Resources

Tuesday, November 3, 2020 10:11 AM

Resources:

- https://www.reddit.com/r/AWSCertifications/comments/b2m2vk/anyone_taken_the_aws_certified_cloud_practitioner/
- https://d0.awsstatic.com/whitepapers/aws-overview.pdf
- Free Course
- Udemy: https://www.udemy.com/course/aws-certified-cloud-practitioner-practice-test/

AWS Cloud Practitioner Essentials

From <https://www.aws.training/SessionSearch?pageNumber=1&courseId=17675>

AWS Cloud Practitioner Essentials (Second Edition)

From <https://aws.amazon.com/training/course-descriptions/cloud-practitioner-essentials/>

Link:

https://www.aws.training/Details/Curriculum?id=27076

Exam:
https://home.pearsonvue.com/aws/onvue#additional-information

Note, must unplug other monitors. Use Chrome.

Def be ready 30 mins prior. Turn on lamp so lighting doesn't change.

Q.

I have an appointment scheduled for an online proctored exam. How do I start it?

A.

To launch your exam, sign in to aws.training and select Certification in the top navigation. Next, click the GO TO YOUR ACCOUNT button, followed by Manage Pearson VUE Exams. In the table for Purchased Online Exams, select your online proctored exam. To get started, click the Begin Exam button. You will be able to launch your exam up to 30 minutes prior to your appointment time.

Important: please ensure that you have run the system test on your computer prior to your appointment. If you are more than 15 minutes late for your appointment, or your computer does not meet the system test, you will

not be able to launch your exam, and your exam fees will be forfeited.

From <https://home.pearsonvue.com/aws/onvue#additional-information>

Intro/Management Interfaces

Tuesday, November 3, 2020 10:10 AM

Introduction to AWS Cloud:

Cloud computing - on-demand delivery of IT resources and applications via the internet

More flexible than fixed IT infrastructure

Scalability = ability to resize resources as necessary. Elasticity

Easy to do "comparative testing", comparing different aws services. Experiment quickly w/ low cost and risk. Quick to fail

Reliability - acquire resources to meet demand, mitigate disruptions. Ability of system to recover from service failures.

AWS Regions - where data centers are hosted. Each region is a separate geographic area w/ multiple isolated locations known as availability zones

availability zone = one or more data centers. One region has many availability zones. One availability zone can have one or more data centers.

Fault tolerance - a system can remain up even if some components of system fail. e.g highly available

Data control - customer gets a lot of control. e.g who holds encryption keys. Auditing used to be done manually, aws can be automatic

Data centers secure af, multi factor stuff with restricted access

AWS Management Interfaces:

AWS management console - GUI. Also has a mobile app. Monitor spending here
    Resource groups - drag/drop aws services into little collections. Each user can have their own.

AWS CLI - cli. Automate/repeat AWS services, programming language agnostic. Open source

AWS SDKs (Software Dev Kits) - programming languages. Integrate ur existing apps w/ this bad boy

All 3 have one shared API(?). Yup one shared AWS API

Knowledge Check

Which of the following terms refers to the power to scale computing resources up or down easily?

Elasticity

Core Services (EC2/EBS/S3)

Tuesday, November 3, 2020 1:00 PM

Core Services:

EC2 (Elastic Compute Cloud):

Instances, not servers. Pay as you go.

AMI - Amazon Machine Image

Security Group - set of firewall rules

Default user is "ec2-user"

EBS (Elastic Block Store):

Can do HDD or SSD

More durable than physical SSDs or HDDs because of block-level replication. Could attach a cheap magnetic drive for logs, for examples.

Can easily make snapshots of EBSs and replicate them to other regions. Encrypted in transit.

EBS volume needs to be created on some availability zone as target EC2 instance. After attaching EBS volumes, remember to make the file system… something like mkfs.ext4 /dev/xvdb

Tags are great for tracking billing. For example, can see how much all EBS volumes with the tag "Database" are costing you

Knowledge Check

What are the benefits of using Amazon EC2 instances compared to physical servers in your infastructure?

1. The ability to have different storage requirements
2. Pay only for the capacity you use

Amazon S3 - Simple Storage Service

S3 - fully managed cloud storage service. API to retrieve data. *Not* associated with a particular instance. Can also access it privately through VPC endpoint. By default, data is private.

Create buckets to hold data.

Buckets are key value pairs… Key : Object

Buckets associated with regions like EC2 instances. By default, held in multiple facilities in the region.

Only billed for what you use, don't need to configure throughput.

Can access data with the 3 ways (GUI,CLI,SDK) or directly with REST endpoint.

S3 bucket names need to be DNS compliant and safe for URLs, including object keys.

S3 good for shared locations many instances need to access (EC2 or phyiscal servers). Logs, user-generated media files. Also good for static web hosting, backups, and "staging area for Big Data".

Bucket versioning - keep multiple variants of an object in the same bucket

Global Infrastructure/VPC/Security Groups

Tuesday, November 3, 2020 1:01 PM

AWS Global Infrastructure:

Regions

Organizing level for AWS services.. e.g us-east-1
Can have devs upload to one region, while customers access from another

Resources in one region are not auto replicated to other regions.. Nor are services available in all regions.

Availability Zones

Collections of data centers in one region. Each availability zone is distinct independent physical infrastructure.
Protected from failures in other zones. If one zone goes down, others can handle requests. Best practice is to provision across multiple availability zones

Edge Locations

AWS edge locations host CDNs called Amazon CloudFront. Delivers to customers. Requests routed to nearest edge location so it's faster for end users.

Which components of the AWS infrastructure can be described as multiple, isolated locations within one geographic area?

Availability Zones

Amazon Virtual Private Cloud (VPC):

Amazon VPC- private network within Amazon Cloud that is similar to an on-premise network

Can choose to isolate/expose resources inside their VPC, like firewall rules in a physical networks

Security - ACLs, subnets, routing rules

Lots of AWS services deploy directly to VPC and inherit security settings already set

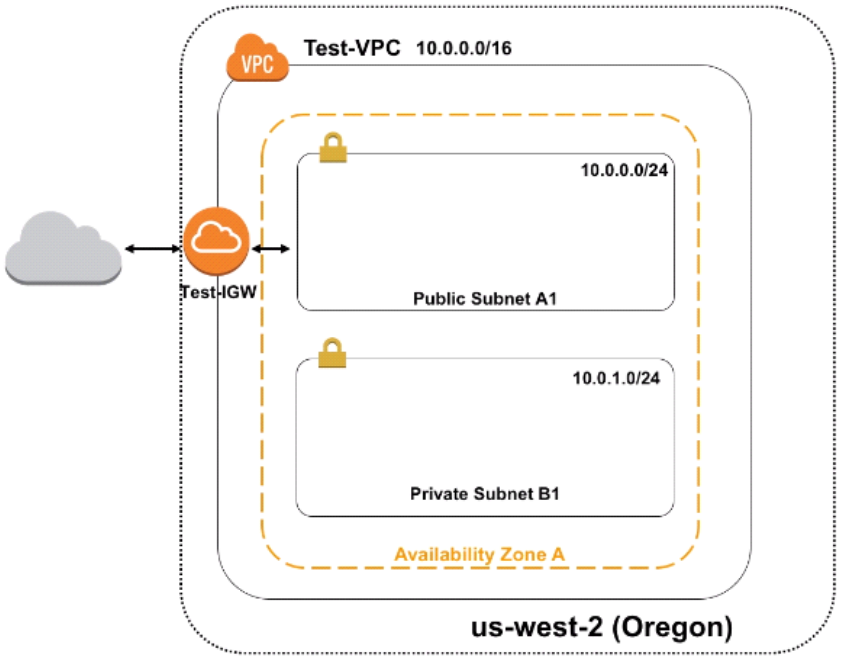
Still has AZ's (Availability Zones) and Regions. One Amazon VPC lives in one region, but can span across multiple AZ's

Subnets are deployed within AZ's, causing VPCs to span across Azs. Public/private - public, direct internet access. Private not direct interact access. Public subnets need to have an Internet Gateway attached to the VPC (and then update route table).

Designing a VPC

1. Select Region
2. Name VPC and assign address - 10.0.0.0/16 (CIDR, 65,000 IP addresses)
3. Create subnet.. Subnet A1. 10.0.0.0/24. This subnet lives in AZ A. Public!

4. Make second subnet. Subnet B1.. 10.0.1.0/24. Private!

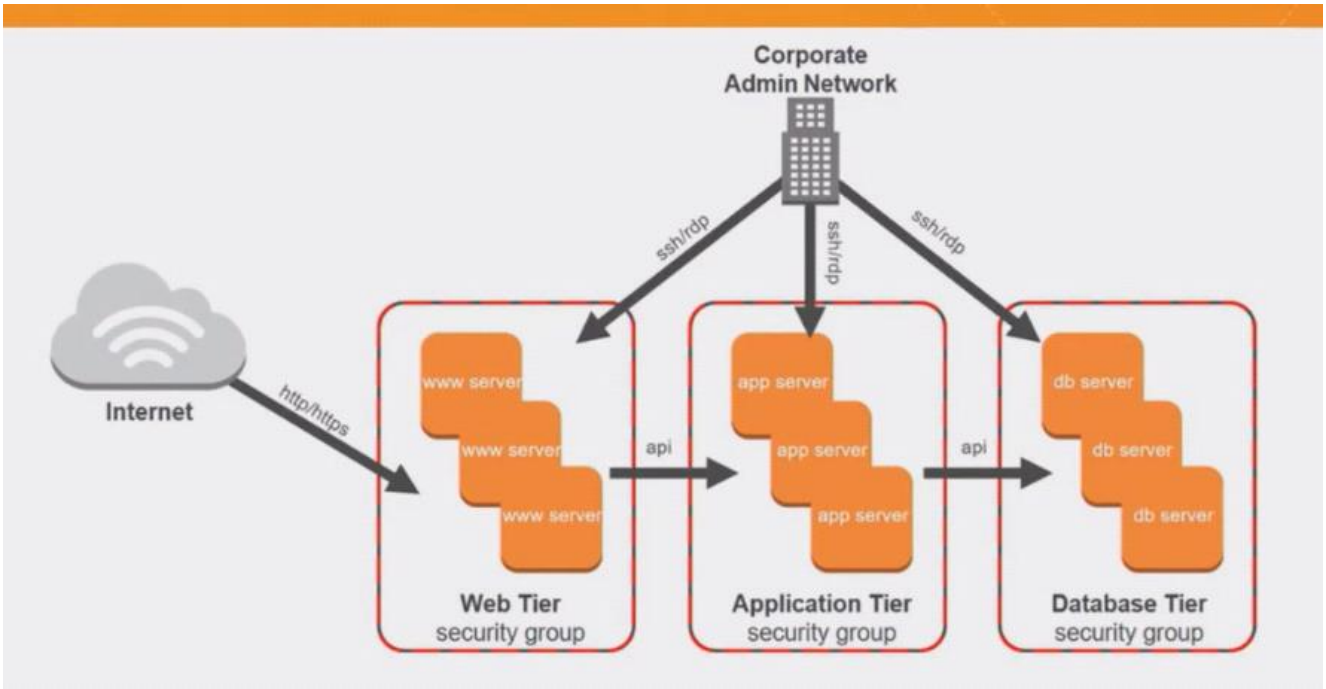


Knowledge Check
Which of the following is true of Amazon VPCs?

