Disclaimer: These slides are copyrighted and strictly for personal use only

- This document is reserved for people enrolled into the Ultimate AWS Solutions Architect Associate Course
- **Please do not share this document**, it is intended for personal use and exam preparation only, thank you.
- If you've obtained these slides for free on a website that is not the course's website, please reach out to piracy@datacumulus.com. Thanks!
- Best of luck for the exam and happy learning!

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Certified Solutions Architect Associate Course

SAA-C02

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Welcome! We're starting in 5 minutes

- We're going to prepare for the Solutions Architect exam - SAA-C02
- It's a challenging certification, so this course will be long and interesting
- Basic IT knowledge is necessary
- This course contains videos…
  - From the Developer and SysOps course - shared knowledge
  - Specific to the Solutions Architect exam - exciting ones on architecture!
- We will cover over 30 AWS services
- AWS / IT Beginners welcome! (but take your time, it's not a race)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

My certification: 98.2%

AWS Certified Solutions Architect - Associate

Notice of Exam Results

Candidate: Stephane Maarek	Exam Date: January 30, 2019
Candidate ID: AWS00614912	Registration Number: 513425
Candidate Score: 982	Pass/Fail: PASS

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

About me

- I'm Stephane!
- Working as in IT consultant and AWS Solutions Architect, Developer & SysOps
- Worked with AWS many years: built websites, apps, streaming platforms
- Veteran Instructor on AWS (Certifications, CloudFormation, Lambda, EC2…)

- You can find me on
  - GitHub: https://github.com/simplesteph
  - LinkedIn: https://www.linkedin.com/in/stephanemaarek
  - Medium: https://medium.com/@stephane.maarek
  - Twitter: https://twitter.com/stephanemaarek

4.6 Instructor Rating
18,852 Reviews
66,021 Students
19 Courses

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What's AWS?

- AWS (Amazon Web Services) is a Cloud Provider
- They provide you with servers and services that you can use on demand and scale easily

- AWS has revolutionized IT over time
- AWS powers some of the biggest websites in the world
  - Amazon.com
  - Netflix

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What we'll learn in this course (and more!)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Navigating the AWS spaghetti bowl

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Udemy Tips

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Fundamentals – Part I

Regions, IAM & EC2

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Regions

- AWS has regions all around the world (us-east-1)
- Each region has availability zones (us-east-1a, us-east-1b…)
- Each availability zone is a physical data center in the region, but separate from the other ones (so that they're isolated from disasters)
- AWS Consoles are region scoped (except IAM, S3 & Route53)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

IAM Introduction

- IAM (Identity and Access Management)
- Your whole AWS security is there:
  - Users
  - Groups
  - Roles
- Root account should never be used (and shared)
- Users must be created with proper permissions
- IAM is at the center of AWS
- Policies are written in JSON (JavaScript Object Notation)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Introduction

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

IAM Introduction

- IAM has a **global** view
- Permissions are governed by Policies (JSON)
- MFA (Multi Factor Authentication) can be setup
- IAM has predefined "managed policies"
- We'll see IAM policies in details in the future
- It's best to give users the minimal amount of permissions they need to perform their job (least privilege principles)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Federation

- Big enterprises usually integrate their own repository of users with IAM
- This way, one can login into AWS using their company credentials
- Identity Federation uses the SAML standard (Active Directory)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM 101 Brain Dump

- One IAM **User** per PHYSICAL PERSON
- One IAM **Role** per Application
- IAM credentials should NEVER BE SHARED
- Never, ever, ever, ever, write IAM credentials in code. EVER.
- And even less, NEVER EVER EVER COMMIT YOUR IAM credentials
- Never use the ROOT account except for initial setup.
- Never use ROOT IAM Credentials

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What is EC2?

- EC2 is one of most popular of AWS offering
- It mainly consists in the capability of :
  - Renting virtual machines (EC2)
  - Storing data on virtual drives (EBS)
  - Distributing load across machines (ELB)
  - Scaling the services using an auto-scaling group (ASG)

- Knowing EC2 is fundamental to understand how the Cloud works

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Hands-On:
Launching an EC2 Instance running Linux

- We'll be launching our first virtual server using the AWS Console
- We'll get a first high level approach to the various parameters
- We'll learn how to start / stop / terminate our instance.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSH Summary Table

	SSH	Putty	EC2 Instance Connect
Mac	✓		✓
Linux	✓		✓
Windows < 10		✓	✓
Windows >= 10	✓	✓	✓

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Which Lectures to watch

- Mac / Linux:
  - SSH on Mac/Linux lecture

- Windows:
  - SSH on Mac/Linux lecture (because of the chmod 0400 command)
  - Putty Lecture
  - **If Windows 10:** SSH on Windows 10 lecture

- All:
  - EC2 Instance Connect lecture

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSH troubleshooting

- Students have the most problems with SSH

- If things don't work…
  1. Re-watch the lecture. You may have missed something
  2. Read the troubleshooting guide

- If things still don't work…
- Use EC2 Instance Connect!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How to SSH into your EC2 Instance
Linux / Mac OS X

- We'll learn how to SSH into your EC2 instance using Linux / Mac
- SSH is one of the most important function. It allows you to control a remote machine, all using the command line.

- We will see how we can configure OpenSSH ~/.ssh/config to facilitate the SSH into our EC2 instances

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How to SSH into your EC2 Instance
Windows

- We'll learn how to SSH into your EC2 instance using Windows
- SSH is one of the most important function. It allows you to control a remote machine, all using the command line.

- We will configure all the required parameters necessary for doing SSH on Windows using the free tool Putty.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Instance Connect

- Connect to your EC2 instance within your browser
- No need to use your key file that was downloaded
- The "magic" is that a temporary key is uploaded onto EC2 by AWS

- Works only out-of-the-box with Amazon Linux 2

- Need to make sure the port 22 is still opened!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Introduction to Security Groups

- Security Groups are the fundamental of network security in AWS
- They control how traffic is allowed into or out of our EC2 Machines.

- It is the most fundamental skill to learn to troubleshoot networking issues
- In this lecture, we'll learn how to use them to allow, inbound and outbound ports

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Security Groups Deeper Dive

- Security groups are acting as a "firewall" on EC2 instances
- They regulate:
  - Access to Ports
  - Authorised IP ranges – IPv4 and IPv6
  - Control of inbound network (from other to the instance)
  - Control of outbound network (from the instance to other)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	test http page
SSH	TCP	22	122.149.196.85/32	
Custom TCP Rule	TCP	4567	0.0.0.0/0	java app

NOT FOR DISTRIBUTION © Stephane Maarek www.

© Stephane Maarek

Security Groups Diagram

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Security Groups
Good to know

- Can be attached to multiple instances
- Locked down to a region / VPC combination
- Does live "outside" the EC2 – if traffic is blocked the EC2 instance won't see it
- It's good to maintain one separate security group for SSH access
- If your application is not accessible (time out), then it's a security group issue
- If your application gives a "connection refused" error, then it's an application error or it's not launched
- All inbound traffic is blocked by default
- All outbound traffic is authorised by default

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Referencing other security groups Diagram

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Private vs Public IP (IPv4)

- Networking has two sorts of IPs. IPv4 and IPv6:
  - IPv4: `1.160.10.240`
  - IPv6: `3ffe:1900:4545:3:200:f8ff:fe21:67cf`

- In this course, we will only be using IPv4.
- IPv4 is still the most common format used online.
- IPv6 is newer and solves problems for the Internet of Things (IoT).

- IPv4 allows for 3.7 billion different addresses in the public space
- IPv4: [0-255].[0-255].[0-255].[0-255].

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Private vs Public IP (IPv4) Example

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Private vs Public IP (IPv4) Fundamental Differences

- Public IP:
  - Public IP means the machine can be identified on the internet (WWW)
  - Must be unique across the whole web (not two machines can have the same public IP).
  - Can be geo-located easily

- Private IP:
  - Private IP means the machine can only be identified on a private network only
  - The IP must be unique across the private network
  - BUT two different private networks (two companies) can have the same IPs.
  - Machines connect to WWW using a NAT + internet gateway (a proxy)
  - Only a specified range of IPs can be used as private IP

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic IPs

- When you stop and then start an EC2 instance, it can change its public IP.
- If you need to have a fixed public IP for your instance, you need an Elastic IP
- An Elastic IP is a public IPv4 IP you own as long as you don't delete it
- You can attach it to one instance at a time

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic IP

- With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.

- You can only have 5 Elastic IP in your account (you can ask AWS to increase that).

- Overall, try to avoid using Elastic IP:
  - They often reflect poor architectural decisions
  - Instead, use a random public IP and register a DNS name to it
  - Or, as we'll see later, use a Load Balancer and don't use a public IP

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Private vs Public IP (IPv4) In AWS EC2 – Hands On

- By default, your EC2 machine comes with:
  - A private IP for the internal AWS Network
  - A public IP, for the WWW.
- When we are doing SSH into our EC2 machines:
  - We can't use a private IP, because we are not in the same network
  - We can only use the public IP.
- If your machine is stopped and then started, the public IP can change

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Launching an Apache Server on EC2

- Let's leverage our EC2 instance

- We'll install an Apache Web Server to display a web page
- We'll create an index.html that shows the hostname of our machine

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 User Data

- It is possible to bootstrap our instances using an EC2 User data script.
- bootstrapping means launching commands when a machine starts
- That script is only run once at the instance first start
- EC2 user data is used to automate boot tasks such as:
  - Installing updates
  - Installing software
  - Downloading common files from the internet
  - Anything you can think of
- The EC2 User Data Script runs with the root user

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 User Data Hands-On

- We want to make sure that this EC2 instance has an Apache HTTP server installed on it – to display a simple web page
- For it, we are going to write a user-data script.
- This script will be executed at the first boot of the instance.

