

# AWS Practioner Study Guide

Content by Jeanne Boyarsky and Janeice DelVecchio

Note: if you are seeing this reference without reading the blog first, please go back and do that. (<https://www.selikoff.net/2019/01/20/how-i-recommend-studying-for-the-aws-certified-cloud-practitioner-exam/>)

AWS Practioner Study Reference .....	1
Infrastructure .....	2
Pricing.....	2
Support.....	3
Compute.....	3
Networking.....	4
Deploying .....	4
Load balancing/scaling .....	4
Basic Storage .....	5
Advanced Storage/Data.....	5
“Simple” services .....	6
Security .....	7
IAM.....	7
Monitoring.....	8
For Programmers and Dev/Ops.....	8
Pillars of Architecture.....	9
Recovery .....	9
Random other services .....	9
Random other concepts .....	10

## Infrastructure

Region	<ul style="list-style-type: none"><li>• Physical location/geographic area with 2+ availability zones.</li><li>• Minimize latency by deploying to 2+ regions</li></ul>
AZ (Availability Zone)	<ul style="list-style-type: none"><li>• Physically/logically isolated data centers</li><li>• Data provisioned across AZs</li><li>• Not all zones offer all services</li></ul>
Data Center	<ul style="list-style-type: none"><li>• 1+ per AZ</li></ul>
Edge Location	<ul style="list-style-type: none"><li>• Host Cloudfront (Content delivery network) for faster delivery of static content with low latency/high transfer speeds</li><li>• More edge locations than AZs</li><li>• Caches data</li></ul>

## Pricing

Basics	<ul style="list-style-type: none"><li>• Usually no charge for inbound data or data within AWS region</li><li>• Pay for CPU, data storage, outbound data transfer</li><li>• The more you use, the less it costs</li></ul>
On demand	<ul style="list-style-type: none"><li>• Pay as you go</li><li>• Most services pay per second of use</li><li>• Good for short term, spiky or unpredictable use</li></ul>
Reservations	<ul style="list-style-type: none"><li>• Up to 75% less</li><li>• 1-3 year commitment</li><li>• Pay none/partial/all up front</li><li>• Costs less if pay more up front</li><li>• Good for steady state usage</li></ul>
Spot	<ul style="list-style-type: none"><li>• Up to 90% less</li><li>• Pay for unused capacity</li><li>• Unpredictable when runs</li><li>• Ends when complete or price goes above bid</li></ul>
Dedicated instance	<ul style="list-style-type: none"><li>• Pay set hourly price</li><li>• Dedicated hardware for VPC</li><li>• Can use existing software licenses</li></ul>
Free tier	<ul style="list-style-type: none"><li>• Some services free forever – VPC, Elastic Beanstalk, CloudFormation, IAM, Autoscaling, Opsworks, DynamoDB, Glacier, Lambda, Glue, Cognito, SNS, SES, SQS, SWF, Cloudwatch, Xray, Storage Gateway, etc</li><li>• Some services free 12 months – EC2, S3, RDS, CloudFront</li></ul>

## Support

Basic	<ul style="list-style-type: none"> <li>• 7 trusted advisor checks, personal health dashboard, docs/support forms</li> </ul>
Developer	<ul style="list-style-type: none"> <li>• Basic + email support</li> <li>• 1 contact</li> <li>• Response time 24 hours for general, 12 hours for impaired system</li> </ul>
Business	<ul style="list-style-type: none"> <li>• Developer + full trusted advisor checks, phone support</li> <li>• Unlimited contacts</li> <li>• Response time 1 hour for prod down</li> </ul>
Enterprise	<ul style="list-style-type: none"> <li>• Business + senior cloud support engineers</li> <li>• Response time 15 minutes for business critical systems</li> <li>• Includes Well Architected Review by AWS Solution Architects, self packed labs, concierge support team, dedicated technical account manager</li> </ul>
Support forms for	<ul style="list-style-type: none"> <li>• Encountering Abuse (sent to Abuse team)</li> <li>• Increasing limits beyond a point</li> <li>• Penetration testing</li> </ul>
Acceptable Use Policy	<ul style="list-style-type: none"> <li>• What you'd expect; don't do bad things</li> </ul>