You can create many subnets in a VPC, though fewer is recommended to limit complexity

Security Groups:

Security Groups act as built-in firewalls. Control access to instances. Basically a way to filter traffic to instances.



0.0.0.0/0 = entire internet

Security groups are not USER ACCESS, firewall access

Load Balancing/Auto Scaling/Route 53

Tuesday, November 3, 2020 1:02 PM

AWS Integrated Services

Application Load Balancer:

One domain, multiple instances behind it.

Second type of Load Balancer (other is Classic Load Balancer)

- More Supported Protocols - HTTP/S, HTTP/2, WebSockets
- Enhanced metrics - load balance metrics
- Access Logs - see connection details for WebSocket connections
- Health Checks - target/application health

Can do routing, path/host-based. Native IPV6. AWS WAF (Web Application Firewall)

Path-based: rules that forward requests to different target groups
Host-based: rules that forward requests to target groups BASED on hostname (basically, multiple domains under the same load balancer)

Dynamic ports: ECS (Containers) expose dynamic ports .. Idk

Deletion Protection/Request Tracing: track HTTP requests from clients to target

Why use?

Can use containers to host microservices, then route to those containers from the application load balancer.

Route -> Same Instance (different path based on port.. If containers listening on different ports, can route to them)

Listener: checks for connection requests, based on protocol/port
Target: destination for traffic based on listener rules
Target Group: routing requests to one or more targets with protocol and port number specified. Health checks can be on a target-group basis

Targets registered to load balancer via target groups. Targets can be members of *multiple* target groups.

Auto Scaling:

Removes the guesswork from determining the correct number of Amazon EC2 instances to handle the load for your application.

CloudWatch monitors load but does NOT add/remove instances for you

1. How can I ensure my workload has enough EC2 resources to meet fluctuating requirements?

Scalability

2. How do I automate that scaling?

Automation

Scales based on metrics *you* define (e.g CPU utilization over 80%, or a schedule)

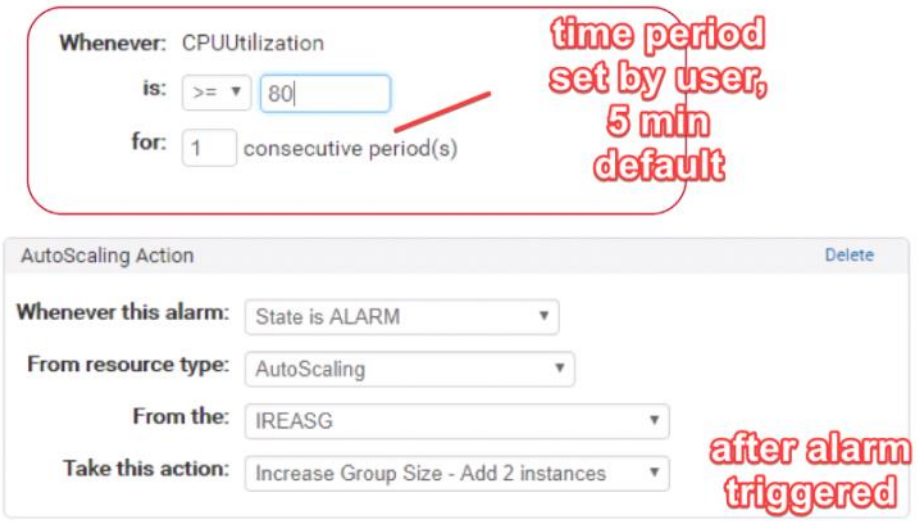
Scaling out: adding instances
Scaling in: terminating instances

How to Auto Scale:

1. Launch configuration - AMI (Machine Image), Instance Type (SSD vs HDD, CPU), Security Groups (Firewalls), Roles (what it can and cannot do)
2. Auto Scaling Group - VPC/Subnets (which VPC), Load Balancer (which Balancer), Minimum/Maximum Instances (e.g 2-8, range), Desired Capacity (number you wish to start with)
3. Auto Scaling Policy (when to launch) - Scheduled, On-demand, Scale-out/in policy

Best practice to have one scale-out/in policy

Scenario: Setup CloudWatch alarm (cpu at 90%), triggers auto scaling event (scale out/in) instances



Target Value:

Something like average CPU utilization 60%. Tries to keep total CPU utilization around that.

Amazon Route 53:

DNS services to help route users to endpoints. Global and highly available. IPv4/IPv6.. Integration w/ other AWS stuff

Users -> Route 53 -> Application

Route 53 > Own DNS server

User requests example.com -> ISP's DNS -> Route 53 (then translates DNS to IP and gives to user)

1. Created Hosted Zone (DNS Data kept here)
2. Give FQDN.. Can also buy from Route 53
3. Hosted zone contains record sets.. DNS translations.. A/PTR/MX, etc

Internal DNS zone:

Application components talking to each other

External:

User -> Website

Basically a fancy DNS settings page. Propagation seems fast as hell. Shortens distances between DNS server and customer.. Dynamic switching based on customer location. Cool.

Lots of different routing options like round robin, geo-location…

Knowledge Check:

You have an application composed of individual services. You need to route a request to a service based on the content of the request. Which service should you use?

Elastic Load Balancing

(Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions.)

RDS, Lambda, Beanstalk

Tuesday, November 3, 2020 4:23 PM

AWS Integrated Services

Relational Database Services:

Standalone/own RDS:
Need to maintain server, backups, patches, security, os install.. Lots of stuff

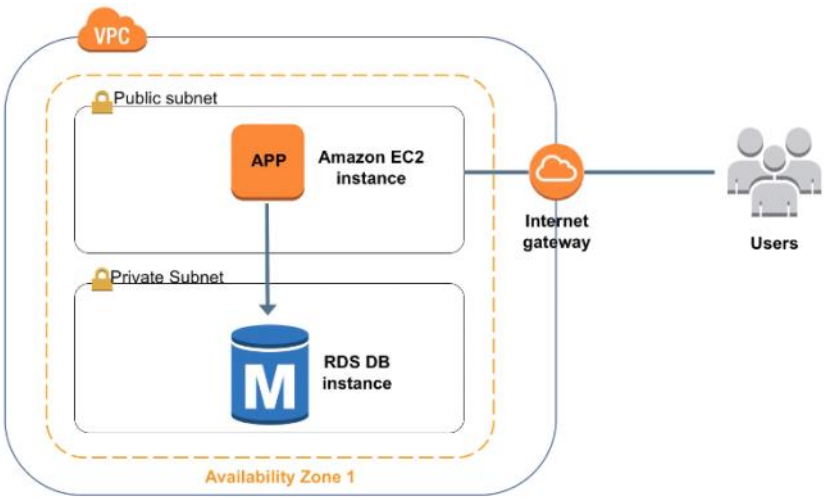
Instead, AWS RDS:
Automates time consuming admin tasks, sets up and operates database. Lets admin focus on the app, instead of the backend.
Manages:

- OS Install
- Database software install
- Backups
- High Availability/Scaling
- Physical stuff, power

Reduces cost somehow.

DB Instance:
Isolated instance that can have multiple DBs. Different types of storage.. Magnetic, SSD, IOPS?

Specify which DB engine… MySQL, SQL Server, Postgres.. Etc



Amazon VPC here. Subnets are associated with a SINGLE availability zone. When choosing subnets, you're also choosing your AZ/Physical Location.

Amazon RDS *automatically* gives you a second AV with replication. Holy shit that's cool. In the same VPC too.

If master DB instance fails, Amazon RDS brings up the second DB. User doesn't need to change anything

for failover. Updates to source DB automatically copied to backup. The second DB can help reduce load, route READ queries to the backup.

Good for web/mobile apps, e-commerce, mobile/online games.
Web/mobile - high throughput
E-commerce - low cost
Mobile/games - scalability

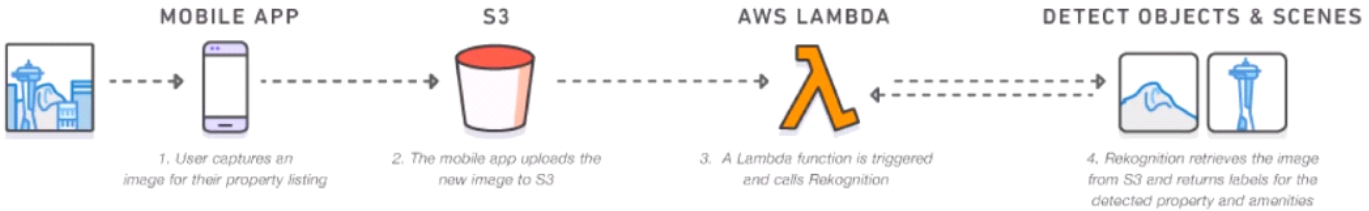
AWS Lambda:

Event driven serverless compute service.

Lambda - compute service that lets you run code without provisioning services. Executes code only when needed and scales.

Only pay for compute you use. Don't pay for when it's not running. Lambda supports programming languages.

Event-driven: run code when stuff changes, like S3 bucket adds or HTTP requests.



5 min is max timeout. Lambda needs a trigger.. E.g when to trigger function. Can use Cloudwatch. DON'T have to program that which is nice.

Example use case: resize images based on user's platform for thumbails. Scales up hella high.

Basically does grunt work. Good for websites/mobile apps and backends. "Connective Tissue"

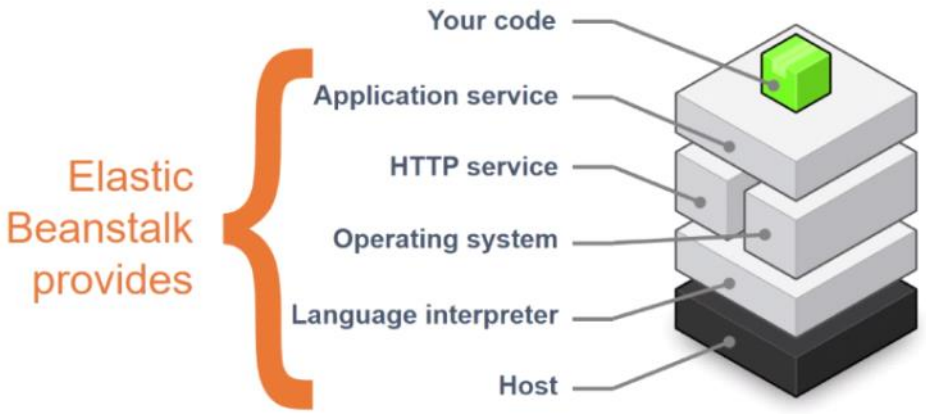
AWS Elastic Beanstalk:

Good for quickly getting an application into the cloud.