- Let's get hands on!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Instance Launch Types

- **On Demand Instances:** short workload, predictable pricing
- **Reserved:** (MINIMUM 1 year)
  - **Reserved Instances:** long workloads
  - **Convertible Reserved Instances:** long workloads with flexible instances
  - **Scheduled Reserved Instances:** example – every Thursday between 3 and 6 pm
- **Spot Instances:** short workloads, for cheap, can lose instances (less reliable)
- **Dedicated Instances:** no other customers will share your hardware
- **Dedicated Hosts:** book an entire physical server, control instance placement

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

EC2 On Demand

- Pay for what you use (billing per second, after the first minute)
- Has the highest cost but no upfront payment
- No long term commitment

- Recommended for short-term and un-interrupted workloads, where you can't predict how the application will behave.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Reserved Instances

- Up to 75% discount compared to On-demand
- Pay upfront for what you use with long term commitment
- Reservation period can be 1 or 3 years
- Reserve a specific instance type
- Recommended for steady state usage applications (think database)

- Convertible Reserved Instance
  - can change the EC2 instance type
  - Up to 54% discount
- Scheduled Reserved Instances
  - launch within time window you reserve
  - When you require a fraction of day / week / month

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Spot Instances

- Can get a discount of up to 90% compared to On-demand
- Instances that you can "lose" at any point of time if your max price is less than the current spot price
- The MOST cost-efficient instances in AWS
- Useful for workloads that are resilient to failure
  - Batch jobs
  - Data analysis
  - Image processing
  - …
- Not great for critical jobs or databases

- Great combo: Reserved Instances for baseline + On-Demand & Spot for peaks

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Dedicated Hosts

- Physical dedicated EC2 server for your use
- Full control of EC2 Instance placement
- Visibility into the underlying sockets / physical cores of the hardware
- Allocated for your account for a 3 year period reservation
- More expensive

- Useful for software that have complicated licensing model (BYOL – Bring Your Own License)
- Or for companies that have strong regulatory or compliance needs

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Dedicated Instances

- Instances running on hardware that's dedicated to you
- May share hardware with other instances in same account
- No control over instance placement (can move hardware after Stop / Start)

Characteristic	Dedicated Instances	Dedicated Hosts
Enables the use of dedicated physical servers	X	X
Per instance billing (subject to a $2 per region fee)	X	
Per host billing		X
Visibility of sockets, cores, host ID		X
Affinity between a host and instance		X
Targeted instance placement		X
Automatic instance placement	X	X
Add capacity using an allocation request		X

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Which host is right for me?

- **On demand:** coming and staying in resort whenever we like, we pay the full price
- **Reserved:** like planning ahead and if we plan to stay for a long time, we may get a good discount.
- **Spot instances:** the hotel allows people to bid for the empty rooms and the highest bidder keeps the rooms. You can get kicked out at any time
- **Dedicated Hosts:** We book an entire building of the resort

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Price Comparison
Example – m4.large – us-east-1

Price Type	Price (per hour)
On-demand	$0.10
Spot Instance (Spot Price)	$0.032 - $0.045 (up to 90% off)
Spot Block (1 to 6 hours)	~ Spot Price
Reserved Instance (12 months) – no upfront	$0.062
Reserved Instance (12 months) – all upfront	$0.058
Reserved Instance (36 months) – no upfront	$0.043
Reserved **Convertible** Instance (12 months) – no upfront	$0.071
Reserved **Dedicated** Instance (12 months) – all upfront	$0.064
Reserved **Scheduled** Instance (recurring schedule on 12 months term)	$0.9 – $0.95 (5%-10% off)
Dedicated Host	On-demand price
Dedicated Host Reservation	Up to 70% off

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Spot Instance Requests

- Can get a discount of up to 90% compared to On-demand

- Define **max spot price** and get the instance while **current spot price < max**
  - The hourly spot price varies based on offer and capacity
  - If the current spot price > your max price you can choose to **stop** or **terminate** your instance with a 2 minutes grace period.
- Other strategy: **Spot Block**
  - "block" spot instance during a specified time frame (1 to 6 hours) without interruptions
  - In rare situations, the instance may be reclaimed

- Used for batch jobs, data analysis, or workloads that are resilient to failures.
- Not great for critical jobs or databases

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Spot Instances Pricing

User-defined max price

https://console.aws.amazon.com/ec2sp/v1/spot/home?region=us-east-1#

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How to terminate Spot Instances?

You can only cancel Spot Instance requests that are **open, active, or disabled.**
Cancelling a Spot Request does not terminate instances
You must first cancel a Spot Request, and then terminate the associated Spot Instances

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/spot-requests.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Spot Fleets

- Spot Fleets = set of Spot Instances + (optional) On-Demand Instances
- The Spot Fleet will try to meet the target capacity with price constraints
  - Define possible launch pools: instance type (m5.large), OS, Availability Zone
  - Can have multiple launch pools, so that the fleet can choose
  - Spot Fleet stops launching instances when reaching capacity or max cost
- Strategies to allocate Spot Instances:
  - **lowestPrice:** from the pool with the lowest price (cost optimization, short workload)
  - **diversified:** distributed across all pools (great for availability, long workloads)
  - **capacityOptimized:** pool with the optimal capacity for the number of instances

- Spot Fleets allow us to automatically request Spot Instances with the lowest price

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Instance Types – Main ones

- R: applications that needs a lot of RAM – in-memory caches
- C: applications that needs good CPU – compute / databases
- M: applications that are balanced (think "medium") – general / web app
- I: applications that need good local I/O (instance storage) – databases
- G: applications that need a GPU – video rendering / machine learning

- T2 / T3: burstable instances (up to a capacity)
- T2 / T3 - unlimited: unlimited burst

- Real-world tip: use https://www.ec2instances.info

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Burstable Instances (T2/T3)

- AWS has the concept of burstable instances (T2/T3 machines)

- Burst means that overall, the instance has OK CPU performance.
- When the machine needs to process something unexpected (a spike in load for example), it can burst, and CPU can be VERY good.
- If the machine bursts, it utilizes "burst credits"
- If all the credits are gone, the CPU becomes BAD
- If the machine stops bursting, credits are accumulated over time

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Burstable Instances (T2/T3)

- Burstable instances can be amazing to handle unexpected traffic and getting the insurance that it will be handled correctly
- If your instance consistently runs low on credit, you need to move to a different kind of non-burstable instance

Credit usage

Credit balance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CPU Credits

Instance type	Launch credits	vCPUs	CPU credits earned per hour	Maximum earned CPU credit balance	vCPUs	Baseline performance (% CPU utilization)
`t2.nano`	30	1	3	72	1	5%
`t2.micro`	30	1	6	144	1	10%
`t2.small`	30	1	12	288	1	20%
`t2.medium`	60	2	24	576	2	40% (of 200% max)*
`t2.large`	60	2	36	864	2	60% (of 200% max)*
`t2.xlarge`	120	4	54	1296	4	90% (of 400% max)*
`t2.2xlarge`	240	8	81	1944	8	135% (of 800% max)*

NOT FOR DISTRIB

iulus.com

© Stephane Maarek

T2/T3 Unlimited

- Nov 2017: It is possible to have an "unlimited burst credit balance"
- You pay extra money if you go over your credit balance, but you don't lose in performance

- Overall, it is a new offering, so be careful, costs could go high if you're not monitoring the health of your instances

- Read more here: https://aws.amazon.com/blogs/aws/new-t2-unlimited-going-beyond-the-burst-with-high-performance/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What's an AMI?

- As we saw, AWS comes with base images such as:
  - Ubuntu
  - Fedora
  - RedHat
  - Windows
  - Etc…
- These images can be customised at runtime using EC2 User data

- But what if we could create our own image, ready to go?
- That's an AMI – an image to use to create our instances
- AMIs can be built for Linux or Windows machines

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why would you use a custom AMI?