## Compute

EC2 (Elastic Compute Cloud)	<ul style="list-style-type: none"> <li>• Virtual server</li> <li>• Proper name is EC2 instances</li> <li>• Pay as you go. Pay for time running</li> <li>• Maintain control</li> <li>• Don't have to provision/maintain server</li> <li>• Assigned both public/private IP</li> <li>• Has instance metadata</li> <li>• Responsible for patching OS</li> </ul>
VPC (Virtual Private Cloud)	<ul style="list-style-type: none"> <li>• Isolate compute resources</li> <li>• Control network config, access, what expose, etc</li> <li>• Can span AZs</li> </ul>
ECS (Elastic Container Service)	<ul style="list-style-type: none"> <li>• Supports Docker containers</li> </ul>
AMI	<ul style="list-style-type: none"> <li>• Amazon Machine Image</li> <li>• Can use variety of preconfigured ones or create own</li> <li>• Specifies type of hardware</li> <li>• Bootable</li> </ul>
Lambda	<ul style="list-style-type: none"> <li>• Serverless</li> <li>• Pay only for compute by fraction of millisecond</li> </ul>

	<ul style="list-style-type: none"> <li>• Ideal for variable/intermittent workloads</li> <li>• Auto-scales</li> <li>• Supports many programming languages</li> <li>• Limited disk space/memory</li> <li>• Must run less than 5 minutes</li> </ul>
--	--

## Networking

IGW (Internet Gateway)	<ul style="list-style-type: none"> <li>• Allows access to internet from VPC</li> </ul>
Subnet	<ul style="list-style-type: none"> <li>• Divides VPC</li> <li>• Public subnets can access internet</li> <li>• Private subnets cannot (by default)</li> <li>• VPC can have multiple subnets</li> </ul>
Route tables	<ul style="list-style-type: none"> <li>• Register traffic leaving subnet</li> </ul>
NAT Gateway	<ul style="list-style-type: none"> <li>• Allows private subnet to access internet</li> </ul>
CIDR (classless interdomain routing)	<ul style="list-style-type: none"> <li>• Internal IP address look like 10.0.0.0/16</li> </ul>
Direct Connect	<ul style="list-style-type: none"> <li>• On premises to VPC connectivity or VPC to VPC connectivity</li> </ul>
PrivateLink	<ul style="list-style-type: none"> <li>• Connects to VPCs through endpoints</li> </ul>
VPC Peering	<ul style="list-style-type: none"> <li>• Connect to VPCs privately</li> </ul>
Route 53	<ul style="list-style-type: none"> <li>• DNS</li> <li>• Geolocation routing</li> <li>• Latency based routing</li> <li>• Defaults to up to 50 domain names</li> <li>• Global service</li> </ul>
Elastic IP	<ul style="list-style-type: none"> <li>• Static IPv4 address</li> <li>• Up to 5 per region</li> <li>• Pay if have more than one and not associated with running instance</li> </ul>

## Deploying

Elastic Beanstalk	<ul style="list-style-type: none"> <li>• PaaS application server</li> <li>• Supplies all infrastructure so can just deploy app</li> </ul>
CloudFormation	<ul style="list-style-type: none"> <li>• Manage/provision collections of servers</li> </ul>

## Load balancing/scaling

Application Load Balancer	<ul style="list-style-type: none"> <li>• HTTP/HTTPS level</li> <li>• Includes HTTPs and WebSockets</li> <li>• Can route by path or hosts</li> </ul>
Network Load Balancer	<ul style="list-style-type: none"> <li>• TCP level</li> </ul>
ELB (Elastic Load Balancer) – classic load balancer	<ul style="list-style-type: none"> <li>• Older loader balancer</li> <li>• Supports both HTTP/TCP levels</li> <li>• Can mix with internal load balancers</li> <li>• Supports single region</li> </ul>

Auto Scaling	<ul style="list-style-type: none"> <li>• Adds more EC2 instances as needed</li> <li>• Specify conditions/policy for when add/remove instances</li> <li>• Create launch config (what create if need new instance), group (constraints on what create) and policy (when to scale)</li> <li>• Limit to 20 EC2 instances per region</li> </ul>
Listener	<ul style="list-style-type: none"> <li>• Checks for connection requests to load balancer</li> </ul>
Target	<ul style="list-style-type: none"> <li>• Destination for traffic based on rules</li> </ul>
Target groups	<ul style="list-style-type: none"> <li>• 1+ targets</li> <li>• Target can be in multiple groups</li> <li>• Can do health check by target group</li> </ul>