Platform As a Service (PaaS). Just put code up. *Quick* deployment. Don't worry about managing the whole system, but can do it if you want.

Choose instance type, database, and auto scaling.

Then, update app, access logs, and enable https on a load balancer.



Just provide the code.

Two types of environment tier:

1. Web Server - serve HTTP requests.. Website/webapp/web API
2. Worker environment - long-running workloads on demand, or scheduled tasks.

Knowledge check:

What is the first step in getting started with AWS Lambda?

Upload your code.

NOT Provision EC2 instances. All you do is upload code.

Elastic Beanstalk vs Lambda…

Beanstalk: quickly deploy and manage apps in AWS Cloud
Lambda: Automatically run code *in response to* change in AWS integrated service.. Like s3 bucket upload

SNS/CloudWatch/CloudFront/CloudFormation

Wednesday, November 4, 2020 12:02 PM

AWS Integrated Services

AWS Simple Notification Service (SNS):

Flexible, fully managed pub/sub messaging and mobile communication service.

Pub/sub (publish/subscribe) - any message published to a topic is received by ALL subscribers to the topic

Coordinates delivery of messages to endpoints and clients.

Decouple/scale microservices

Topics have security - some users can only send, some can only subscribe (receive). Sort of like the AWX groups of sending emails. A topic subscription endpoint as an email will send a confirmation email so you don't spam someone.

Subscriber = receiver of notification. Publisher = who calls/generates the notification.

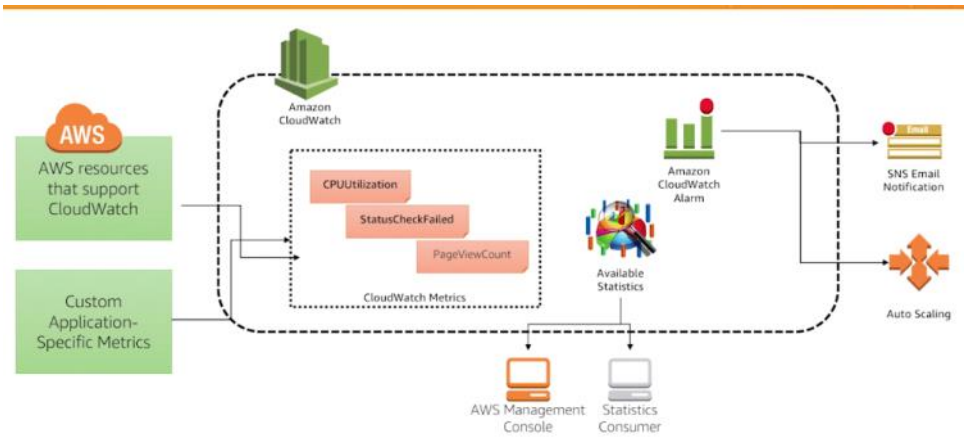
To actually send notifications, go to S3 Bucket, add new "Event" (e.g everytime objected created/deleted from bucket), send information to SNS Topic "SNSDemo".

Amazon CloudWatch:

CloudWatch monitors your AWS resources and applications you run on AWS

I assume uses SNS for notifications

Can collect/track metrics (CPU utilization, Disk I/O), monitor logs… set alarms and react to changes *automatically*



Use cases:

- Respond to state changes in AWS
- Invoke Lambda functions to update DNS when a new EC2 instance comes online
- Take snapshots of EBS

CloudWatch

1. Metrics - data about performance of systems. Time ordered.. Some stuff gives you free metrics (EC2). Can also make your own metrics.
2. Alarms - watch a *single* metric. Can perform one or more actions based on thresholds of metric. Can do auto scaling, EC2 stuff, notifications to SNS

Recall: scale out, more stuff. Scale in, less stuff.

3. Events: real-time stream of system events. Aware of operational changes and take corrective actions. Can also schedule actions that self trigger.. Cron.

Example: Detect/Revoke Unintended IAM Access. If IAM user does some sktechy API call, revoke their access.

4. Logs - monitor systems/application logs, then do something if you see a certain phrase.
5. Store/Monitor logs - metrics can be stored for a while as CloudWatch Logs. Admins can review directly.. Stored in S3 not in CloudWatch.
6. Dashboard - get a cool customizable home page to monitor stuff. Create dashboards with CLI, console, *PutDashboard* API

Create a CloudWatch Event Rule:

1. Select source. Can be a pattern or schedule (schedule can be cron).
2. Log state of an EC2 Instance (can use Lambda, Cloudwatch will run Lambda when EC2 instance made)

Cloudwatch Alarm:

1. EC2 Instances -> Actions
2. Create alarm in EC2 menu. Can do stuff like CPU utilization here.

Dashboard is like Libre and other stuff.

Basically, CloudWatch can do Events (corrective actions) or Alarms (auto scaling/SNS), and display cool stats.

Amazon CloudFront:

CDN - don't confuse with CloudFlare, lol

More than 80 edge locations. Deliver content to users with low latency.

If ur app lives in Singapore, but users in New York, cache things in New York for users.

Integrated into AWS WAF (Web App Firewall), Cert Manager, Route 53, S3, etc.

CDN Types:

- Web (Not Video Streaming.. Doesn't have to be static. Can be dynamic)
- RTMP (Video Streaming)

Origin - S3 Bucket, Any App. Need one origin but can edit later. Can have *multiple* origins that have multiple behaviors.

Use S3 bucket as origin, for requests that are yourapp.com/images

Can automate new CDNs for new environments

Use Cases:

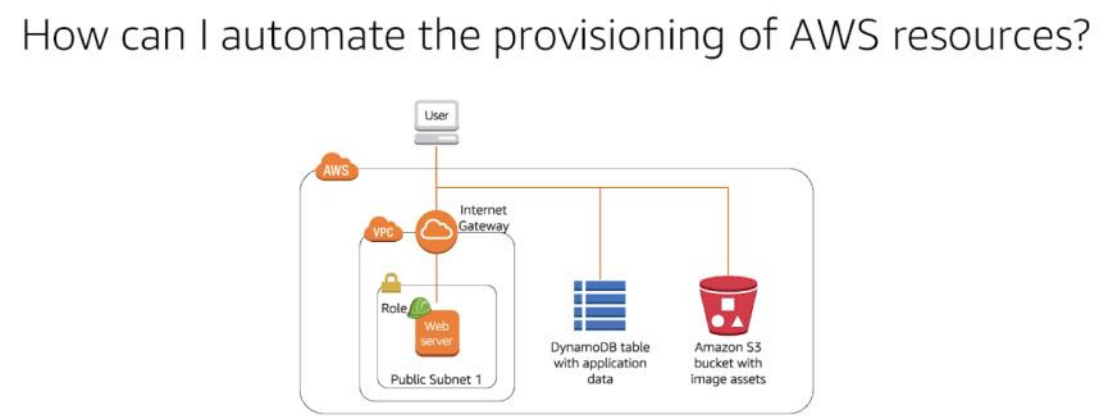
- Static Asset Caching
- Live/On-Demand Video Streaming
- Security/DDoS
- Dynamic Content
- API Acceleration (?)
- Software Distribution (Downloads)

Notice that it will be *cached* content, not like you're proxying from Seattle -> Singapore. Access Logs on the original web server will not show the cached content pulls.

AWS CloudFormation:

CloudFormation *simplifies* the task of *repeatedly* and *predictability* creating groups of related resources that power your applications.

Automating resource provisioning



Basically, how do I automate setting this up, or easily duplicate this setup?

CloudFormation creates, updates and deletes resources in sets (known as stacks)

CloudFormation reads *template files* (yaml/json). Then it makes the resources listed in the template files, known as a stack.

Stacks are units of deployments. Delete a stack, and every resource in that stack is gone.

So, create separate templates for like DB stuff, security, etc. Templates self-documenting like ansible.

See, template is GUI stuff with ansible key-value like pairs.

Can control *order* of resources with "DependsOn". Templates can even make multiple environments with dependencies. e.g, parameter 1 = production, parameter 2 = dev

CloudFormation is *Infrastructure as Code*

Can set permissions on who can process templates

There's a GUI CloudFormation Designer (like MySQL Workbench) if you don't want to do the yaml. It'll give you a template file.

Note that VPC is under EC2. Starting with a VPC seems like a good idea. Designer will throw the template in an S3 bucket for you.

Knowledge Check

Which of the following statements best describes Amazon CloudFront?

Speeds up the delivery of your content to viewers across the globe

AWS Architecture - HA/FT

Wednesday, November 4, 2020 1:33 PM

AWS Well-Architected Framework

Here to:

- Assess/improve architectures
- See how design decisions impact business
- Learn the 5 pillars and design principles

5 Pillars:

- Security
- Reliability
- Performance efficiency
- Cost optimization
- Operational excellence

Security:

Risk assessment

IAM: identity/access management - any authorized and authenticated users can access resources
Detective controls - identify potential security incidents (log analyzing, auditing)
Infrastructure protection - systems/services are protected against unintended access (e.g patching, WAF, firewalls)
Data protection - encryption, data classification, backups, replication/recovery
Incident response - mitigate security incidents

Design principles for Security:

- Security at *all* layers - at perimeter, and *within*, between resources
- Enable traceability - logging/auditing all actions
- Apply POLP (least privilege) - least amount of privilege necessary to do task
- Focus on securing system - shared responsibility (AWS handles secure infrastructure, customer secure virtual stuff)
- Automate - automate best practices, can scale rapidly.. Save hardened image of sever, then can use same hardened image for instances. Automate response to security events too

Reliability:

- Recover from issues or failures - mitigate problems
- Apply best practices in..
    - Foundations - well-planned foundation that can handle changes and heal itself
    - Change management - able to adjust to change quickly and reliably
    - Failure management - automation w/ monitoring to replace failed systems then troubleshoot them after
- Anticipate, respond, prevent failures

Design principles for Reliability:

- Test recovery procedures - simulate failures then test (chaos monkey)
- Automatically recover - trigger automated responses when thresholds breached

- Scale horizontally - increase aggregate system availability.. Better than 1 large monolithic resource
- Stop guessing capacity - can monitor demand and react to it
- Manage change in automation - only need to change automation system, not individual systems

Performance Efficiency:

- Select (selection) Customized solutions - right tool for right job
- Review to continually innovate - take advantage of new tech and approaches.. Could help performance
- Monitor AWS services - monitor performance to remediate issues (CloudWatch, Kinesis, SQS, Lambda)
- Consider trade-offs - e.g, trade consistency, durability, space vs time, latency. Idk

Design principles for Performance Efficiency:

- Democratize advanced technologies - push shit that's hard to do into AWSes domain. IT learn just gets the benefit without knowing how to do it
- Go global fast - multiple regions/AZ (CloudFront)
- Serverless architecture - remove need to run servers (Lambda)
- Experiment more often - fail fast. Test often
- Mechanical sympathy - use technological approach to meet goals? Wtf?