- Using a custom built AMI can provide the following advantages:
  - Pre-installed packages needed
  - Faster boot time (no need for ec2 user data at boot time)
  - Machine comes configured with monitoring / enterprise software
  - Security concerns – control over the machines in the network
  - Control of maintenance and updates of AMIs over time
  - Active Directory Integration out of the box
  - Installing your app ahead of time (for faster deploys when auto-scaling)
  - Using someone else's AMI that is optimised for running an app, DB, etc…
- AMI are built for a specific AWS region (!)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Using Public AMIs

- You can leverage AMIs from other people
- You can also pay for other people's AMI by the hour
  - These people have optimised the software
  - The machine is easy to run and configure
  - You basically rent "expertise" from the AMI creator

- AMI can be found and published on the Amazon Marketplace

- Warning:
  - Do not use an AMI you don't trust!
  - Some AMIs might come with malware or may not be secure for your enterprise

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AMI Storage

- Your AMI take space and they live in Amazon S3
- Amazon S3 is a durable, cheap and resilient storage where most of your backups will live (but you won't see them in the S3 console)
- By default, your AMIs are private, and locked for your account / region
- You can also make your AMIs public and share them with other AWS accounts or sell them on the AMI Marketplace

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AMI Pricing

- AMIs live in Amazon S3, so you get charged for the actual space in takes in Amazon S3

- Amazon S3 pricing in US-EAST-1:
  - First 50 TB / month: $0.023 per GB
  - Next 450 TB / month: $0.022 per GB

- Overall it is quite inexpensive to store private AMIs.
- Make sure to remove the AMIs you don't use

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Cross Account AMI Copy (FAQ + Exam Tip)

- You can share an AMI with another AWS account.
- Sharing an AMI does not affect the ownership of the AMI.
- If you copy an AMI that has been shared with your account, you are the owner of the target AMI in your account.
- To copy an AMI that was shared with you from another account, the owner of the source AMI must grant you read permissions for the storage that backs the AMI, either the associated EBS snapshot (for an Amazon EBS-backed AMI) or an associated S3 bucket (for an instance store-backed AMI).
- Limits:
- You can't copy an encrypted AMI that was shared with you from another account. Instead, if the underlying snapshot and encryption key were shared with you, you can copy the snapshot while re-encrypting it with a key of your own. You own the copied snapshot, and can register it as a new AMI.
- You can't copy an AMI with an associated **billingProduct** code that was shared with you from another account. This includes Windows AMIs and AMIs from the AWS Marketplace. To copy a shared AMI with a **billingProduct** code, launch an EC2 instance in your account using the shared AMI and then create an AMI from the instance.

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/CopyingAMIs.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Placement Groups

- Sometimes you want control over the EC2 Instance placement strategy
- That strategy can be defined using placement groups
- When you create a placement group, you specify one of the following strategies for the group:
  - *Cluster*—clusters instances into a low-latency group in a single Availability Zone
  - *Spread*—spreads instances across underlying hardware (max 7 instances per group per AZ)
  - *Partition*—spreads instances across many different partitions (which rely on different sets of racks) within an AZ. Scales to 100s of EC2 instances per group (Hadoop, Cassandra, Kafka)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Placement Groups
Cluster

- Pros: Great network (10 Gbps bandwidth between instances)
- Cons: If the rack fails, all instances fails at the same time
- Use case:
  - Big Data job that needs to complete fast
  - Application that needs extremely low latency and high network throughput

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Placement Groups Spread

- Pros:
  - Can span across Availability Zones (AZ)
  - Reduced risk is simultaneous failure
  - EC2 Instances are on different physical hardware
- Cons:
  - Limited to 7 instances per AZ per placement group
- Use case:
  - Application that needs to maximize high availability
  - Critical Applications where each instance must be isolated from failure from each other

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Placements Groups Partition

- Up to 7 partitions per AZ
- Up to 100s of EC2 instances
- The instances in a partition do not share racks with the instances in the other partitions
- A partition failure can affect many EC2 but won't affect other partitions
- EC2 instances get access to the partition information as metadata
- Use cases: HDFS, HBase, Cassandra, Kafka

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Elastic Network Interfaces (ENI)

- Logical component in a VPC that represents a **virtual network card**
- The ENI can have the following attributes:
  - Primary private IPv4, one or more secondary IPv4
  - One Elastic IP (IPv4) per private IPv4
  - One Public IPv4
  - One or more security groups
  - A MAC address
- You can create ENI independently and attach them on the fly (move them) on EC2 instances for failover
- Bound to a specific availability zone (AZ)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

EC2 Hibernate

- We know we can stop, terminate instances
  - Stop: the data on disk (EBS) is kept intact in the next start
  - Terminate: any EBS volumes (root) also set-up to be destroyed is lost

- On start, the following happens:
  - First start: the OS boots & the EC2 User Data script is run
  - Following starts: the OS boots up
  - Then your application starts, caches get warmed up, and that can take time!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Hibernate

- Introducing **EC2 Hibernate**:
  - The in-memory (RAM) state is preserved
  - The instance boot is much faster!
    (the OS is not stopped / restarted)
  - Under the hood: the RAM state is written to a file in the root EBS volume
  - The root EBS volume must be encrypted

- **Use cases:**
  - long-running processing
  - saving the RAM state
  - services that take time to initialize

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Hibernate.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 Hibernate – Good to know

- **Supported instance families** - C3, C4, C5, M3, M4, M5, R3, R4, and R5.
- **Instance RAM size** - must be less than 150 GB.
- **Instance size** - not supported for bare metal instances.
- **AMI:** Amazon Linux 2, Linux AMI, Ubuntu & Windows…
- **Root Volume:** must be EBS, encrypted, not instance store, and large
- Available for On-Demand and Reserved Instances

- An instance cannot be hibernated more than 60 days

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2 for Solutions Architects

- EC2 instances are billed by the second, t2.micro is free tier
- On Linux / Mac we use SSH, on Windows we use Putty
- SSH is on port 22, lock down the security group to your IP
- Timeout issues => Security groups issues
- Permission issues on the SSH key => run "chmod 0400"
- Security Groups can reference other Security Groups instead of IP ranges (very popular exam question)
- Know the difference between Private, Public and Elastic IP
- You can customize an EC2 instance at boot time using EC2 User Data

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

EC2 for Solutions Architects

- Know the 4 EC2 launch modes:
  - On demand
  - Reserved
  - Spot instances
  - Dedicated Hosts
- Know the basic instance types: R,C,M,I,G,T2/T3
- You can create AMIs to pre-install software on your EC2 => faster boot
- AMI can be copied across regions and accounts
- EC2 instances can be started in placement groups:
  - Cluster
  - Spread

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Fundamentals – Part II

Load Balancing, Auto Scaling Groups and EBS Volumes

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Scalability & High Availability

- Scalability means that an application / system can handle greater loads by adapting.
- There are two kinds of scalability:
  - Vertical Scalability
  - Horizontal Scalability (= elasticity)
- **Scalability is linked but different to High Availability**

- Let's deep dive into the distinction, using a call center as an example

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Vertical Scalability

- Vertically scalability means increasing the size of the instance
- For example, your application runs on a t2.micro
- Scaling that application vertically means running it on a t2.large
- Vertical scalability is very common for non distributed systems, such as a database.
- RDS, ElastiCache are services that can scale vertically.
- There's usually a limit to how much you can vertically scale (hardware limit)

junior operator senior operator

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Horizontal Scalability

- Horizontal Scalability means increasing the number of instances / systems for your application

- Horizontal scaling implies distributed systems.
- This is very common for web applications / modern applications

- It's easy to horizontally scale thanks the cloud offerings such as Amazon EC2

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

High Availability

- High Availability usually goes hand in hand with horizontal scaling
- High availability means running your application / system in at least 2 data centers (== Availability Zones)
- The goal of high availability is to survive a data center loss

- The high availability can be passive (for RDS Multi AZ for example)
- The high availability can be active (for horizontal scaling)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

High Availability & Scalability For EC2

- Vertical Scaling: Increase instance size (= scale up / down)
  - From: t2.nano - 0.5G of RAM, 1 vCPU
  - To: u-12tb1.metal – 12.3 TB of RAM, 448 vCPUs

- Horizontal Scaling: Increase number of instances (= scale out / in)
  - Auto Scaling Group
  - Load Balancer

- High Availability: Run instances for the same application across multi AZ
  - Auto Scaling Group multi AZ
  - Load Balancer multi AZ

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What is load balancing?

- Load balancers are servers that forward internet traffic to multiple servers (EC2 Instances) downstream.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why use a load balancer?

- Spread load across multiple downstream instances
- Expose a single point of access (DNS) to your application
- Seamlessly handle failures of downstream instances
- Do regular health checks to your instances
- Provide SSL termination (HTTPS) for your websites
- Enforce stickiness with cookies
- High availability across zones
- Separate public traffic from private traffic

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why use an EC2 Load Balancer?

- An ELB (EC2 Load Balancer) is a managed load balancer
  - AWS guarantees that it will be working
  - AWS takes care of upgrades, maintenance, high availability
  - AWS provides only a few configuration knobs

- It costs less to setup your own load balancer but it will be a lot more effort on your end.

- It is integrated with many AWS offerings / services

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Health Checks

- Health Checks are crucial for Load Balancers
- They enable the load balancer to know if instances it forwards traffic to are available to reply to requests
- The health check is done on a port and a route (/health is common)
- If the response is not 200 (OK), then the instance is unhealthy

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Types of load balancer on AWS

- AWS has 3 kinds of managed Load Balancers

- Classic Load Balancer (v1 - old generation) – 2009
  - HTTP, HTTPS, TCP
- Application Load Balancer (v2 - new generation) – 2016
  - HTTP, HTTPS, WebSocket
- Network Load Balancer (v2 - new generation) – 2017
  - TCP, TLS (secure TCP) & UDP
- Overall, it is recommended to use the newer / v2 generation load balancers as they provide more features

- You can setup internal (private) or external (public) ELBs

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Load Balancer Security Groups

Load Balancer Security Group:

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	Allow HTTP from an...
HTTPS	TCP	443	0.0.0.0/0	Allow HTTPS from a...

Application Security Group: Allow traffic only from Load Balancer

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	sg-054b5ff5ea02f2b6e (load-b	Allow Traffic only...