### Basic Storage

S3 (Simple Storage Service)	<ul style="list-style-type: none"> <li>• Object data up to 5TB</li> <li>• Can access by URL</li> <li>• API to get data; not associated with specific server</li> <li>• Can access via HTTP/HTTPS</li> <li>• Objects grouped into S3 buckets. Can have up to 100. Can set policies on buckets.</li> <li>• Can replicate across regions</li> <li>• Durability is always 11 nines. Means probability of losing an object.</li> <li>• Availability is 4 nines for standard and 3 nines for SIA (standard infrequent access)</li> </ul>
EBS (Elastic Block Store)	<ul style="list-style-type: none"> <li>• Block storage</li> <li>• Storage for EC2</li> <li>• Persistent data</li> <li>• General Purpose (SSD), Provisioned IOPS (SSD), magnetic</li> <li>• Automatically replicated within AZ. Can copy to other region for recovery</li> <li>• Snapshots are backups</li> </ul>
EFS (Elastic File System)	<ul style="list-style-type: none"> <li>• File storage for EC2</li> </ul>

### Advanced Storage/Data

Aurora	<ul style="list-style-type: none"> <li>• Managed database service</li> <li>• 5x faster than MySQL/Postgres</li> <li>• Faster version of MySQL</li> <li>• Defaults to replicating twice in each of 3 AZs</li> </ul>
RDS (Relational Database Service)	<ul style="list-style-type: none"> <li>• Supports Aurora, MySQL, PostgreSQL, Oracle, MS SQL Server and MariaDB</li> <li>• Set up own IP, subnet, access control, etc</li> </ul>

	<ul style="list-style-type: none"> <li>Automatically generates standby database in another AZ</li> <li>Can create read replicas in different region for all but Oracle and MS SQL Server</li> </ul>
DynamoDB	<ul style="list-style-type: none"> <li>Managed NoSQL service</li> <li>Access by query (key) or scan (non-key attribute)</li> </ul>
RedShift	<ul style="list-style-type: none"> <li>Managed data warehouse service</li> <li>Uses SQL</li> <li>Supports petabytes of data</li> <li>OLAP</li> </ul>
Snowball Edge	<ul style="list-style-type: none"> <li>Physically transport 100TB of data</li> </ul>
Snowball	<ul style="list-style-type: none"> <li>Physically transport petabytes of data</li> </ul>
Snowmobile	<ul style="list-style-type: none"> <li>Physically transport up to 100 petabytes of data</li> </ul>
Glacier	<ul style="list-style-type: none"> <li>Data archiving</li> <li>Each archive up to 40TB</li> <li>Infrequent access</li> <li>Data encrypted by default</li> <li>Archive – document stored</li> <li>Vault – container for storing archives. Has access policy and lock policy (can't alter when locked)</li> <li>Data comes from S3 (via lifecycle policies), SDK, CLI or snowball/snowmobile import</li> <li>Takes minutes or hours to retrieve data depending on cost Bulk/Standard/Expedited</li> </ul>
Transfer Acceleration	<ul style="list-style-type: none"> <li>Transfer files over the internet across long distances with S3 bucket</li> </ul>
DMS (Data Migration Service)	<ul style="list-style-type: none"> <li>Migrate non-AWS database to cloud</li> </ul>
EMR (Elastic map reduce)	<ul style="list-style-type: none"> <li>Hadoop</li> </ul>
Glue	<ul style="list-style-type: none"> <li>ETL (extract load transform)</li> </ul>
Storage Gateway	<ul style="list-style-type: none"> <li>Links to on premises data environment</li> </ul>
Athena	<ul style="list-style-type: none"> <li>Serverless queries</li> </ul>
Kinesis	<ul style="list-style-type: none"> <li>Streaming data</li> </ul>
Kinesis Firehose	<ul style="list-style-type: none"> <li>Data load</li> </ul>
Neptune	<ul style="list-style-type: none"> <li>Graph database</li> </ul>

### “Simple” services

SES (Simple email service)	<ul style="list-style-type: none"> <li>Email</li> </ul>
SNS (Simple Notification Service)	<ul style="list-style-type: none"> <li>Publish messages</li> <li>Supports HTTP/S, Email, Email JSON, SMS, SQS</li> </ul>
SQS (Simple Queue Service)	<ul style="list-style-type: none"> <li>Hosted queue</li> <li>Visible for 12 hours by default</li> </ul>

SWF (Simple Workflow)	<ul style="list-style-type: none"> <li>• Workflow</li> <li>• Activity worker implements a task</li> </ul>
-----------------------	---