Cost Optimization:

Maximize ROI, best outcome at lowest price

- Use cost-effective resources - cost-optimization
- Match supply with demand - scale-in/scale-out.. Elasticity.. Auto scale
- Expenditure awareness - fully aware of what spending and cost drivers are happening.. Break down current costs and predict future costs
- Optimize over time - improve architecture from data collected on AWS

Design principles for Cost Optimization:

- Adopt a *consumption model* - pay only for what you use
- Measure overall efficiency - measure business output of systems
- Reduce spending on data center operations - stop spending $ on physical data center stuff. A/C, racking/stacking servers
- Analyze/attribute expenses - identify system costs easily
- Use managed services - reduce cost of ownership, don't have to maintain simple severs like SMTP, DNS

Operational Excellence Pillar:

- Manage and automate changes
- Respond to events
- Define standards

No more willy-nilly (ad-hoc) sysadmin shit.

Knowledge Check

Which of the following is NOT a pillar of the AWS Well-Architected Framework?

Persistence

Remember SRPCO

Fault Tolerance and High Availability:

Fault tolerance: ability of a system to remain operational with some components failed
High availability: the entire system is always functional and available, downtime minimized *without human interaction*

Little upfront investment and human interaction

On-prem: $$, good for mission-critical applications
AWS: multiple servers, AZ, regions, fault-tolerant services

Tools for High Availability:

- Elastic load balancers (ELB) - distributes traffic (load) among instances. Sends metric to CloudWatch. (e.g, can recognize unhealthy EC2 instances)
- Elastic IP addresses - static IP addresses that mask failures. Can keep accessing applications if an instance fails. Dynamic.
- Route 53 - Authoritative DNS. Designed for High Availability.. Latency-based routing, DNS failover
- Auto Scaling - terminates/launches instances based on conditions (like customer demand) scale-in/scale-out
- CloudWatch -used with *auto scaling*. Stat gathered system. Tracks metrics of AWS infrastructure. Can also use custom metrics

Tools for Fault Tolerance:

- Amazon Simple Queue Service (SQS) - backbone of fault tolerance. Distributed messaging system.. Queue is always available (Wtf?).. Send/store/receive data between software components. Maybe something like Capital One queuing up payments for credit cards?
- Amazon Simple Storage Service (S3 buckets) - fault tolerant data storage. Redundantly stores data
- Amazon Relational Database Service - operate/scale relational databases. Automated backups and snapshots.. Also multiple AZ deployments

Web Hosting:

Traditional web hosting: infrastructure, architectural issue, cost. Responds *slow*
AWS: solves fucking everything, apparently. Responds *quickly*

Handle traffic peaks easily and cheaply. Basically physical vs cloud stuff

Can get testing fleets only when you need them. Nothing sitting around

Security - Shared Responsibility/IAM

Thursday, November 5, 2020 11:03 AM

Introduction to AWS Security:

AWS likes security! No shit!

AWS innovates rapidly unlike us peasants.

Cloud-based governance:
- Lower cost of entry
- Improved agility

Can inherit many, existing AWS security controls. Own your own compliance programs, strengthened by AWS.

Network security:
- Built-in firewalls
- Encryption in transit (TLS)
- Private/dedicated connections (from office/on-prem)
- DDOS mitigation - CDN/Auto-Scale

Move fast and with best practices.

Track/manage changes to resource over time.

Encryption:
- EBS, S3, Glacier, Oracle, Redshift all have encryption
- Flexible key management options - maintain your own keys, or let AWS do it
- Hardware-based cryptographic keys
    - AWS CloudHSM (Hardware Security Modules) - storage cryptographic keys, process SSL/TLS

Access Control:
- IAM - individual user account permissions
- MFA - can also use hardware-based auth, e.g yubikey
- Integrate corporate directories - API
- Amazon Cognito - signup/user registration in apps
- AWS SSO

Can see lots of stuff about API keys (who called and when).. Log aggregations.. Alert notifications

AWS Marketplace:
- Secure store to sell software that runs on AWS

Shared Responsibility Model:

Who is responsible for maintaining secure on your AWS app?

You and AWS! 100% responsibility between magic line

The Shared Responsibility Model: (who is responsible for what)

- **(USER)** User Data - no idea about user data held in apps
- **(USER)** Application - customer chooses apps
- **(USER)** Guest OS - customer chooses OS

~~~~~~

(magic dividing line, AWS can't see anything above this. Can't read any of this if they wanted to, even for marketing)

- **(AWS)** Hypervisor - zen-based.
- **(AWS)** Network - networking protocols, VPC can work at scale. They don't tell customer how it works but they get audited
- **(AWS)** Physical - iron/concrete/barbed wire. No tours!

Identity and Access Management (IAM):

Permission management.

Authentication:

> *User:* **permanent** named operator. Could be human or machine. Creds are permanent and stay with that named user until forced rotation (access key, user/pass).

> *Group:* collection of users. Many to Many relationship.

> **Role**: NOT PERMISSIONS. Role is an authentication method. Role is an operator - could be human or machine, but the credentials are **temporary**.

Everything in AWS is an API. To execute API, need to:

1. Authenticate (ROLE)
2. Authorize (Policy Document)

Policy Document: json, attaches to either user, group, or role. Lists API(s) that you whitelist for which resources. Can do stuff like only allowing from home, VPN, world. Certain times of day.

API call:
Example - operator wants to put object into S3 bucket. API call.

Execute API call and present set of credentials. User/pass/key… results in API execution statement.

1. The IAM engine first looks at credentials. Validates they're active, and yes, that's the role/user/group. Agree you are the operator you claim to be.

Authorization:

2. THEN, look at policy document. Is the action you're doing authorized by any policy documents? If so, great. Then can execute API call.
~~~~~~

a. Explicit deny - overrides any allow statement. Permanently prevent certain actions. For example, never want some intern to stop/terminate all production.
b. No allow is an implicit deny.

Lets say credentials were compromised. If you use policy documents attached to user, and NOT root, security manager can execute 1 API statement to REMOVE all policy documents in one action. Hacker then tries to delete, can't because IAM engine authenticates hacked user, but can't authenticate w/ policy documents. So doesn't delete. And also, CloudTrail will track the attempt at mass deletion.

Knowledge Check

Your web application requires temporary authorization to use AWS services. Which IAM entity should be used?

Role

Role temporary, user permanent

Security - Inspector, Shield, Compliance

Thursday, November 5, 2020 11:54 AM

Amazon Inspector

Help improve security/compliance of AWS apps. Tools to secure IT infrastructure are complex/$$$. Companies that do manual inspections can miss stuff and takes time.

AWS Inspector assesses apps for vulnerabilities and makes detailed reports.

Can integrate security into DevOps pipelines and stuff. That's cool. Helps w/ development by taking care of security proactively. Thus, streamlines security compliance.

Can access Amazon Inspector via the Inspector Console, SDKs, HTTPS API, CLI

Has built-in rules… like checking for remote root login, or vulnerable software versions installed.

More or less, offloads security assessments so user can focus on complex security issues.

AWS Shield

DDoS protection service.

DDoS application layer attack - sometimes, DDoS's tries to attack how sites communicate with other sites. Application layer usually low throughput so this can be successful.

WAFs can block application layer DDoSes.

Normally, hard to mitigate DDoSes… complex/$$$ to setup. Bandwidth limitations too. May require manual intervention.

As you can guess, AWS Shield is automated and can scale up easier. Shield is faster than silly humans.

AWS Shield Standard: auto protection, free, all customers

- Any AWS resource in all regions
- Self-service - traffic inspection.. Algorithms…

AWS Shield Advanced: 24/7 DDoS response team, protects against large attacks

- 24/7 access to DDoS response team (DRT).
- Post-attack analysis is pretty cool.
- Larger DDoS attack prevention (rate-based blacklisting)
- *Application-layer attacks* on Route 53, CloudFront, ELB, Elastic IP
- DDoS cost protection - protects AWS bill from usage spikes from DDoS

Protecting Route 53:

Shield Standard - infrastructure layer DNS attacks (reflection attacks/SYN floods)
Shield Advanced - visibility into attacks, DRT team

CloudFront (CDN) or Application Load Balancer:
99% of attacks detected by shield here and mitigated in one second. L3/L4

Shield Standard - scrubs bad traffic, always-on
Shield Advanced - same as always, DRT, application layer protection

Non-TCP apps (UDP/SIP) | Elastic IP address, direct EC2 instances:

Shield Standard: normal stuff
Shield Advanced: additional bandwidth

Security Compliance:

Recall the shared responsibility model.

AWS shares their security info by getting auditing, compliance reports, and pushing some of their practices.

3rd party, independent auditing. They'll provide whitepapers and guidelines to help meet security compliance stuff.

AWS helps customers:

- Document a complete control and governance framework
- Meet standards like HPPA

1. Risk management
    a. Business plan that includes risk management, re-evaluated biannually
    b. Process - identify risks, implement measures, assess risk
    c. CIA - Confidentiality, integrity, availability
    d. AWS Security scans stuff for vulnerabilities and informs customers. Doesn't replace customer scans to meet compliance requirements.
2. Control environment
    a. Processes to secure delivery of AWS offerings
    b. Integrates cloud-specific controls
    c. Leading industry practices
3. Information Security
    a. CIA - see above.
    b. Publishes security whitepaper

Strong approach looks like:

- Review trusted information and document compliance requirements
- Design/implement objectives to meet requirements
- Identify/document controls owned by outside parties
- Verify all control objectives are met and operating effectively

Helps customers understand robust controls, establish/operate in an AWS security control environment.

Which of the following is a managed DDoS protection service?

AWS Shield

NOT Amazon Inspector - looks for vulnerabilities (e.g insightVM)
NOT AWS Trusted Advisor - helps follow best practices

Pricing/Support - Trusted Advisor, Support Plans

Friday, November 6, 2020 12:47 PM

Fundamentals of Pricing

Only pay for what you need when you use them, no licensing or long term contracts

Pay less the more you use it - less you use it, less per unit

Adjust infrastructure by need, not by forecasting. Pay as you go model

Reversed capacity (EC2/RDS):
Save up to 75% over on-demand. Larger the payment you make upfront, the greater the discount.

Reserved instances

- AURI (All up-front) - greatest discount, most $ upfront
- PURI (Partial up-front) - lower discounts, but less $ upfront
- NURI (No upfront) - small discount, no $ upfront

Easier to manage budgets with more upfront. The more data throughput you use, the less you pay per gigabyte *out.* Data *in* is free. E.g EC2

Free tier for a year w/ EC2. EBS, ELB, AWS Data Transfer also have free tier.

Multiple AWS accounts? Combine them with consolidated billings, can upgrade tiers to pay less per X/unit/gigabyte.

Pricing Details

3 Cost fundamentals:

- Compute
- Storage
- Data transfer *out* (inbound always free)
    - Aggregated from all cool services, then charged at outbound data transfer rate.