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Load Balancer Good to Know

- LBs can scale but not instantaneously – contact AWS for a "warm-up"
- Troubleshooting
  - 4xx errors are client induced errors
  - 5xx errors are application induced errors
  - Load Balancer Errors 503 means at capacity or no registered target
  - If the LB can't connect to your application, check your security groups!
- Monitoring
  - ELB access logs will log all access requests (so you can debug per request)
  - CloudWatch Metrics will give you aggregate statistics (ex: connections count)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Classic Load Balancers (v1)

- Supports TCP (Layer 4), HTTP & HTTPS (Layer 7)
- Health checks are TCP or HTTP based
- Fixed hostname XXX.region.elb.amazonaws.com

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Application Load Balancer (v2)

- Application load balancers is Layer 7 (HTTP)

- Load balancing to multiple HTTP applications across machines (target groups)
- Load balancing to multiple applications on the same machine (ex: containers)
- Support for HTTP/2 and WebSocket
- Support redirects (from HTTP to HTTPS for example)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Application Load Balancer (v2)

- Routing tables to different target groups:
  - Routing based on path in URL (example.com/users & example.com/posts)
  - Routing based on hostname in URL (one.example.com & other.example.com)
  - Routing based on Query String, Headers (example.com/users?id=123&order=false)

- ALB are a great fit for micro services & container-based application (example: Docker & Amazon ECS)
- Has a port mapping feature to redirect to a dynamic port in ECS
- In comparison, we'd need multiple Classic Load Balancer per application

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Application Load Balancer (v2)
HTTP Based Traffic

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Application Load Balancer (v2)
Target Groups

- EC2 instances (can be managed by an Auto Scaling Group) – HTTP
- ECS tasks (managed by ECS itself) – HTTP
- Lambda functions – HTTP request is translated into a JSON event
- IP Addresses – must be private IPs

- ALB can route to multiple target groups
- Health checks are at the target group level

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Application Load Balancer (v2) Good to Know

- Fixed hostname (XXX.region.elb.amazonaws.com)
- The application servers don't see the IP of the client directly
  - The true IP of the client is inserted in the header **X-Forwarded-For**
  - We can also get Port (X-Forwarded-Port) and proto (X-Forwarded-Proto)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Network Load Balancer (v2)

- Network load balancers (Layer 4) allow to:
  - Forward TCP & UDP traffic to your instances
  - Handle millions of request per seconds
  - Less latency ~100 ms (vs 400 ms for ALB)

- NLB has one static IP per AZ, and supports assigning Elastic IP (helpful for whitelisting specific IP)

- NLB are used for extreme performance, TCP or UDP traffic
- Not included in the AWS free tier

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Network Load Balancer (v2)
TCP (Layer 4) Based Traffic

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Load Balancer Stickiness

- It is possible to implement stickiness so that the same client is always redirected to the same instance behind a load balancer
- This works for Classic Load Balancers & Application Load Balancers
- The "cookie" used for stickiness has an expiration date you control
- Use case: make sure the user doesn't lose his session data
- Enabling stickiness may bring imbalance to the load over the backend EC2 instances

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Cross-Zone Load Balancing

- With Cross Zone Load Balancing: each load balancer instance distributes evenly across all registered instances in all AZ
- Otherwise, each load balancer node distributes requests evenly across the registered instances in its Availability Zone only.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Cross-Zone Load Balancing

- Classic Load Balancer
  - Disabled by default
  - No charges for inter AZ data if enabled
- Application Load Balancer
  - Always on (can't be disabled)
  - No charges for inter AZ data
- Network Load Balancer
  - Disabled by default
  - You pay charges ($) for inter AZ data if enabled

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSL/TLS - Basics

- An SSL Certificate allows traffic between your clients and your load balancer to be encrypted in transit (in-flight encryption)

- **SSL** refers to Secure Sockets Layer, used to encrypt connections
- **TLS** refers to Transport Layer Security, which is a newer version
- Nowadays, **TLS certificates are mainly used**, but people still refer as SSL

- Public SSL certificates are issued by Certificate Authorities (CA)
- Comodo, Symantec, GoDaddy, GlobalSign, Digicert, Letsencrypt, etc…

- SSL certificates have an expiration date (you set) and must be renewed

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Load Balancer - SSL Certificates

- The load balancer uses an X.509 certificate (SSL/TLS server certificate)
- You can manage certificates using ACM (AWS Certificate Manager)
- You can create upload your own certificates alternatively
- HTTPS listener:
  - You must specify a default certificate
  - You can add an optional list of certs to support multiple domains
  - Clients can use SNI (Server Name Indication) to specify the hostname they reach
  - Ability to specify a security policy to support older versions of SSL / TLS (legacy clients)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSL – Server Name Indication (SNI)

- SNI solves the problem of loading **multiple SSL certificates onto one web server** (to serve multiple websites)
- It's a "newer" protocol, and requires the client to **indicate** the hostname of the target server in the initial SSL handshake
- The server will then find the correct certificate, or return the default one

Note:

- Only works for ALB & NLB (newer generation), CloudFront
- Does not work for CLB (older gen)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic Load Balancers – SSL Certificates

- Classic Load Balancer (v1)
  - Support only one SSL certificate
  - Must use multiple CLB for multiple hostname with multiple SSL certificates
- Application Load Balancer (v2)
  - Supports multiple listeners with multiple SSL certificates
  - Uses Server Name Indication (SNI) to make it work
- Network Load Balancer (v2)
  - Supports multiple listeners with multiple SSL certificates
  - Uses Server Name Indication (SNI) to make it work

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ELB – Connection Draining

- Feature naming:
  - CLB: Connection Draining
  - Target Group: Deregistration Delay (for ALB & NLB)

- Time to complete "in-flight requests" while the instance is de-registering or unhealthy
- Stops sending new requests to the instance which is de-registering
- Between 1 to 3600 seconds, default is 300 seconds
- Can be disabled (set value to 0)
- Set to a low value if your requests are short

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

What's an Auto Scaling Group?

- In real-life, the load on your websites and application can change
- In the cloud, you can create and get rid of servers very quickly

- The goal of an Auto Scaling Group (ASG) is to:
  - Scale out (add EC2 instances) to match an increased load
  - Scale in (remove EC2 instances) to match a decreased load
  - Ensure we have a minimum and a maximum number of machines running
  - Automatically Register new instances to a load balancer

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Group in AWS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Group in AWS With Load Balancer

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ASGs have the following attributes

- A launch configuration
  - AMI + Instance Type
  - EC2 User Data
  - EBS Volumes
  - Security Groups
  - SSH Key Pair
- Min Size / Max Size / Initial Capacity
- Network + Subnets Information
- Load Balancer Information
- Scaling Policies

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Alarms

- It is possible to scale an ASG based on CloudWatch alarms
- An Alarm monitors a metric (such as Average CPU)
- Metrics are computed for the overall ASG instances
- Based on the alarm:
  - We can create scale-out policies (increase the number of instances)
  - We can create scale-in policies (decrease the number of instances)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling New Rules

- It is now possible to define "better" auto scaling rules that are directly managed by EC2
  - Target Average CPU Usage
  - Number of requests on the ELB per instance
  - Average Network In
  - Average Network Out
- These rules are easier to set up and can make more sense

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Custom Metric

- We can auto scale based on a custom metric (ex: number of connected users)

- 1. Send custom metric from application on EC2 to CloudWatch (PutMetric API)
- 2. Create CloudWatch alarm to react to low / high values
- 3. Use the CloudWatch alarm as the scaling policy for ASG

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ASG Brain Dump

- Scaling policies can be on CPU, Network… and can even be on custom metrics or based on a schedule (if you know your visitors patterns)
- ASGs use Launch configurations or Launch Templates (newer)
- To update an ASG, you must provide a new launch configuration / launch template
- IAM roles attached to an ASG will get assigned to EC2 instances
- ASG are free. You pay for the underlying resources being launched
- Having instances under an ASG means that if they get terminated for whatever reason, the ASG will automatically **create new ones as a replacement.** Extra safety!
- ASG can terminate instances marked as unhealthy by an LB (and hence replace them)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Groups – Scaling Policies

- Target Tracking Scaling
  - Most simple and easy to set-up
  - Example: I want the average ASG CPU to stay at around 40%
- Simple / Step Scaling
  - When a CloudWatch alarm is triggered (example CPU > 70%), then add 2 units
  - When a CloudWatch alarm is triggered (example CPU < 30%), then remove 1
- Scheduled Actions
  - Anticipate a scaling based on known usage patterns
  - Example: increase the min capacity to 10 at 5 pm on Fridays

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Auto Scaling Groups - Scaling Cooldowns

- The cooldown period helps to ensure that your Auto Scaling group doesn't launch or terminate additional instances before the previous scaling activity takes effect.
- In addition to default cooldown for Auto Scaling group, we can create cooldowns that apply to a specific simple scaling policy
- A scaling-specific cooldown period overrides the default cooldown period.
- One common use for scaling-specific cooldowns is with a scale-in policy—a policy that terminates instances based on a specific criteria or metric. Because this policy terminates instances, Amazon EC2 Auto Scaling needs less time to determine whether to terminate additional instances.
- If the default cooldown period of 300 seconds is too long—you can reduce costs by applying a scaling-specific cooldown period of 180 seconds to the scale-in policy.
- If your application is scaling up and down multiple times each hour, modify the Auto Scaling Groups cool-down timers and the CloudWatch Alarm Period that triggers the scale in

https://docs.aws.amazon.com/autoscaling/ec2/userguide/Cooldown.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ASG for Solutions Architects

- ASG Default Termination Policy (simplified version):

1. Find the AZ which has the most number of instances
2. If there are multiple instances in the AZ to choose from, delete the one with the oldest launch configuration

- ASG tries the balance the number of instances across AZ by default

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ASG for Solutions Architects
Lifecycle Hooks

- By default as soon as an instance is launched in an ASG it's in service.
- You have the ability to perform extra steps before the instance goes in service (Pending state)
- You have the ability to perform some actions before the instance is terminated (Terminating state)

https://docs.aws.amazon.com/autoscaling/ec2/userguide/lifecycle-hooks.html

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

ASG for Solutions Architect
Launch Template vs Launch Configuration

- Both:
  - ID of the Amazon Machine Image (AMI), the instance type, a key pair, security groups, and the other parameters that you use to launch EC2 instances (tags, EC2 user-data...)

