target groups** on the ALB/NLB, plus a `test listener` for integration tests before traffic shift.
- **Lambda alias traffic shifting** only moves new invocations — long-running Lambdas keep running on the old version. Keep initialization idempotent.

- **Auto Scaling instance refresh** honours minimum healthy %; setting it at 90% with a 4-instance ASG replaces 1 at a time — slow. Drop it to 50% for faster refreshes.
- **EventBridge default bus vs custom bus** — AWS service events arrive on *default*. Custom application events should go on a custom bus so rule scope is obvious and no accidental cross-contamination.
- **S3 Object Lock Compliance mode is truly immutable** — not even root can shorten retention. Governance mode allows `s3:BypassGovernanceRetention`.
- **Config rule vs Security Hub** — use Config for resource-level compliance (rules, auto-remediation); Security Hub for finding aggregation across sources (GuardDuty, Inspector, Macie, 3rd-party) with standards (FSBP, CIS, PCI).
- **Service Catalog Launch Constraint** = role CFN assumes on launch (lets end-user without CFN perms launch stack). TagOptions push required tags at launch.
- **Parameter Store rate limits** — Standard throughput is 40 TPS; switch to Advanced (+ cost) for higher throughput and parameter policies (expiration, notification).
- **CodeBuild report group** must exist (or be auto-created via `reports` block in buildspec) — tests won't appear in the CodeBuild console otherwise.
- **CloudFormation nested stacks** — if a nested stack fails, the root stack rolls back; nested is bounded by 500 resource per stack limit (soft).
- **AWS Backup vs Snapshot on source service** — Backup is the only way to get a cross-Region + cross-account copy with org-wide backup plans. Prefer Backup for anything subject to compliance.
- **DynamoDB PITR** restores to a new table; you then swap via blue/green traffic or app config. Doesn't overwrite the source.

26 · DOP Architecture Deep Dives

Centralised Logging Across 50 Accounts

The question: "corporate audit requires all API, VPC Flow, DNS, and application logs indexed centrally with 7-year retention and tamper-proof storage." The pattern:

- Organization-level CloudTrail trail → S3 in Log Archive account with Object Lock Compliance + SSE-KMS + lifecycle to Glacier Deep Archive after 90 days.
- AWS Config aggregator + Conformance Packs → S3 in Log Archive.
- VPC Flow Logs + Route 53 Resolver query logs to Firehose → S3.
- Application CloudWatch Logs subscribed (subscription filter → Kinesis Data Streams cross-account destination in Log Archive) → Firehose → OpenSearch for search, S3 for retention.
- Athena on S3 for ad-hoc forensics; Security Hub for findings.

Automated Patching with Change Calendar

Patch a 5000-EC2 fleet monthly without breaching change-freeze windows:

- Patch Manager patch baselines: "CriticalSecurityOnly" for prod, "AllApproved" for non-prod.
- Maintenance Windows schedule cron, task = SSM Run Command `AWS-RunPatchBaseline`.
- Systems Manager Change Calendar gates maintenance windows — window won't execute during a closed calendar event.
- Compliance reporting surfaces non-compliant nodes in Systems Manager → OpsCenter OpsItem if SLA breached.

Immutable Pipeline for Regulated Workloads

Compliance (PCI, HIPAA) requires every production change go through an approved pipeline with provenance:

- Source only from protected branch in Git; branch protection rules in GitHub, reviewed by CODEOWNERS.
- CodePipeline uses OIDC (no long-lived secrets) to assume role in Deployments account.
- CodeBuild produces signed container image (Signer) and SBOM (Syft); scan with Inspector.
- Artefacts stored in KMS-encrypted S3 with Object Lock Governance (30 days), replicated cross-Region.
- CloudFormation hooks validate against Config rules pre-deploy (policy-as-code); Change Manager requires two approvers and fires only during a calendar-open window.
- Post-deploy, Audit Manager collects evidence automatically.