EC2:

Web service that provides resizable compute capacity in cloud.
Clock hours of server time! Resources incur charges when they run. Pay from launch to termination.
Reversed instances, make up-front payment instance reversed. Discount!
Spot instances - bid for unused EC2 capacity.
ELB can help distribute traffic among EC2 instances.

Can pay for detailed CloudWatch monitoring
Auto Scaling free.
Elastic IP addresses - can have 1 E IP address per instance for free.

Still need to get existing licenses for stuff like Oracle or Windows Server.

S3:

Storage for the internet.
First, determine storage class:

    - Standard Storage
        - 99.99% durability, 99.99% availability inside a given year
    - Standard-Infrequent Access (S-IA)
        - 99.99% durability, 99.99% availability
        - Less frequently accessed data, less redundancy.
- Pricing based on number of requests, GETS/POSTs

EBS (Elastic Bock Store)

Block-level storage for instances. Persist independently from the instances (virtual disks)

- General purpose (SSD)
- Provisioned IOPS (SSD)
    - Input Output per Second. Charged by the amount you provision, *multiplied* by percentage of days you provision for the month.
- Magnetic (HDD)

Charged by volume provisioned by user per gigabyte/month. Pay for what you *don't* use.
Snapshots of EBS volumes also cost $ per GB-month of data stored.
Inbound data transfer for snapshot free, outbound $$$$ (restore = $$$)

RDS:

Relational database in the cloud.
Cost is *clock-hour* billing. Incur charges when running.
Engine, size, memory of the database costs more $.

- On-demand - pay by the hour it runs
- Reserved database - up-front payment for DB instances (1 year/3 year term)

Provisioned storage - no charge for backup storage up to 100% of ur DB storage.
After DB terminated, then you get billed for the backups if you keep them.

CloudFront:

Global CDN, integrates w/ other AWS services.
Pricing varies across geographic regions. (Based on requests/data transfered out of CloudFront).
To estimate cost, estimate usage from ur AWS instances.

AWS Trusted Advisor

Gives the best practices and checks all of your account resources to see if they're in accordance with best practices. Saved customers over $500M

Best practices for:

- Cost optimization
    - e.g $XX,XXX potential monthly savings
- Performance
- Security
- Fault Tolerance

Can do stuff like:

- Underutilized EC2 instances
- EC2 Reserved Instances
- Underutilized EBS volumes
- Security/IAM checks (like old keys that aren't rotated)
- Fault tolerance - avoid outages with Direct Connect, check age of EBS snapshots

Trusted advisor is a console and API. Can automate Trusted Advisor with CloudWatch Events (Lambda powered automatic actions). Can also make it fetch new data for checks with a refresh button. Lastly, can filter checks by resource tags..

AWS Support Plans

Developed to provide complete support and resources to aid success.

Supports those that are experimenting with AWS, Prod Use, and Business Critical. Basically everyone.

Proactive guidance:
    TAM (Technical account manager) - user's primary point of contact
Best practices:
    Trusted Advisor. Automated cloud expert that checks for opportunities and issues.
Account Assistance:
    AWS Support Concierge. Billing/Account expert to solve issues.

4 support plans:

- Basic Support
- Developer Support
- Business Support
- Enterprise Support

Knowledge Check

Which of the following statements best describes AWS Trusted Advisor?

A tool that provides you real time guidance to help you provision your resources following AWS best practices.

Remember, Trusted Advisor = best practices (cost, fault tolerance, etc).

Knowledge Check

Friday, November 6, 2020 1:32 PM

Remember:

Users vs Roles vs Groups:

User - person or service who uses IAM to interact with AWS
Groups - collection of users
Roles - role has no credentials, can be used by anyone who needs it. Kind of like sudoers.

AWS Trusted Advisor vs Inspector:

Trusted Advisor helps with best practices for cost, performance, security, fault tolerance
Inspector: checks config of EC2 instances, like patching, known-vulnerabilities

AWS Regions vs Edge Locations:
Edge Locations serve requests for CloudFront/Route 53
Regions just have multiple Azs (data centers)

AWS Shared responsibility model:

AWS responsible for security of the cloud (Network)

The Shared Responsibility Model: (who is responsible for what)

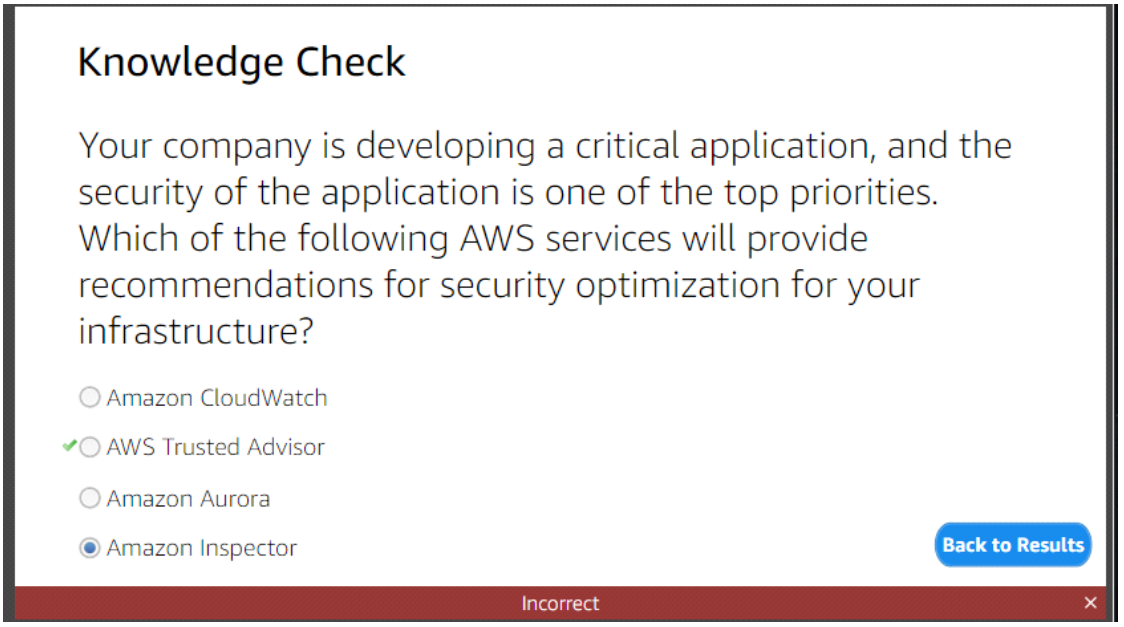
- **(USER)** User Data - no idea about user data held in apps
- **(USER)** Application - customer chooses apps
- **(USER)** Guest OS - customer chooses OS

~~~~~~

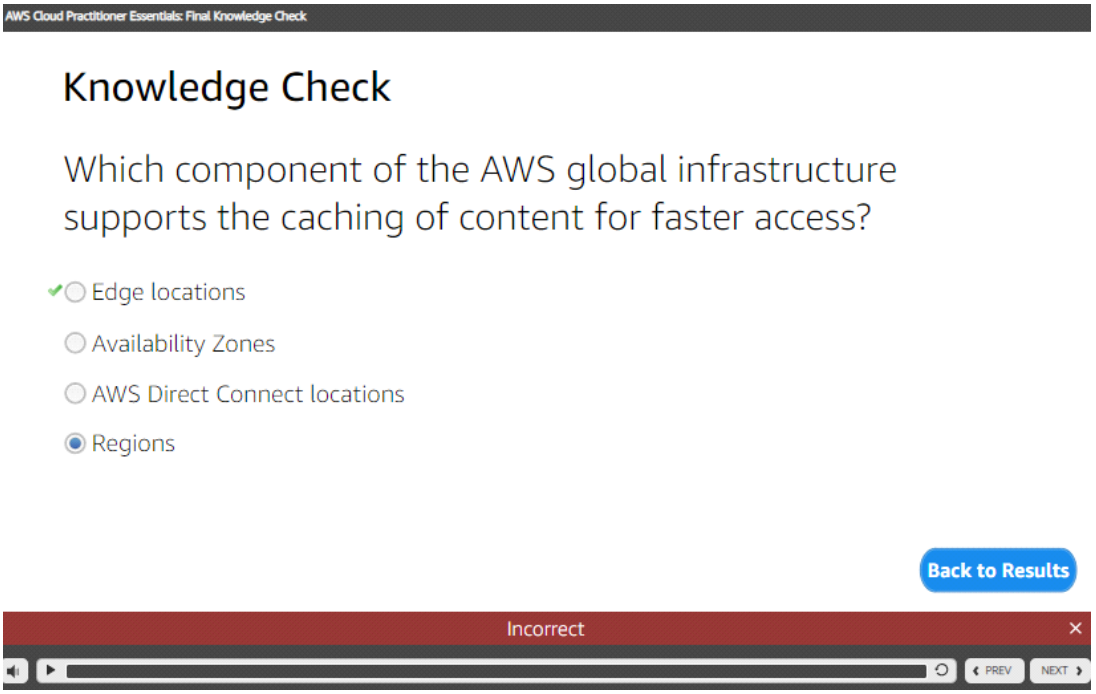
(magic dividing line, AWS can't see anything above this. Can't read any of this if they wanted to, even for marketing)

- **(AWS)** Hypervisor - zen-based.
- **(AWS)** Network - networking protocols, VPC can work at scale. They don't tell customer how it works but they get audited
- **(AWS)** Physical - iron/concrete/barbed wire. No tours!

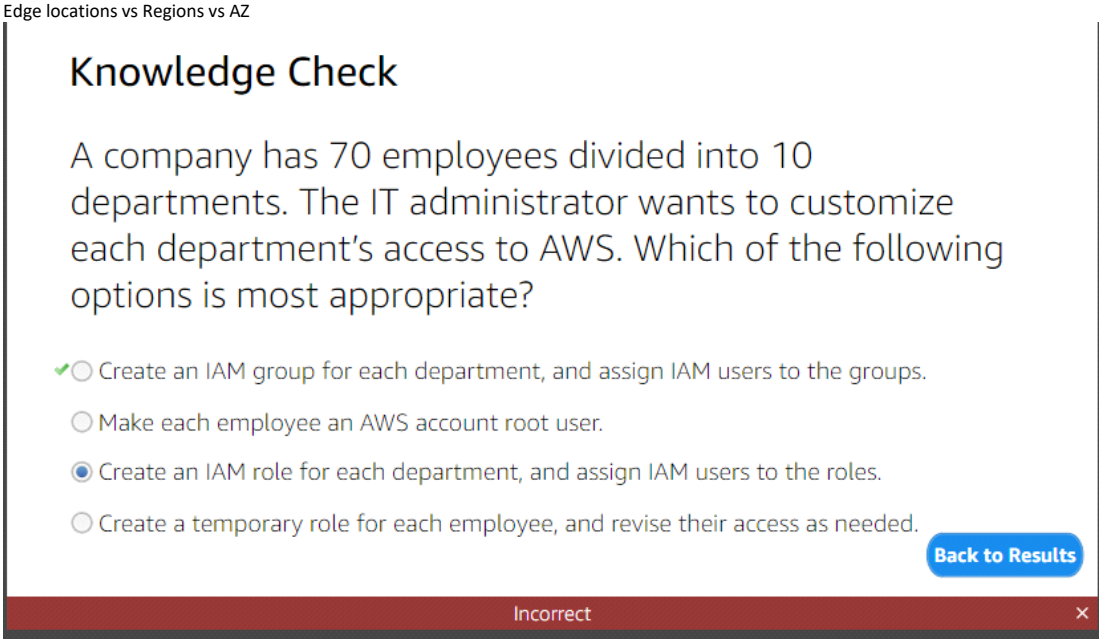
Questions I missed:

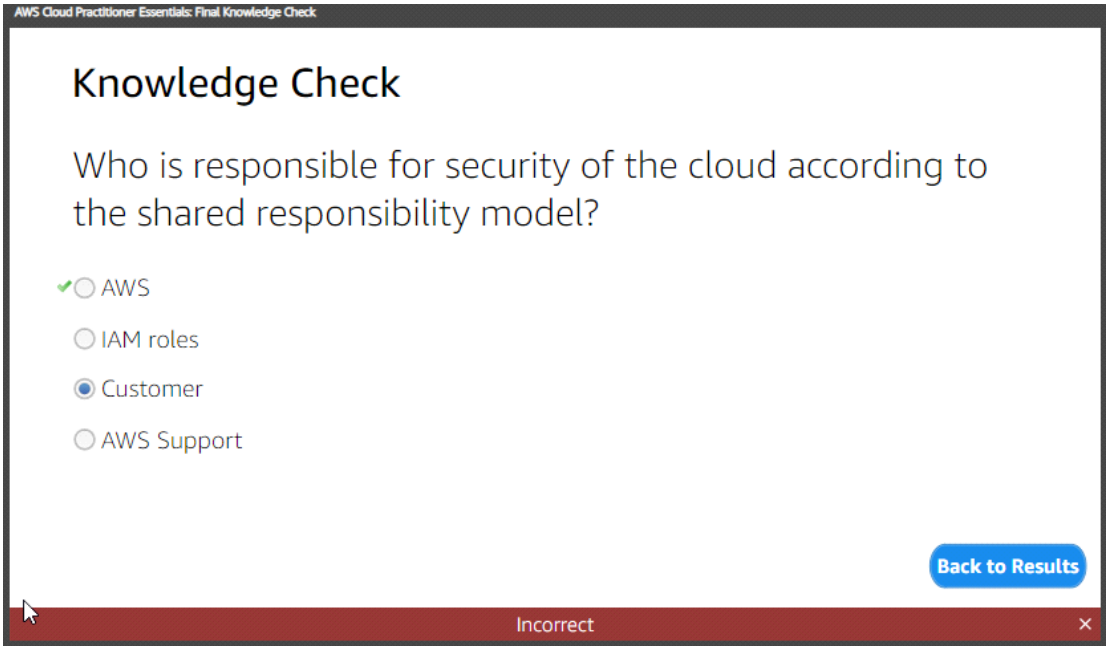


Trusted Advisor vs Inspector
~~~~~~