## Security

NACL (network access control list)	<ul style="list-style-type: none"> <li>• Stateless</li> <li>• Like passport control</li> <li>• Checks access each time on entry/exit</li> <li>• Optional</li> <li>• At subnet level</li> </ul>
Security Groups	<ul style="list-style-type: none"> <li>• Built in firewall for virtual servers</li> <li>• Set up rules</li> <li>• Can control by protocol/port/IP</li> <li>• By default, controls inbound (blocks all) and outbound traffic (allows all)</li> </ul>
Shield	<ul style="list-style-type: none"> <li>• Protects against DDoS (distributed denial of service)</li> <li>• Free level built into EC 2</li> <li>• Two levels</li> <li>• Advanced level requires Business plan or higher</li> </ul>
WAF (Web Application Firewall)	<ul style="list-style-type: none"> <li>• Blocks common attacks (ex: XSS)</li> <li>• Global service</li> </ul>
Shared responsibility model	<ul style="list-style-type: none"> <li>• Amazon – “of the cloud”</li> <li>• Customer – “in the cloud”</li> </ul>
Guard Duty	<ul style="list-style-type: none"> <li>• Threat detection</li> </ul>

## IAM

IAM (Identity and Access Management)	<ul style="list-style-type: none"> <li>• Control access</li> <li>• Can't recover lost credentials</li> <li>• Allows each user up to two active keys</li> <li>• Global service</li> </ul>
Identities	<ul style="list-style-type: none"> <li>• People/processes/services</li> <li>• Unit of authentication</li> </ul>
Groups	<ul style="list-style-type: none"> <li>• Collections of users</li> </ul>
Root user	<ul style="list-style-type: none"> <li>• Initial user created</li> <li>• Unrestricted access</li> <li>• Only use to create initial other users</li> <li>• Required to use CLI</li> <li>• Recommended to delete access keys</li> </ul>
Role	<ul style="list-style-type: none"> <li>• Identity with permission policies</li> <li>• Does not have own credentials</li> <li>• Used for apps</li> <li>• Used for SSO where authenticated at company</li> </ul>
Temporary credentials	<ul style="list-style-type: none"> <li>• Credentials with restricted permission for a specific task</li> </ul>

Policy	<ul style="list-style-type: none"> <li>Applied to user/role/group to grant permissions</li> </ul>
Access types	<ul style="list-style-type: none"> <li>Programmatic access</li> <li>Management console access</li> </ul>

### Monitoring

TCO (Total Cost of Ownership) Calculator	<ul style="list-style-type: none"> <li>Determine costs before using</li> <li>Don't need to be AWS customer yet</li> <li>Compares on-prem and collocation to pure AWS</li> </ul>
Trusted Advisor	<ul style="list-style-type: none"> <li>Check security, fault tolerance, performance, cost savings.</li> <li>For existing customers</li> <li>Red (immediate action), yellow (investigate), green (good)</li> <li>Can get notification when checks fail</li> <li>Focuses on services</li> </ul>
Cost Explorer	<ul style="list-style-type: none"> <li>Billing visibility for current customers</li> <li>Can see last 13 months of data</li> <li>Forecasts costs for next three months</li> </ul>
Budgets	<ul style="list-style-type: none"> <li>Alerts when costs exceed plan</li> </ul>
Cost and Usage Report	<ul style="list-style-type: none"> <li>Shows costs by category</li> </ul>
CloudTrail	<ul style="list-style-type: none"> <li>Records user activity/API calls</li> </ul>
CloudWatch	<ul style="list-style-type: none"> <li>Monitoring logs</li> <li>Aggregates logs</li> <li>Can set billing alarm</li> <li>Basic and Detailed plans</li> <li>Defaults to 5 minute granularity for basic and 1 minute for detailed</li> </ul>
Inspector	<ul style="list-style-type: none"> <li>Find possible security issues</li> <li>Focuses on S3 level</li> <li>Automated compliance</li> </ul>
Artifact	<ul style="list-style-type: none"> <li>View compliance reports</li> </ul>
Migration Hub	<ul style="list-style-type: none"> <li>Track progress of migrations across AWS and partners</li> </ul>

### For Programmers and Dev/Ops

AWS SDKs	<ul style="list-style-type: none"> <li>APIs</li> </ul>
OpsWorks	<ul style="list-style-type: none"> <li>DevOps platform</li> <li>Uses Chef</li> </ul>
CodeStar	<ul style="list-style-type: none"> <li>UI for Development</li> </ul>
CodeCommit	<ul style="list-style-type: none"> <li>Version control</li> </ul>
CodeDeploy	<ul style="list-style-type: none"> <li>Automated deployment</li> </ul>
CodePipeline	<ul style="list-style-type: none"> <li>Continuous Delivery</li> </ul>