Self-healing Microservice Platform

"Something broke at 3 am, who fixes it?" Nobody — the platform does:

- Composite CloudWatch alarm (latency + 5xx + error rate) on the ECS service.
- CodeDeploy alarm rollback triggers automatically on the active deployment.
- If no active deploy, EventBridge rule matches alarm state change → Step Functions playbook: (1) page on-call via Incident Manager, (2) capture diagnostics (SSM AWS-GatherSoftwareInventory, ECS execute-command), (3) attempt remediation (scale out, redeploy last good), (4) if still unhealthy, Route 53 failover to secondary Region via ARC routing control.
- After incident, ChatOps records timeline, OpsCenter tracks action items.

27 · Runbook Library & Observability Playbook

Out-of-the-box SSM Automation Documents Worth Memorising

Document	Use
`AWS-RestartEC2Instance` / `AWS-StopEC2Instance`	Simple fleet actions.
`AWS-RunPatchBaseline`	Patch via Patch Manager.
`AWSConfigRemediation-*`	Large catalog: encrypt unencrypted EBS, enable CloudTrail, restrict public S3, block public RDS, rotate access keys, etc.
`AWS-DisablePublicAccessForSecurityGroup`	Removes 0.0.0.0/0 ingress on specified port.
`AWS-EnableCloudTrail`	Creates + starts a CloudTrail if missing.
`AWSSupport-*`	Triage + diagnostics (CollectTroubleshootingLogs, AnalyzeCloudTrail).

Observability Golden Signals Cheat Table

Signal	CloudWatch metric(s)	Typical alarm
Latency (ALB)	*TargetResponseTime* p95	> 1 s for 5 min
Errors (ALB)	*HTTPCode_Target_5XX_Count*	> 1% of total for 3 periods
Traffic	*RequestCount* / *NewConnectionCount*	Anomaly detection band
Saturation (ASG)	*CPUUtilization*, *GroupInServiceInstances*	Target tracking 60–70%
Lambda errors	*Errors* / *Throttles* / *Duration p99*	> 0 errors for 1 period
DLQ buildup	*ApproximateNumberOfMessagesVisible*	> 0 for 5 min
Storage	*FreeStorageSpace* (RDS/EBS)	< 20% free
DynamoDB	*ThrottledRequests*, *ConsumedCapacity* vs provisioned	Throttles > 0

Incident Manager Response Plan Anatomy

- **Contact / Escalation Plan** — rotas and stages (e.g., primary on-call → secondary → manager after 15 min).
- **Engagements** — SMS, voice, email, Chatbot channel.
- **Response plan** — metric filter or EventBridge pattern → response plan invocation. Runbook attached (SSM Automation).
- **Incident timeline** — auto-captured events from CloudWatch, CloudTrail, ECS, plus manual notes; feeds postmortem template.
- **Chat channel** — AWS Chatbot posts to Slack / Teams with "start incident", "acknowledge", "resolve" buttons.

28 · Well-Architected for DevOps

- **Operational Excellence** — everything is code (IaC, config, runbooks). Make frequent, small, reversible changes. Learn from failure — blameless postmortems feed OpsCenter + Incident Manager.
- **Security** — least privilege, defence-in-depth, audit every action (CloudTrail), automate response (EventBridge + SSM).
- **Reliability** — automatic recovery (ASG, multi-AZ), horizontal scale, test recovery (FIS), RPO/RTO matched to tier.
- **Performance Efficiency** — right-sized compute, managed services where possible, data close to users.
- **Cost Optimization** — pay for what you use (Serverless, Spot, SPs), rightsize continuously, delete unused.
- **Sustainability** — use Graviton / ARM, share instances (containers, Lambda), pick Regions with renewables (`eu-west-1`, `us-west-2`), shift workloads to off-peak via EventBridge Scheduler.

> **The DevOps default answer:** Automate it. If a human touches production, it's not done. Git is the source of truth; pipelines are the only deployment path; EventBridge + SSM Automation is the standard remediation fabric; CloudWatch + alarm-driven rollback is the safety net.

Built by **CloudNinja** · **cloudninja.pro** · Free AWS practice exams & cheat sheets. No signup, no paywall.