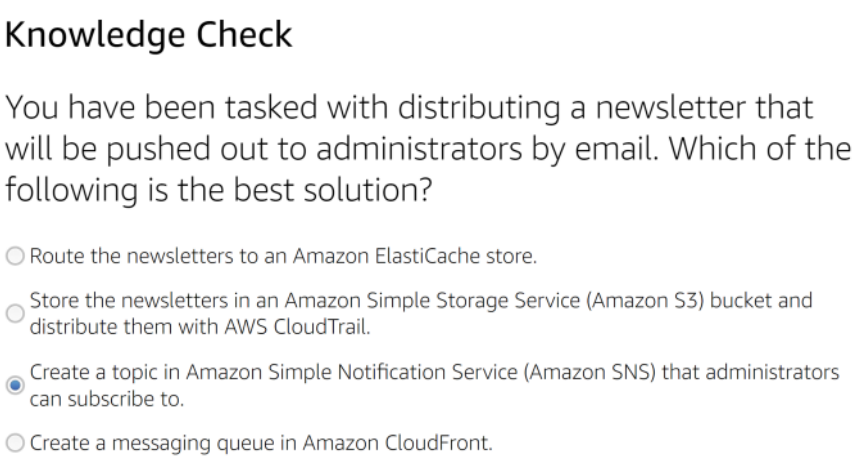
Edge locations vs Regions vs AZ





Study 5 pillars of Well-Architected Framework.

Questions I wasn't sure about:



Yup

Knowledge Check

Which of the following components is included in the value proposition of the AWS Cloud?

- ○ Informal security restrictions
- ○ Fully independent development without parameters
- ◉ Massive economies of scale
- ○ Physical relocation of your servers

Yup

Knowledge Check

Which of the following use cases is appropriate for Amazon CloudFront? (Select **Three**.)

- ☐ Auto Scaling
- ☐ Schema generation
- ☐ Security and encryption
- ☐ Database backups
- ☑ Live on-demand video streaming
- ☑ Static asset caching

Security and Encryption in the CDN

Knowledge Check

A company has 70 employees divided into 10 departments. The IT administrator wants to customize each department's access to AWS. Which of the following options is most appropriate?

- ○ Create an IAM group for each department, and assign IAM users to the groups.
- ○ Make each employee an AWS account root user.
- ○ Create an IAM role for each department, and assign IAM users to the roles.
- ○ Create a temporary role for each employee, and revise their access as needed.

Knowledge Check

Which of the following AWS concepts refers to "Established best practices developed through lessons learned by working with customers"?

- ○ Security of the Cloud
- ◉ Well-Architected Framework
- ○ Reference architecture
- ○ AWS Trusted Advisor

Yup, NOT Trusted advisor. AWS Concept != Service

Knowledge Check

Your company is developing a critical application, and the security of the application is one of the top priorities. Which of the following AWS services will provide recommendations for security optimization for your infrastructure?

- ○ Amazon CloudWatch
- ○ AWS Trusted Advisor
- ○ Amazon Aurora
- ◉ Amazon Inspector

Knowledge Check

Which component of the AWS global infrastructure supports the caching of content for faster access?

- ○ Edge locations
- ○ Availability Zones
- ○ AWS Direct Connect locations
- ◉ Regions

Knowledge Check

Which of the following tasks is the customer's responsibility when creating Amazon VPC security groups?

- ○ Choosing the level of physical security for the network.
- ○ Selecting an appropriate load balancing strategy for the network routers.
- ◉ Adding rules regarding inbound traffic to the security group.
- ○ Ensuring that the security groups are linked to Amazon EC2.

Knowledge Check

Which service sends notifications or automatically makes changes to the resources being monitored based on rules you established?

- ○ Amazon EC2
- ○ Amazon Aurora
- ○ Elastic Load Balancing
- ◉ Amazon CloudWatch

Knowledge Check

Which of the following services provides you with data and actionable insights to monitor your applications, understand and respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health?

- ○ AWS CloudTrail
- ○ AWS Cloud9
- ○ AWS CloudFormation
- ◉ Amazon CloudWatch

Knowledge Check

Which of the pillars of the Well-Architected Framework is defined as the ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures?

- ○ Security
- ○ Reliability
- ○ Operational excellence
- ○ Performance efficiency

Yes, not performance effiency

AWS Practice Exam 1

Monday, November 9, 2020 12:32 PM

SECOND Attempt:
63 -> 93
Snowball can transfer data in and *out* of AWS
Snowball, Petabyte, Snowmobile, Exabyte

AWS Quick Start reference deployments - Deploying random popular technologies like IBM MQ
AWS Cost & Usage Reports - detailed information about past usage
AWS Managed Services:

- Amazon DynamoDB
- **Amazon Elastic MapReduce** (EMR) - launch Big Data clusters in minutes (Hadoop) (NOT SERVERLESS)

AWS CLI needs X to work:

- **Access Keys** (access key ID and secret access key)

NOT Secret token

AWS Artifact - Manage agreements with AWS and compliance
AWS Systems Manager - visibility and control of infrastructure on AWS. Unified UI

Remember there is an **AWS Abuse Team** to handle malicious AWS usage. Not always Security Team.

CloudFront uses **AWS Edge Locations** to distribute content to users.

CloudTrail - service that enables governance, compliance, auditing of AWS account. Log activity in AWS.

AWS Abuse Team - helps when AWS resources are engaged in *malicious*/abusive behavior.
AWS Security Team - only responsible for security of services offered by AWS
AWS Concierge Team - billing/account management

CloudFront - CDN that helps achieve high transfer speeds all over the world.
Amazon Kinesis Video Streams - securely stream video from IoT devices

CloudFront uses Edge Locations!!

AWS Shared Responsibility Model - responsibilities will vary depending on the service used. AWS handles security of *managed services*

On-demand vs Reserved EC2 - reserved instances must be *for at least a year*

Snowmobile vs Snowball - Snowmobile larger, therefore exabyte. Snowball is petabyte.

AWS Artifact - self-service audit portal, look at AWS agreements and compliance documents

Spot Instances vs On-demand - spot instances much more cost effective, good if application can be

flexible and run on spare EC2 capacity

Amazon Athena - SQL service for S3
ECR - Elastic Container Registry (Docker repo)

What would be the best option to notify company when bill exceeds $2000?
Cloudwatch/SNS

Amazon Connect - customer service tech

Infrastructure Event Management (IEM) - short-term engagement with AWS to provide architectural and scaling guidance
Vs Support Concierge, only helps with account/billing

AWS Shared Controls - AWS provides requirements for the infrastructure and customer provides control implementation…
AWS handles Patch Management, Configuration Management (of underlying infrastructure), and Awareness/Training of AWS employees

AWS Operations Support - provides operations assessments to identify gap in operations lifecycles

Amazon Elastic MapReduce (EMR)/DynamoDB - AWS managed services. EMR launches clusters (Hadoop), DynamoDB serverless DB tables. AWS IAM is *not* a managed service

Penetration testing - can be performed by customer on their own *without* AWS authorization

AWS Quick Start Reference Deployments - launches/runs AWS compute to deploy specific workloads. Good for deploying popular technologies w/ least effort and time
AWS OpsWorks is for Chef/Puppet managed instances.

AWS CLI requires access keys

Amazon DynamoDB - NoSQL
Amazon Aurora - MySQL/PostgresSQL relational DBs and backups.
Amazon Redshift - Data warehouse

Amazon ElastiCache - web service to cache stuff in the cloud. Helps performance

S3 Storage classes…
Unpredictable access patterns - Intelligent-Tiering. Optimizes costs/
Standard - high durability/availability for frequently accessed data
Standard-Infrequent - data accessed less frequently, but rapid access when it is needed

S3 is a storage service, not compute service, can't run anything. S3 is *not* scaled manually. But it does provide unlimited storage for any type of data.