- Launch Configuration (legacy):
  - Must be re-created every time

- Launch Template (newer):
  - Can have multiple versions
  - Create parameters subsets (partial configuration for re-use and inheritance)
  - Provision using both On-Demand and Spot instances (or a mix)
  - Can use T2 unlimited burst feature
  - Recommended by AWS going forward

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS & EFS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What's an EBS Volume?

- An EC2 machine loses its root volume (main drive) when it is manually terminated.
- Unexpected terminations might happen from time to time (AWS would email you)
- Sometimes, you need a way to store your instance data somewhere

- An EBS (Elastic Block Store) Volume is a network drive you can attach to your instances while they run
- It allows your instances to persist data

Amazon EBS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume

- It's a network drive (i.e. not a physical drive)
  - It uses the network to communicate the instance, which means there might be a bit of latency
  - It can be detached from an EC2 instance and attached to another one quickly
- It's locked to an Availability Zone (AZ)
  - An EBS Volume in us-east-1a cannot be attached to us-east-1b
  - To move a volume across, you first need to snapshot it
- Have a provisioned capacity (size in GBs, and IOPS)
  - You get billed for all the provisioned capacity
  - You can increase the capacity of the drive over time

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume Example

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

EBS Volume Types

- EBS Volumes come in 4 types
  - GP2 (SSD): General purpose SSD volume that balances price and performance for a wide variety of workloads
  - IO1 (SSD): Highest-performance SSD volume for mission-critical low-latency or high-throughput workloads
  - ST1 (HDD): Low cost HDD volume designed for frequently accessed, throughput-intensive workloads
  - SC1 (HDD): Lowest cost HDD volume designed for less frequently accessed workloads

- EBS Volumes are characterized in Size | Throughput | IOPS (I/O Ops Per Sec)
- When in doubt always consult the AWS documentation – it's good!
- Only GP2 and IO1 can be used as boot volumes

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume Types Use cases
GP2 (from AWS doc)

- Recommended for most workloads
- System boot volumes
- Virtual desktops
- Low-latency interactive apps
- Development and test environments

- 1 GiB - 16 TiB
- Small gp2 volumes can burst IOPS to 3000
- Max IOPS is 16,000…
- 3 IOPS per GB, means at 5,334GB we are at the max IOPS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume Types Use cases
IO1 (from AWS doc)

- Critical business applications that require sustained IOPS performance, or more than 16,000 IOPS per volume (gp2 limit)
- Large database workloads, such as:
- MongoDB, Cassandra, Microsoft SQL Server, MySQL, PostgreSQL, Oracle

- 4 GiB - 16 TiB
- IOPS is provisioned (PIOPS) – MIN 100 - MAX 64,000 (Nitro instances) else MAX 32,000 (other instances)
- The maximum ratio of provisioned IOPS to requested volume size (in GiB) is 50:1

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume Types Use cases
ST1 (from AWS doc)

- Streaming workloads requiring consistent, fast throughput at a low price.
- Big data, Data warehouses, Log processing
- Apache Kafka
- Cannot be a boot volume

- 500 GiB - 16 TiB
- Max IOPS is 500
- Max throughput of 500 MiB/s – can burst

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Volume Types Use cases
SC1 (from AWS doc)

- Throughput-oriented storage for large volumes of data that is infrequently accessed
- Scenarios where the lowest storage cost is important
- Cannot be a boot volume

- 500 GiB - 16 TiB
- Max IOPS is 250
- Max throughput of 250 MiB/s – can burst

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS – Volume Types Summary

- gp2: General Purpose Volumes (cheap)
  - 3 IOPS / GiB, minimum 100 IOPS, burst to 3000 IOPS, max 16000 IOPS
  - 1 GiB – 16 TiB , +1 TB = +3000 IOPS
- io1: Provisioned IOPS (expensive)
  - Min 100 IOPS, Max 64000 IOPS (Nitro) or 32000 (other)
  - 4 GiB - 16 TiB. Size of volume and IOPS are independent
- st1: Throughput Optimized HDD
  - 500 GiB – 16 TiB , 500 MiB /s throughput
- sc1: Cold HDD, Infrequently accessed data
  - 250 GiB – 16 TiB , 250 MiB /s throughput

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Snapshots

- Incremental – only backup changed blocks
- EBS backups use IO and you shouldn't run them while your application is handling a lot of traffic
- Snapshots will be stored in S3 (but you won't directly see them)
- Not necessary to detach volume to do snapshot, but recommended
- Max 100,000 snapshots
- Can copy snapshots across AZ or Region
- Can make Image (AMI) from Snapshot
- EBS volumes restored by snapshots need to be pre-warmed (using fio or dd command to read the entire volume)
- Snapshots can be automated using Amazon Data Lifecycle Manager

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Migration

- EBS Volumes are only locked to a specific AZ
- To migrate it to a different AZ (or region):
  - Snapshot the volume
  - (optional) Copy the volume to a different region
  - Create a volume from the snapshot in the AZ of your choice

- Let's practice!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS Encryption

- When you create an encrypted EBS volume, you get the following:
  - Data at rest is encrypted inside the volume
  - All the data in flight moving between the instance and the volume is encrypted
  - All snapshots are encrypted
  - All volumes created from the snapshot
- Encryption and decryption are handled transparently (you have nothing to do)
- Encryption has a minimal impact on latency
- EBS Encryption leverages keys from KMS (AES-256)
- Copying an unencrypted snapshot allows encryption
- Snapshots of encrypted volumes are encrypted

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Encryption: encrypt an unencrypted EBS volume

- Create an EBS snapshot of the volume
- Encrypt the EBS snapshot (using copy)
- Create new ebs volume from the snapshot (the volume will also be encrypted)
- Now you can attach the encrypted volume to the original instance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS vs Instance Store

- Some instance do not come with Root EBS volumes
- Instead, they come with "Instance Store" (= ephemeral storage)
- Instance store is physically attached to the machine (EBS is a network drive)
- Pros:
  - Better I/O performance (EBS gp2 has an max IOPS of 16000, io1 of 64000)
  - Good for buffer / cache / scratch data / temporary content
  - Data survives reboots
- Cons:
  - On stop or termination, the instance store is lost
  - You can't resize the instance store
  - Backups must be operated by the user

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Local EC2 Instance Store

- Physical disk attached to the physical server where your EC2 is
- Very High IOPS (because physical)
- Disks up to 7.5 TiB (can change over time), stripped to reach 30 TiB (can change over time…)
- Block Storage (just like EBS)
- Cannot be increased in size
- Risk of data loss if hardware fails

Very high IOPS

Instance Size	100% Random Read IOPS	Write IOPS
i3.large *	100,125	35,000
i3.xlarge *	206,250	70,000
i3.2xlarge	412,500	180,000
i3.4xlarge	825,000	360,000
i3.8xlarge	1.65 million	720,000
i3.16xlarge	3.3 million	1.4 million
i3.metal	3.3 million	1.4 million
i3en.large *	42,500	32,500
i3en.xlarge *	85,000	65,000
i3en.2xlarge *	170,000	130,000
i3en.3xlarge	250,000	200,000
i3en.6xlarge	500,000	400,000
i3en.12xlarge	1 million	800,000
i3en.24xlarge	2 million	1.6 million
i3en.metal	2 million	1.6 million

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS RAID Options

- EBS is already redundant storage (replicated within an AZ)
- But what if you want to increase IOPS to say 100 000 IOPS?
- What if you want to mirror your EBS volumes?
- You would mount volumes in parallel in RAID settings!
- RAID is possible as long as your OS supports it
- Some RAID options are:
  - RAID 0
  - RAID 1
  - RAID 5 (not recommended for EBS – see documentation)
  - RAID 6 (not recommended for EBS – see documentation)
- We'll explore RAID 0 and RAID 1

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RAID 0 (increase performance)

- Combining 2 or more volumes and getting the total disk space and I/O
- But one disk fails, all the data is failed
- Use cases would be:
  - An application that needs a lot of IOPS and doesn't need fault-tolerance
  - A database that has replication already built-in
- Using this, we can have a very big disk with a lot of IOPS
- For example
  - two 500 GiB Amazon EBS io1 volumes with 4,000 provisioned IOPS each will create a…
  - 1000 GiB RAID 0 array with an available bandwidth of 8,000 IOPS and 1,000 MB/s of throughput

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

RAID 1 (increase fault tolerance)