## Pillars of Architecture

Operational Excellence	<ul style="list-style-type: none"><li>• Operations as code</li><li>• Annotate documentation</li><li>• Make frequent, small, reversible changes</li><li>• Refine operations procedures frequently</li><li>• Anticipate failure</li><li>• Learn from operational failures</li></ul>
Security	<ul style="list-style-type: none"><li>• Implement a strong security foundation</li><li>• Enable traceability</li><li>• Apply security at all layers</li><li>• Automate security best practices</li><li>• Protect data in transit and at rest</li><li>• Prepare for security events</li></ul>
Reliability	<ul style="list-style-type: none"><li>• Test recovery procedures</li><li>• Automatically recover from failure</li><li>• Scale horizontally to increase aggregate system availability</li><li>• Stop guessing capacity</li><li>• Manage change in automation</li></ul>
Performance Efficiency	<ul style="list-style-type: none"><li>• Democratize advanced technologies</li><li>• Go global in minutes</li><li>• Use serverless architectures</li><li>• Experiment more often</li><li>• Mechanical sympathy</li></ul>
Cost Optimization	<ul style="list-style-type: none"><li>• Adopt a consumption model</li><li>• Measure overall efficiency</li><li>• Stop spending money on data center operations</li><li>• Analyze and attribute expenditure</li><li>• Use managed services to reduce cost of ownership</li></ul>

## Recovery

Pilot Light	<ul style="list-style-type: none"><li>• Quick recovery option&gt; Minimal version always running</li></ul>
Slowest to fastest	<ul style="list-style-type: none"><li>• Backup &amp; Restore</li><li>• Pilot Light</li><li>• Warm Standby</li><li>• Multi Site</li></ul>
Fault tolerance	<ul style="list-style-type: none"><li>• Stays up even if parts fail</li><li>• More strict than High Availability</li></ul>

## Random other services

CloudFront	<ul style="list-style-type: none"><li>• CDN (content delivery network)</li><li>• Can act as a cache to serve objects from S3</li></ul>
------------	--

	<ul style="list-style-type: none"> <li>• Global service</li> </ul>
Cognito	<ul style="list-style-type: none"> <li>• User sign up/access control</li> </ul>
Config	<ul style="list-style-type: none"> <li>• Configuration history</li> </ul>
Fargate	<ul style="list-style-type: none"> <li>• Run containers</li> </ul>
Macie	<ul style="list-style-type: none"> <li>• Machine learning about security</li> </ul>
QuickSight	<ul style="list-style-type: none"> <li>• Business analytics</li> </ul>
Server Migration Service	<ul style="list-style-type: none"> <li>• Agentless migration from on-prem</li> </ul>
Transcoder	<ul style="list-style-type: none"> <li>• Media conversion</li> </ul>
Workspaces	<ul style="list-style-type: none"> <li>• Virtual desktop</li> </ul>
Xray	<ul style="list-style-type: none"> <li>• Distributed debugging/tracing</li> </ul>

### Random other concepts

Assurance Programs	<ul style="list-style-type: none"> <li>• Include Certification/Attestation and Laws/Regulation/Privacy</li> </ul>
Risk/Compliance Program	<ul style="list-style-type: none"> <li>• Risk Management, Control Environment and Information Security</li> </ul>
Marketplace	<ul style="list-style-type: none"> <li>• Find software solutions</li> </ul>

### Pricing Details

Free	<ul style="list-style-type: none"> <li>• Data in usually free</li> <li>• Data transfer within a region usually free</li> </ul>
EC2	<ul style="list-style-type: none"> <li>• Server time used</li> <li>• Machine (type and config)</li> <li>• # instances</li> <li>• Load balancing and autoscaling</li> <li>• Monitoring level</li> <li>• OS &amp; Software packages</li> </ul>
S3	<ul style="list-style-type: none"> <li>• Storage (amount and class)</li> <li>• Requests (# and types)</li> <li>• Data transfer (out)</li> </ul>
EBS	<ul style="list-style-type: none"> <li>• Volumes (data used)</li> <li>• IO Operations per second</li> <li>• Snapshot (backups)</li> <li>• Data transfer (out)</li> </ul>
RDS	<ul style="list-style-type: none"> <li>• Server time used</li> <li>• Database (type, #)</li> <li>• Storage</li> <li>• # Requests</li> <li>• Data transfer (out)</li> </ul>
Cloudfront	<ul style="list-style-type: none"> <li>• Traffic distribution (regions)</li> <li>• Requests (# and type)</li> <li>• Data transfer (out)</li> </ul>