Enterprise support plan needs - TAM (Technical Account Manager). Expert that helps with stuff.

AWS Organizations - centrally govern their environments, good if you have many accounts globally

Convertible RIs - (EC2 reserved instances) - can modify EC2 reserved instances.

Highest level of availability - deploy across multiple *regions* and AZs

AWS Personal Health Dashboard - alerts and remediation guidance when **AWS has outages**. Also is personalized for your AWS managed services that may have outages

Data structured in key-value format - DynamoDB (NoSQL, MongoDB.. Json-like)
Remember RedShift is a data warehouse

AWS Groups, organize users into teams and manage permissions per team

AWS Practice Exam 2

Monday, November 9, 2020 1:25 PM

Second Attempt - ~75 to 93
dedicated host vs dedicated instance
Dedicated Host - most BYOL scenarios
Dedicated Instance - still share hardware somewhat, not great for all BYOL

primary storage service used by Amazon RDS database instances
Amazon EBS, NOT S3 for RDS. S3 isn't good for live databases (for backups tho maybe)

Reduce S3 class by *using right combination of **storage classes*** based on different use cases..
e.g

- S3 Standard-IA
- S3 Intelligent Tiering
- S3 One Zone (1 AZ)
- S3 Glacier

APN Consulting Partners (help other companies USE AWS) vs APN Technology Partners (sell other customers software HOSTED on AWS)

Impacts price paid on EC2:

- Instance Type
- Storage Capacity

NOT the AZ where it's deployed. Vary on *region*, NOT AZ

Amazon Neptune - Graph database product

ClondEndure - automates large-scale migrations from on-prem to AWS

RDS features to improve availability of your database -
Read Replicas, Multi-AZ deployments (patching doesn't help availability)

AWS X-Ray - help devs analyze/debug applications

CloudTrail - track users actions taken in your AWS account. Helps with compliance

CloudEndure - process of migrating applications from physical/virtual stuff to AWS regions

Security Groups - nothing to do w/ IAM. Acts a virtual firewall for instances.

Change management tools to audit/monitor resource changes:
AWS Config - "what did my AWS resource look like" at this point and time. EC2 config
AWS CloudTrail

AWS services with native Multi-AZ fault tolernace -
S3
DynamoDB

NOT EBS- EBS is 1 AZ

Cloudwatch logs - monitor/collect log files from EC2 Instances
NOT Amazon Inspector - automated security assesment. CloudWatch Logs can make events on log patterns.

AWS WAF - protect from SQL injection in apoplication code.
Amazon Cognito - SSO

Amazon EMR - analyze/process vast amounts of data
Amazon MQ is a messing service (between apps I think)

APN Consulting Partners - professional service firms that help customers do stuff on AWS
Also APN Technology Partners - SaaS/PaaS on cloud

AWS Service Limits - helps guarantee availability of AWS resources and help w/ billing risks for new customers. You can:

1. Contact AWS support to increase service limits
2. Use Trusted Advisor to monitor service limits.

S3 Transfer Acceleration - fast/easy/secure transfers of files long distances between client and an S3 bucket. Uses CloudFront

RDS uses EBS for storage.

Well-Architected Framework
Operational Excellence (Well-Architected Framework) - ability to monitor systems and improve supporting processes and procedures

Security - protect info/systems/assets

Reliability - recover from failure

Performance efficiency - use computing resources efficiently to meet system requirements.

Cost optimization - avoid or eliminate unneeded costs or sub-optimal resources

ElastiCache - in-memory caching for read-heavy applications

Under shared responsibility model, what controls do customers fully inherit from AWS?
NOT Awareness and Training - while AWS trains their employees, customer must also train their own employees

Environment controls/physical controls fully done by AWS

Glacier - active archives and long-term analyic data

Reducated Capital Expenditure (CapEx) - benefit of moving from on-prem to AWS. Don't have to keep up with buildings, equipment, machinery. AWS does not provide "free support for all enterprise customers" (only w/ paid support plans)

EC2 is pure IaaS, not SaaS at all.

BYOL and EC2 (Bring your Own License) - *Dedicated Hosts*. Use existing, server-bound licenses with hosts. On-demand/reserved instances don't support this.

How are AWS customers billed for Linux-based Amazon EC2 usage?
NOT one hour increments with minimum of one day.
One-second increments with a minimum of one minute.

Windows-based instances are charged with minimum of one hour, however.

Security Groups - firewall for EC2. NACL is optional security for VPC, not EC2.

AWS Practice Exam 3

Monday, November 9, 2020 2:01 PM

SECOND ATTEMPT - 80% -> 86%

****Amazon SWF**** (Simple Workflow Service) - coordinates tasks across distributed application components

Note that EMR (Map Reduce) does not scale by itself - needs auto scaling to do so.

Planning *application* migration to AWS - AWS Application Discovery Service.
NOT AWS Migration Hub - just track migrations already in-progress

Use cases for S3:
- Media store for CloudFront
- Hosting *static* websites

Not good for active databases. But can host static, non-changing websites. (basically nothing that needs PHP)

SSL deployment
- AWS ACM (AWS Certificate Manager)
- IAM can somehow, for SSL in regions where there is no ACM

Benefits of RDS:
- *Resizable compute capacity* (does NOT scale automatically, only Aurora does)
- Lowers administrative burden

Performance efficiency in AWS:
- Serverless architectures
- Multi-region architectures to better reach customers

AWS Shared controls:
Controls that apply to both the infrastructure and customer layers

What does AWS offer to secure your network?
- Customer-controlled encryption in transit

NACLs are fully customer controlled, NOT aws controlled

Minimum level of support for 24x7 access to tech support via phone/chat:
- Business

Simplify connection management among VPCs:
- AWS Transit Gateway (network hub to interconnect VPCs)

NOT VPC Peering (*only between **TWO*** VPCs)

Cost and Usage Report vs Cost Explorer:

Cost and Usage - granular data about their AWS costs/usage. Source of truth for billing pipeline
Cost Explorer - visualize/understand AWS costs and usage over time

What does AWS offer to secure your network?
Customer-controlled encryption in transit - AWS customer responsible for this.. But AWS offers it?

NACL (network access control lists) are customer controlled NOT AWS-controlled

Dedicated Instances - AWS hardware NOT shared by other AWS customers

AWS QuickSight - business analytics service for visualizations

AWS Transit Gateway - service that simplifies the connection management among the VPCs.
Interconnect many VPCs
Amazon Connect - cloud-based customer service center!!!!!

Choosing the right database technology
1. Nature of queries
2. Number of reads/writes per second

NOT Data sovereignty - concept that information is subject to laws of the country where it is located

AWS Support Concierge service included in AWS Enterprise support.

Protecting data at rest on S3 - permissions and versioning

AWS Macie - protects sensitive data in S3. Machine Learning to discover PII sitting exposed

AWS Application Discovery service - helps with planning application migration to AWS Cloud
Aws Migration Hub - single location to *track progress* of your migrations

AWS DMS - database migration service

Tools for AWS to accerlate Enterprise adoption of AWS:
- AWS Partners
- AWS Professional Services

Amazon Athena - SQL
Amazon PinPoint - targeted email/SMS to customers
AWS Artifact - AWS compliance

Need EC2 instances always available for 2 months? Use on-demand instances

AWS Budgets - track/avoid over-spending on underutilized reserved instances. Set up alert notifications for utilization drops too

AWS Service Catalog - organizing and governing commonly deployed IT services. Helps w/ in-house compliance requirements

SSL cert management and deployment

- AWS ACM (Amazon Certificate Manager)
- AWS **IAM** (NOT Route 53) can also be used for certs

AWS shared controls - controls that apply to both infrastructure layer and customer layers
Securing infrastructure is always the responsibility of AWS

AWS CAF - Cloud Adoption Framework. Helps organizations design a road map to successful cloud adoption.

Amazon SWF - Simple Workflow service, makes it easy to coordinate work across distributed applications. *Coordinates tasks across distributed application components*

Benefits of DynamoDB:
1. Extremely low latency
2. Automatically scales to meet requirements

DOES NOT SUPPORT other database engines like CouchDB/MongoDB - DynamoDB is it's own engine

AWS Service Health Dashboard vs **AWS Personal Health Dashboard**
Service Health Dashboard - AWS service availability for ALL services
AWS Personal Health Dashboard - *personalized* view of AWS services that power your applications, NOT ALL SERVICES

AWS TCO Calculator - Total Cost of Ownership, cost-benefit analysis of moving from on-prem to AWS Cloud

Benefits of RDS (relational database service):
- Lower administrative burden (patching)
- Resizable compute capacity - *does not automatically scale.* Only Aurora will scale (MySQL/PostgreSQL)

Business Support vs Basic vs Enterprise:
- Business Support - 24/7 technical support engineers, phone and chat
- Enterprise Support - Business Support + lots of shit
- Basic Support - does not get 24/7 support
- Developer Support - tech support only during business hours and only email

AWS Cost Explorer - visualize AWS spending in the last few months

AWS Compute Service that executes code **when triggered by events**
AWS Lambda. CloudWatch does not execute code.

Use cases for S3:
- Hosting static websites
- Media store for Cloud-front

Not good for *database storage.* Okay for logs though.

AWS Practice Exam 4

Tuesday, November 10, 2020 10:54 AM

SECOND ATTEMPT: 78 -> 96
Amazon Cloud Directory - web-based directories to organize/manage app resources like users, locations, devices, etc
AWS Directory Service is AD SSO

AWS Global Accelerator - improve availability/performance of apps globally

Well-Architected Review is Enterprise only.

AWS Business - 24x7 access to customer service, IEM for an additional fee

What is the name of the DynamoDB replication capability that provides fast read \ write performance for globally deployed applications?

Global Tables - DynamoDB feature for fast read/write in global apps
DynamoDB DAX - cache to reduce response times

Fargate - serverless Docker

ECS - also docker stuff

Hosted Zones are factors in Route 53 costs. Not EC2.

Elastic Beanstalk/IAM are free.