- RAID 1 = Mirroring a volume to another
- If one disk fails, our logical volume is still working
- We have to send the data to two EBS volume at the same time (2x network)
- Use case:
  - Application that need increase volume fault tolerance
  - Application where you need to service disks
- For example:
  - two 500 GiB Amazon EBS io1 volumes with 4,000 provisioned IOPS each will create a…
  - 500 GiB RAID 1 array with an available bandwidth of 4,000 IOPS and 500 MB/s of throughput

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

EFS – Elastic File System

- Managed NFS (network file system) that can be mounted on many EC2
- EFS works with EC2 instances in multi-AZ
- Highly available, scalable, expensive (3x gp2), pay per use

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EFS – Elastic File System

- Use cases: content management, web serving, data sharing, Wordpress
- Uses NFSv4.1 protocol
- Uses security group to control access to EFS
- Compatible with Linux based AMI (not Windows)
- Encryption at rest using KMS

- POSIX file system (~Linux) that has a standard file API
- File system scales automatically, pay-per-use, no capacity planning!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EFS – Performance & Storage Classes

- EFS Scale
  - 1000s of concurrent NFS clients, 10 GB+ /s throughput
  - Grow to Petabyte-scale network file system, automatically
- Performance mode (set at EFS creation time)
  - General purpose (default): latency-sensitive use cases (web server, CMS, etc…)
  - Max I/O – higher latency, throughput, highly parallel (big data, media processing)
- Storage Tiers (lifecycle management feature – move file after N days)
  - Standard: for frequently accessed files
  - Infrequent access (EFS-IA): cost to retrieve files, lower price to store

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS vs EFS – Elastic Block Storage

- EBS volumes…
  - can be attached to only one instance at a time
  - are locked at the Availability Zone (AZ) level
  - gp2: IO increases if the disk size increases
  - io1: can increase IO independently
- To migrate an EBS volume across AZ
  - Take a snapshot
  - Restore the snapshot to another AZ
  - EBS backups use IO and you shouldn't run them while your application is handling a lot of traffic
- Root EBS Volumes of instances get terminated by default if the EC2 instance gets terminated. (you can disable that)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EBS vs EFS – Elastic File System

- Mounting 100s of instances across AZ
- EFS share website files (WordPress)
- Only for Linux Instances (POSIX)

- EFS has a higher price point than EBS
- Can leverage EFS-IA for cost savings

- Remember: EFS vs EBS vs Instance Store

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS, Aurora & ElastiCache

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS RDS Overview

- RDS stands for Relational Database Service
- It's a managed DB service for DB use SQL as a query language.
- It allows you to create databases in the cloud that are managed by AWS
  - Postgres
  - MySQL
  - MariaDB
  - Oracle
  - Microsoft SQL Server
  - Aurora (AWS Proprietary database)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Advantage over using RDS versus deploying DB on EC2

- RDS is a managed service:
  - Automated provisioning, OS patching
  - Continuous backups and restore to specific timestamp (Point in Time Restore)!
  - Monitoring dashboards
  - Read replicas for improved read performance
  - Multi AZ setup for DR (Disaster Recovery)
  - Maintenance windows for upgrades
  - Scaling capability (vertical and horizontal)
  - Storage backed by EBS (gp2 or io1)
- BUT you can't SSH into your instances

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Backups

- Backups are automatically enabled in RDS
- Automated backups:
  - Daily full backup of the database (during the maintenance window)
  - Transaction logs are backed-up by RDS every 5 minutes
  - => ability to restore to any point in time (from oldest backup to 5 minutes ago)
  - 7 days retention (can be increased to 35 days)

- DB Snapshots:
  - Manually triggered by the user
  - Retention of backup for as long as you want

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Read Replicas for read scalability

- Up to 5 Read Replicas
- Within AZ, Cross AZ or Cross Region
- Replication is **ASYNC**, so reads are eventually consistent
- Replicas can be promoted to their own DB
- Applications must update the connection string to leverage read replicas

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Read Replicas – Use Cases

- You have a production database that is taking on normal load
- You want to run a reporting application to run some analytics
- You create a Read Replica to run the new workload there
- The production application is unaffected
- Read replicas are used for SELECT (=read) only kind of statements (not INSERT, UPDATE, DELETE)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Read Replicas – Network Cost

- In AWS there's a network cost when data goes from one AZ to another
- To reduce the cost, you can have your Read Replicas in the same AZ

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Multi AZ (Disaster Recovery)

- **SYNC** replication
- One DNS name – automatic app failover to standby
- Increase **availability**
- Failover in case of loss of AZ, loss of network, instance or storage failure
- No manual intervention in apps
- Not used for scaling

- Note: The Read Replicas be setup as Multi AZ for Disaster Recovery (DR)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Security - Encryption

- At rest encryption
  - Possibility to encrypt the master & read replicas with AWS KMS - AES-256 encryption
  - Encryption has to be defined at launch time
  - **If the master is not encrypted, the read replicas cannot be encrypted**
  - Transparent Data Encryption (TDE) available for Oracle and SQL Server

- In-flight encryption
  - SSL certificates to encrypt data to RDS in flight
  - Provide SSL options with trust certificate when connecting to database
  - To enforce SSL:
    - **PostgreSQL:** rds.force_ssl=1 in the AWS RDS Console (Parameter Groups)
    - **MySQL:** Within the DB:
      GRANT USAGE ON *.* TO 'mysqluser'@'%' **REQUIRE SSL;**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Encryption Operations

- Encrypting RDS backups
  - Snapshots of un-encrypted RDS databases are un-encrypted
  - Snapshots of encrypted RDS databases are encrypted
  - Can copy a snapshot into an encrypted one

- To encrypt an un-encrypted RDS database:
  - Create a snapshot of the un-encrypted database
  - Copy the snapshot and enable encryption for the snapshot
  - Restore the database from the encrypted snapshot
  - Migrate applications to the new database, and delete the old database

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Security – Network & IAM

- Network Security
  - RDS databases are usually deployed within a private subnet, not in a public one
  - RDS security works by leveraging security groups (the same concept as for EC2 instances) – it controls which IP / security group can **communicate** with RDS

- Access Management
  - IAM policies help control who can **manage** AWS RDS (through the RDS API)
  - Traditional Username and Password can be used to **login into** the database
  - IAM-based authentication can be used to login into RDS MySQL & PostgreSQL

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS - IAM Authentication

- IAM database authentication works with **MySQL and PostgreSQL**
- You don't need a password, just an authentication token obtained through IAM & RDS API calls
- Auth token has a lifetime of 15 minutes

- Benefits:
  - Network in/out must be encrypted using SSL
  - IAM to centrally manage users instead of DB
  - Can leverage IAM Roles and EC2 Instance profiles for easy integration

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS Security – Summary

- Encryption at rest:
  - Is done only when you first create the DB instance
  - or: unencrypted DB => snapshot => copy snapshot as encrypted => create DB from snapshot
- Your responsibility:
  - Check the ports / IP / security group inbound rules in DB's SG
  - In-database user creation and permissions or manage through IAM
  - Creating a database with or without public access
  - Ensure parameter groups or DB is configured to only allow SSL connections
- AWS responsibility:
  - No SSH access
  - No manual DB patching
  - No manual OS patching
  - No way to audit the underlying instance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon Aurora

- Aurora is a proprietary technology from AWS (not open sourced)
- Postgres and MySQL are both supported as Aurora DB (that means your drivers will work as if Aurora was a Postgres or MySQL database)
- Aurora is "AWS cloud optimized" and claims 5x performance improvement over MySQL on RDS, over 3x the performance of Postgres on RDS
- Aurora storage automatically grows in increments of 10GB, up to 64TB.
- Aurora can have 15 replicas while MySQL has 5, and the replication process is faster (sub 10 ms replica lag)
- Failover in Aurora is instantaneous. It's HA (High Availability) native.
- Aurora costs more than RDS (20% more) – but is more efficient

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Aurora High Availability and Read Scaling

- 6 copies of your data across 3 AZ:
  - 4 copies out of 6 needed for writes
  - 3 copies out of 6 need for reads
  - Self healing with peer-to-peer replication
  - Storage is striped across 100s of volumes
- One Aurora Instance takes writes (master)
- Automated failover for master in less than 30 seconds
- Master + up to 15 Aurora Read Replicas serve reads
- Support for Cross Region Replication

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Aurora DB Cluster

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Features of Aurora

- Automatic fail-over
- Backup and Recovery
- Isolation and security
- Industry compliance
- Push-button scaling
- Automated Patching with Zero Downtime
- Advanced Monitoring
- Routine Maintenance
- Backtrack: restore data at any point of time without using backups

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Aurora Security

- Similar to RDS because uses the same engines
- Encryption at rest using KMS
- Automated backups, snapshots and replicas are also encrypted
- Encryption in flight using SSL (same process as MySQL or Postgres)
- Possibility to authenticate using IAM token (same method as RDS)
- You are responsible for protecting the instance with security groups
- You can't SSH

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Aurora Serverless

- Automated database instantiation and auto-scaling based on actual usage
- Good for infrequent, intermittent or unpredictable workloads
- No capacity planning needed
- Pay per second, can be more cost-effective

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Global Aurora

- Aurora Cross Region Read Replicas:
  - Useful for disaster recovery
  - Simple to put in place
- Aurora Global Database (recommended):
  - 1 Primary Region (read / write)
  - Up to 5 secondary (read-only) regions, replication lag is less than 1 second
  - Up to 16 Read Replicas per secondary region
  - Helps for decreasing latency
  - Promoting another region (for disaster recovery) has an RTO of < 1 minute

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Amazon ElastiCache Overview

- The same way RDS is to get managed Relational Databases…
- ElastiCache is to get managed Redis or Memcached
- Caches are in-memory databases with really high performance, low latency
- Helps reduce load off of databases for read intensive workloads
- Helps make your application stateless
- AWS takes care of OS maintenance / patching, optimizations, setup, configuration, monitoring, failure recovery and backups
- <u>Using ElastiCache involves heavy application code changes</u>

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElastiCache
Solution Architecture - DB Cache

- Applications queries ElastiCache, if not available, get from RDS and store in ElastiCache.
- Helps relieve load in RDS
- Cache must have an invalidation strategy to make sure only the most current data is used in there.