Auto Scaling Groups - scales EC2 in multiple AZs

Remember ACM is Amazon Cert Manager

Amazon DocumentDB - AWS managed database

AWS WAF - monitor HTTP/HTTPS requests to CloudFront
CloudWatch just monitors utilization of resources

Global Tables - DynamoDB replication capability with fast read/write for global apps
AWS Global Accelerator - network service that enables organizations to route traffic to multiple regions

DynamoDB DAX - dynamodb cache
Dynamodb point-in-time-recovery - backup data per-second and restore

Best option for processing a large number of binary files - ec2 instances in parallel

Monitor HTTP/HTTPS traffic in CloudFront - Amazon WAF (Web Application Firewall)

Low-latency links in AZs allow synchronous replication of data

Lambda programming languages -
- Natively supports node.js, python, java, etc
- Can also support any programming language w/ an API

OpsWorks - supports Chef/Puppet by default

AWS Managed databases-
RDS for MySQL
DocumentDB

Amazon CloudSearch - google search for orgs

AWS ACM - cert manage, SSL/TLS

Security in the cloud done by customer:
- File system encryption
- Building a schema for an application

AWS CodeDeploy - service that automates application code deployments to EC2

Multiple AZs = highly available

EBS vs EFS (Elastic File System)
EBS ***can only be attached to one instance at a time***
EFS can be used by multiple EC2 instances

Linux v Windows billing -
Windows by the hour, Linux by the second

Security Bulletins - AWS publishes stuff about latest security/privacy events

Auto Scaling Groups - scales EC2 instances in multiple AZs for avaaiblity/fault tolerance

Estimate cost of AWS:
- AWS Simple Monthly Calculator - estimate monthly AWS bill
- AWS Budgets - set custom budgets and get alerted when exceeded
- AWS Cost and Usage Report - see detailed info on AWS costs/usages
- AWS Cost Explorer - *historical* trend of cost

AWS Business Support Plan:
1. 24x7 access to customer service
2. Access to IEM (infrastructure event management) for an additional fee
3. 24x7 access to Cloud Support Engineers

Only enterprise gets the TAM

AWS KMS - keys to encrypt data (Key Management Service)

Route 53 can *perform health checks on EC2 instances* (e.g, route traffic to only healthy endpoints)

Amazon SQS - can decouple components on an application
Amazon Artifact - compliance related info

Response times:
- Enterprise - under 15 min
- Business - 1 hour
- Developer - 12 hour
- Basic - nothing

IAM:
- Users - permanent long-term creds
- Groups - logically manage users
- Roles - temporary permissions

S3 - scales automatically while minimizing costs

Loosely coupling - minimize dependencies so that failure of one component doesn't impact others

Well-Architected Review:
Enterprise support, reviews best practices for business critical workloads. Business support doesn't get this.

Two factors when determining region AWS resources will be deployed:
- Cost
- Data sovereignty

AWS Services to *run* Microsoft SQL server on AWS:
- Amazon EC2 (Windows host)
- Amazon RDS

Amazon Database Migration Service (DMS) will only help migrate, not *run* the database

Type of MF advice that customers can use to protect resources:
- U2F (universal second factor) security key

Amazon cognito - login with other SSO accounts

Amazon Cloud Directory -
- **NOT Amazon Directory Service** (Amazon DS lets you use existing AD creds)
- Directory service that allows organization of hierarchies of data across multiple dimensions

AWS Artificant - SOC/PCI, compliance

An AWS AZ is an isolated location within an AWS region, however edge locations are located in multiple cities worldwide. *Edge locations may or may not exist within a region*

Amazon S3 provides volume discounts based on usage.

AWS Teams:

- Concierge Team - AWS billing/account experts
- AWS Professional Services - global team of experts to get desired business outcomes in AWS

Rekognition - facial recognition

Remember DynamoDB is serverless like Lambda

Free AWS Services:

- Elastic Beanstalk
- AWS IAM

NOT Elastic Load Balancing

AWS Practice Exam 5

Tuesday, November 10, 2020 11:36 AM

SECOND ATTEMPT - 84 -> 96
AWS Direct Connect - transfer large data sets to and from AWS everyday

Data transfer IN has no cost on Amazon stuff

AWS Resource Groups - custom consoles for dev/prod

AWS Application Discovery Service - useful only for planning a migration

Amazon Neptune - graph database service

Amazon MQ - message broker for cloud

IAM policy - assign permissions directly to a user

S3 Reserved Capacity doesn't exist. Pay for what you use.

AWS Cost Explorer - forecasts future costs based on *past usage*
Vs
AWS Simple Monthly Calculator - future costs based on *expected usage*

Amazon Fargate - ***serverless*** compute engine for containers that works with ECS

NOT a factor when estimating cost of CloudFront -
- Inbound Traffic

Data Transfer out, number/type of requests (HTTP/HTTPS), and the edge location cost $$

DAX - DynamoDB feature to reduce latency of requests

ACID transactions - use RDS! Not Redshift (data warehouse)

Server-based stuff:
RDS, EMR

Cheapest EC2: Reserved, Standard, All Upfront

Cost-effective storage option to immediate retrieval database backups:
Amazon S3. EBS isn't as cost effective, and glacier can't retrieve that fast.

EBS pricing:
- Amount of data stored in snapshots
- Size of volumes provisioned per month

What Amazon does for you on Amazon RDS databases:
- Database setup

- Management of OS

AWS Resource Groups - allows you to create a custom console for each environment to view and manage resources easily

Amazon ElastiCache for Redis - fast in-memory data store that gives sub-millisecond latency for stuff like IoT

RDS supports:
- MS SQL Server
- Oracle
- PostgreSQL

Tags to group resources helps analyze costs in AWS

RDS Multi-AZ:
Performs *automatic failover* when the primary database fails to respond. Feature of RDS.

Serverless:
- Lambda
- Fargate (Serverless Docker engine)

ECS launches Fargate or EC2, with only Fargate being serverless.

Custom relational database software:
Use EC2, NOT RDS. If it's full control over DB, need EC2. RDS is managed.

AWS Direct Connect - transfer large amounts of data from on-prem to AWS everyday

Dedicated Host vs Dedicated Instance:
- Host - BYOL, hardware dedicated to one customer only
- Instance - instances run on hardware dedicated to one AWS account, but may share hardware with other instances from the same AWS account. Dedicated Hosts allow more control.

There are no reservations in S3. Pay for what you use.

AWS Server Migration Service - migrate large number of on-prem workloads to AWS

AWS has eliminated bidding in the new AWS Spot pricing model.

Greatest impact on cost:
- Data transfer out
- Compute charges

AWS Global Accelerator - networking service that improves availability and performance of applications to global users

Loose coupling > tight coupling

Don't create an API access key for the root account unless you need to

Assign permissions directly to an IAM user:
Use *IAM Policy*, NOT IAM role. IAM Role, temporary permissions assigned to users/groups. Policy, direct

permissions.

Use cases for Amazon EMR (**NOT DOCKER - Docker is ECS)**
EMR - Elastic Map Reduce, web service that enables big data stuff

- Hadoop, Apache Spark, Big Data Frameworks
- Process large amounts of data in a timely manner

Remember regions are geographic locations.

S3 storage classes:

- S3 Standard - high durability for frequently accessed data, like websites
- Standard-IA (infrequently accessed) - not for popular websites, long-term storage and backups
- Intelligent-Tiering - access patterns that are unknown or unpredictable
- S3 Glacier Deep Archive - lowest-cost storage for long term retention

Amazon ElastiCache - improves performance of applications by allowing retrieval of information from fast caches
DAX is similar but only DynamoDB

Snowball is smaller than Snowmobile. Petabyte is 1000TB, Exabyte is 1000PB

AWS Cost Explorer - highly accurate forecasts for up to 12 months ahead. *Forecasts future costs on past usage*, NOT expected usage (that's AWS Simple Monthly calculator)

AWS Practice Exam 6

Tuesday, November 10, 2020 1:04 PM

Free stuff to all users:
- Security Blogs and Bulletins (also security stuff)

Elastic load balancing does not scale resources. Just distributes traffic.
Serverless computing is highly elastic.

AWS Batch - enables devs/scientists/etc to run thousands of batch jobs on AWS. Maybe like processing tons of covid tests

AWS Pricing:
- Only pay for services you need, no long term contracts
- With AWS, don't have to pay an upfront fee

AWS will provide licenses for *some* software not developed by AWS

Avaiblity Zone:
- Distinct location in a region that is insulated from failures in other AZs
- Note that it's NOT completely isolated from other DCs

Services that auto replicate data across AZs:
- S3
- DynamoDB

Durability - benefit of Ebs volumes being auto replicated within the same AZ

24/7 access to cloud support engineers via email and phone:
Business and Enterprise. Developer doesn't have *phone* 24/7 but does have email

AWS Organizations can add accounts together for consolidated billing. One bill and combined usage.

AWS Tagging - categorize, manage and filter resources

S3 Storage Classes Part 2:
- Standard - high avaibility rating at 99.99%
- S3 Standard-IA (Infrequent Access) - 99.9% avaibility
- Glacier - 99.99% avaibility as long as it's used for long term storage
- **One Zone-IA** (Infrequent Access) - 99.5%. LOWEST avaibility. One uses 1 AZ (One Zone)

EBS cost effected by:
- Volume types (magnetic, SSD)
- Snapshots

Amazon Elastic Transcoder - Transcode media for mobile playback

AWS Price List API - know the price of AWS services programattically

AWS CodeCommitt - repo management for versioning code, AWS GitHub

IAM user and AWS account root user both get an access key ID and secret access key. Both are long-lived (roles are not)

Reduce EBS costs:
1. Delete unnecessary snapshots
2. Change type of volume

AWS Lambda charges:
- Compute time consumed
- Number of requests to functions

Seven design principles for security in the cloud (well 2):
- Use IAM roles to grant temp acccess instead of long-term creds
- *Enable real-time traceability* (Monitor, alert, audit actions in real time)

Others:
- POLP
- Security at all layers
- Automate security best practices
- Protect data in transit and at rest
- Keep people away from data
- Prepare for security events (incident management process)

https://status.aws.amazon.com
AWS Service Health Dashboard

AWS Marketplace:
- Provides flexible pricing options that suit most customers needs
- Protect customers by performing periodic security checks on listed products (like Google Play Store)

Reserved Instance Types:
- Convertible RIs - good for steady usage, but can change attributes of RI to equal or greater value
- Standard RIs- steady, long term usage
- **Scheduled RIs -** launch within *specific time windows*

AWS Well Architected Framework
- Performance effiecny: ability to use computing resources effiecntly to meet system requirements. Help customers select the right compute resources based on workload requirements
- Operational excellence: ability to run/monitor systems and improve support processing/procedures over time

AWS Business support plan provides:
AWS Support API
Access to *full set of Trusted Advisor checks*

Only Enterprise gets the TAM, less than 15 minute response time, and Support Concierge (billing/account experts)

Remember EBS = 1 EC2 instance, EFS = many

Historical billing information: Billing and Cost management console
AWS Budgets is just for custom budget alerting

Benefits of AWS Organizations:

- Consolidated billing
- Control *access to AWS services*

Does NOT manage payment methods. That's Billing and Cost Management

Provides access to *only* seven core AWS trusted advisor checks:
Basic and Developer support plans. (see only)

ElastiCache used for:

- Improve web application performance
- Provide an in-memory data storage service (cache)

Only stuff to use Chef/Puppet is OpsWorks

AWS Lambda is not resizable, automatically scales itself to meet workload

CloudHSM - generate/use your own encryption keys

Federation - sign in to AWS accounts with existing corporate credentials
(IAM, AWS Directory Service (AD), and AWS SSO)

Lightsail - VPS

"Cloud" deployment model elimnates need to run and maintain physical data centers

Amazon GuardDuty - threat detection
Macie - ML to identify PII

AWS KMS - can be used to *encrypt EBS volumes* (key management service used to encrypt, can also be used in S3/Redshift)