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

ElastiCache
Solution Architecture – User Session Store

- User logs into any of the application
- The application writes the session data into ElastiCache
- The user hits another instance of our application
- The instance retrieves the data and the user is already logged in

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

ElastiCache – Redis vs Memcached

REDIS

- **Multi AZ** with Auto-Failover
- **Read Replicas** to scale reads and have **high availability**
- Data Durability using AOF persistence
- **Backup and restore features**

MEMCACHED

- Multi-node for partitioning of data (sharding)
- **Non persistent**
- **No backup and restore**
- Multi-threaded architecture

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElastiCache – Cache Security

- All caches in ElastiCache:
  - Support SSL in flight encryption
  - <u>Do not support IAM authentication</u>
  - IAM policies on ElastiCache are only used for AWS API-level security
- Redis AUTH
  - You can set a "password/token" when you create a Redis cluster
  - This is an extra level of security for your cache (on top of security groups)
- Memcached
  - Supports SASL-based authentication (advanced)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElastiCache for Solutions Architects

- Patterns for ElastiCache
  - **Lazy Loading:** all the read data is cached, data can become stale in cache
  - **Write Through:** Adds or update data in the cache when written to a DB (no stale data)
  - **Session Store:** store temporary session data in a cache (using TTL features)

- *Quote: There are only two hard things in Computer Science: cache invalidation and naming things*

Lazy Loading illustrated

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Route 53

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Route 53 Section

- TTL
- CNAME vs Alias
- Health Checks
- Routing Policies
  - Simple
  - Weighted
  - Latency
  - Failover
  - Geolocation
  - Multi Value
- 3rd party domains integration

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Route 53 Overview

- Route53 is a Managed DNS (Domain Name System)
- DNS is a collection of rules and records which helps clients understand how to reach a server through URLs.
- In AWS, the most common records are:
  - A: URL to IPv4
  - AAAA: URL to IPv6
  - CNAME: URL to URL
  - Alias: URL to AWS resource.

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Route 53 – Diagram for A Record

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Route 53 Overview

- Route53 can use:
  - public domain names you own (or buy)
    application1.mypublicdomain.com
  - private domain names that can be resolved by your instances in your VPCs.
    application1.company.internal

- Route53 has advanced features such as:
  - Load balancing (through DNS – also called client load balancing)
  - Health checks (although limited…)
  - Routing policy: simple, failover, geolocation, latency, weighted, multi value

- You pay $0.50 per month per hosted zone

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DNS Records TTL (Time to Live)

DNS Cache
For TTL duration

Web browser

DNS Request
http://myapp.mydomain.com

Send back IP: 32.45.67.85
(A record: URL to IP)
+ TTL : 300 s

DNS Request
http://myapp.mydomain.com

Send back IP: 195.23.45.22
(A record: URL to IP)
+ TTL : 300 s

Route 53

- High TTL: (e.g. 24hr)
  - Less traffic on DNS
  - Possibly outdated records
- Low TTL: (e.g 60 s)
  - More traffic on DNS
  - Records are outdated for less time
  - Easy to change records
- TTL is mandatory for each DNS record

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

CNAME vs Alias

- AWS Resources (Load Balancer, CloudFront, etc…) expose an AWS URL: lb1-1234.us-east-2.elb.amazonaws.com and you want it to be myapp.mydomain.com
- CNAME:
  - Points a URL to any other URL. (app.mydomain.com => blabla.anything.com)
  - ONLY FOR NON ROOT DOMAIN (aka. something.mydomain.com)
- Alias:
  - Points a URL to an AWS Resource (app.mydomain.com => blabla.amazonaws.com)
  - Works for ROOT DOMAIN and NON ROOT DOMAIN (aka mydomain.com)
  - Free of charge
  - Native health check

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Simple Routing Policy

- Maps a domain to one URL
- Use when you need to redirect to a single resource
- You can't attach health checks to simple routing policy

- If multiple values are returned, a random one is chosen by the client

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Weighted Routing Policy

- Control the % of the requests that go to specific endpoint
- Helpful to test 1% of traffic on new app version for example
- Helpful to split traffic between two regions
- Can be associated with Health Checks

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Latency Routing Policy

- Redirect to the server that has the least latency close to us
- Super helpful when latency of users is a priority
- **Latency is evaluated in terms of user to designated AWS Region**
- Germany may be directed to the US (if that's the lowest latency)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Health Checks

- Have X health checks failed => unhealthy (default 3)
- After X health checks passed => health (default 3)
- Default Health Check Interval: 30s (can set to 10s – higher cost)
- **About 15 health checkers will check the endpoint health**
- => one request every 2 seconds on average
- Can have HTTP,TCP and HTTPS health checks (no SSL verification)
- Possibility of integrating the health check with CloudWatch

- **Health checks can be linked to Route53 DNS queries!**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Failover Routing Policy

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Geo Location Routing Policy

- Different from Latency based!
- **This is routing based on user location**
- Here we specify: traffic from the UK should go to this specific IP
- Should create a "default" policy (in case there's no match on location)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Multi Value Routing Policy

- Use when routing traffic to multiple resources
- Want to associate a Route 53 health checks with records
- Up to 8 healthy records are returned for each Multi Value query
- Multi Value is not a substitute for having an ELB

Name	Type	Value	TTL	Set ID	Health Check
www.example.com	A Record	192.0.2.2	60	Web1	A
www.example.com	A Record	198.51.100.2	60	Web2	B
www.example.com	A Record	203.0.113.2	60	Web3	C

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Route53 as a Registrar

- A **domain** name **registrar** is an organization that manages the reservation of Internet **domain** names
- Famous names:
  - GoDaddy
  - Google Domains
  - Etc…
- And also… Route53 (e.g. AWS)!

- Domain Registrar != DNS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

3rd Party Registrar with AWS Route 53

- If you buy your domain on 3rd party website, you can still use Route53.

- 1) Create a Hosted Zone in Route 53
- 2) Update NS Records on 3rd party website to use Route 53 name servers

- Domain Registrar != DNS
- (But each domain registrar usually comes with some DNS features)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Classic Solutions Architecture

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Introduction

- These solutions architectures are the best part of this course
- Let's understand how all the technologies we've seen work together
- This is a section you need to be 100% comfortable with

- We'll see the progression of a Solution's architect mindset through many sample case studies:
  - WhatIsTheTime.Com
  - MyClothes.Com
  - MyWordPress.Com
  - Instantiating applications quickly
  - Beanstalk

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless Web App: WhatIsTheTime.com

- WhatIsTheTime.com allows people to know what time it is
- We don't need a database
- We want to start small and can accept downtime
- We want to fully scale vertically and horizontally, no downtime
- Let's go through the Solutions Architect journey for this app

- Let's see how we can proceed!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless web app: What time is it? Starting simple

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateless web app: What time is it? Scaling vertically

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless web app: What time is it? Scaling horizontally

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless web app: What time is it? Scaling horizontally

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateless web app: What time is it?
Scaling horizontally, adding and removing instances

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateless web app: What time is it? Scaling horizontally, with a load balancer

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless web app: What time is it?
Scaling horizontally, with an auto-scaling group

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateless web app: What time is it? Making our app multi-AZ

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Minimum 2 AZ => Let's reserve capacity

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

In this lecture we've discussed…

- Public vs Private IP and EC2 instances
- Elastic IP vs Route 53 vs Load Balancers
- Route 53 TTL, A records and Alias Records
- Maintaining EC2 instances manually vs Auto Scaling Groups
- Multi AZ to survive disasters
- ELB Health Checks
- Security Group Rules
- Reservation of capacity for costing savings when possible
- We're considering 5 pillars for a well architected application: costs, performance, reliability, security, operational excellence

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com

- MyClothes.com allows people to buy clothes online.
- There's a shopping cart
- Our website is having hundreds of users at the same time
- We need to scale, maintain horizontal scalability and keep our web application as stateless as possible
- Users should not lose their shopping cart
- Users should have their details (address, etc) in a database

- Let's see how we can proceed!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com
Introduce Stickiness (Session Affinity)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com
Introduce User Cookies

Stateless
HTTP requests are heavier
Security risk
(cookies can be altered)
Cookies must be validated
Cookies must be less than 4KB

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateful Web App: MyClothes.com
Introduce Server Session

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateful Web App: MyClothes.com
Storing User Data in a database

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com
Scaling Reads

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com
Scaling Reads (Alternative) – Write Through

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyClothes.com
Multi AZ – Survive disasters

ElastiCache
Multi AZ

RDS
Multi AZ

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateful Web App: MyClothes.com Security Groups

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

In this lecture we've discussed… 3-tier architectures for web applications

- ELB sticky sessions
- Web clients for storing cookies and making our web app stateless
- ElastiCache
  - For storing sessions (alternative: DynamoDB)
  - For caching data from RDS
  - Multi AZ
- RDS
  - For storing user data
  - Read replicas for scaling reads
  - Multi AZ for disaster recovery
- Tight Security with security groups referencing each other

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyWordPress.com

- We are trying to create a fully scalable WordPress website
- We want that website to access and correctly display picture uploads
- Our user data, and the blog content should be stored in a MySQL database.

- Let's see how we can achieve this!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyWordPress.com RDS layer

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyWordPress.com
Scaling with Aurora: Multi AZ & Read Replicas

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Stateful Web App: MyWordPress.com
Storing images with EBS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyWordPress.com Storing images with EBS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Stateful Web App: MyWordPress.com
Storing images with EFS

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

In this lecture we've discussed…

- Aurora Database to have easy Multi-AZ and Read-Replicas
- Storing data in EBS (single instance application)
- Vs Storing data in EFS (distributed application)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Instantiating Applications quickly

- When launching a full stack (EC2, EBS, RDS), it can take time to:
  - Install applications
  - Insert initial (or recovery) data
  - Configure everything
  - Launch the application

- We can take advantage of the cloud to speed that up!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Instantiating Applications quickly

- EC2 Instances:
  - **Use a Golden AMI:** Install your applications, OS dependencies etc.. beforehand and launch your EC2 instance from the Golden AMI
  - **Bootstrap using User Data:** For dynamic configuration, use User Data scripts
  - **Hybrid:** mix Golden AMI and User Data (Elastic Beanstalk)
- RDS Databases:
  - Restore from a snapshot: the database will have schemas and data ready!
- EBS Volumes:
  - Restore from a snapshot: the disk will already be formatted and have data!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Typical architecture Web App 3-tier

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Developer problems on AWS

- Managing infrastructure
- Deploying Code
- Configuring all the databases, load balancers, etc
- Scaling concerns

- Most web apps have the same architecture (ALB + ASG)
- All the developers want is for their code to run!
- Possibly, consistently across different applications and environments

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Elastic Beanstalk Overview

- Elastic Beanstalk is a developer centric view of deploying an application on AWS

- It uses all the component's we've seen before: EC2, ASG, ELB, RDS, etc…
- But it's all in one view that's easy to make sense of!
- We still have full control over the configuration

- Beanstalk is free but you pay for the underlying instances

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic Beanstalk

- Managed service
  - Instance configuration / OS is handled by Beanstalk
  - Deployment strategy is configurable but performed by Elastic Beanstalk
- Just the application code is the responsibility of the developer
- Three architecture models:
  - Single Instance deployment: good for dev
  - LB + ASG: great for production or pre-production web applications
  - ASG only: great for non-web apps in production (workers, etc..)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic Beanstalk

- Elastic Beanstalk has three components
  - Application
  - Application version: each deployment gets assigned a version
  - Environment name (dev, test, prod…): free naming

- You deploy application versions to environments and can promote application versions to the next environment
- Rollback feature to previous application version
- Full control over lifecycle of environments

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Elastic Beanstalk

- Support for many platforms:
  - Go
  - Java SE
  - Java with Tomcat
  - .NET on Windows Server with IIS
  - Node.js
  - PHP
  - Python
  - Ruby
  - Packer Builder
  - Single Container Docker
  - Multicontainer Docker
  - Preconfigured Docker
- If not supported, you can write your custom platform (advanced)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Storage and Data Management

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section introduction

- Amazon S3 is one of the main building blocks of AWS
- It's advertised as "infinitely scaling" storage
- It's widely popular and deserves its own section

- Many websites use AWS S3 as a backbone
- Many AWS services uses AWS S3 as an integration as well

- We'll have a step-by-step approach to S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS S3 Overview - Buckets

- Amazon S3 allows people to store objects (files) in "buckets" (directories)
- Buckets must have a **globally unique name**
- Buckets are defined at the region level
- Naming convention
  - No uppercase
  - No underscore
  - 3-63 characters long
  - Not an IP
  - Must start with lowercase letter or number

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS S3 Overview - Objects

- Objects (files) have a Key. The key is the **FULL** path:
  - <my_bucket>/my_file.txt
  - <my_bucket>/my_folder1/another_folder/my_file.txt
- There's no concept of "directories" within buckets (although the UI will trick you to think otherwise)
- Just keys with very long names that contain slashes ("/")
- Object Values are the content of the body:
  - Max Size is 5TB
  - If uploading more than 5GB, must use "multi-part upload"
- Metadata (list of text key / value pairs – system or user metadata)
- Tags (Unicode key / value pair – up to 10) – useful for security / lifecycle
- Version ID (if versioning is enabled)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS S3 - Versioning

- You can version your files in AWS S3
- It is enabled at the **bucket level**
- Same key overwrite will increment the "version": 1, 2, 3….
- It is best practice to version your buckets
  - Protect against unintended deletes (ability to restore a version)
  - Easy roll back to previous version
- Any file that is not versioned prior to enabling versioning will have version "null"

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Encryption for Objects

- There are 4 methods of encrypting objects in S3
  - SSE-S3: encrypts S3 objects using keys handled & managed by AWS
  - SSE-KMS: leverage AWS Key Management Service to manage encryption keys
  - SSE-C: when you want to manage your own encryption keys
  - Client Side Encryption
- It's important to understand which ones are adapted to which situation for the exam

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSE-S3

- SSE-S3: encryption using keys handled & managed by AWS S3
- Object is encrypted server side
- AES-256 encryption type
- Must set header: "x-amz-server-side-encryption": "AES256"

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSE-KMS

- SSE-KMS: encryption using keys handled & managed by KMS
- KMS Advantages: user control + audit trail
- Object is encrypted server side
- Must set header: "x-amz-server-side-encryption": "aws:kms"

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSE-C

- SSE-C: server-side encryption using data keys fully managed by the customer outside of AWS
- Amazon S3 does not store the encryption key you provide
- HTTPS must be used
- Encryption key must provided in HTTP headers, for every HTTP request made

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Client Side Encryption

- Client library such as the Amazon S3 Encryption Client
- Clients must encrypt data themselves before sending to S3
- Clients must decrypt data themselves when retrieving from S3
- Customer fully manages the keys and encryption cycle

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Encryption in transit (SSL)

- AWS S3 exposes:
  - HTTP endpoint: non encrypted
  - HTTPS endpoint: encryption in flight

- You're free to use the endpoint you want, but HTTPS is recommended
- HTTPS is mandatory for SSE-C
- Encryption in flight is also called SSL / TLS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Security

- User based
  - IAM policies - which API calls should be allowed for a specific user from IAM console

- Resource Based
  - Bucket Policies - bucket wide rules from the S3 console - allows cross account
  - Object Access Control List (ACL) – finer grain
  - Bucket Access Control List (ACL) – less common

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Bucket Policies

- JSON based policies
  - Resources: buckets and objects
  - Actions: Set of API to Allow or Deny
  - Effect: Allow / Deny
  - Principal: The account or user to apply the policy to
- Use S3 bucket for policy to:
  - Grant public access to the bucket
  - Force objects to be encrypted at upload
  - Grant access to another account (Cross Account)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Security - Other

- Networking:
  - Supports VPC Endpoints (for instances in VPC without www internet)
- Logging and Audit:
  - S3 access logs can be stored in other S3 bucket
  - API calls can be logged in AWS CloudTrail
- User Security:
  - MFA (multi factor authentication) can be required in versioned buckets to delete objects
  - Signed URLs: URLs that are valid only for a limited time (ex: premium video service for logged in users)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Websites

- S3 can host static websites and have them accessible on the www
- The website URL will be:
  - <bucket-name>.s3-website-<AWS-region>.amazonaws.com
  - OR
  - <bucket-name>.s3-website.<AWS-region>.amazonaws.com
- If you get a 403 (Forbidden) error, make sure the bucket policy allows public reads!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 CORS

- If you request data from another S3 bucket, you need to enable CORS
- Cross Origin Resource Sharing allows you to limit the number of websites that can request your files in S3 (and limit your costs)
- It's a popular exam question

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS S3 - Consistency Model

- Read after write consistency for PUTS of new objects
  - As soon as an object is written, we can retrieve it
    ex: (PUT 200 -> GET 200)
  - This is true, **except** if we did a GET before to see if the object existed
    ex: (GET 404 -> PUT 200 -> GET 404) – eventually consistent

- Eventual Consistency for DELETES and PUTS of existing objects
  - If we read an object after updating, we might get the older version
    ex: (PUT 200 -> PUT 200 -> GET 200 (might be older version))
  - If we delete an object, we might still be able to retrieve it for a short time
    ex: (DELETE 200 -> GET 200)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Developing on AWS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Introduction

- So far, we've interacts with services manually and they exposed standard information for clients:
  - EC2 exposes a standard Linux machine we can use any way we want
  - RDS exposes a standard database we can connect to using a URL
  - ElastiCache exposes a cache URL we can connect to using a URL
  - ASG / ELB are automated and we don't have to program against them
  - Route53 was setup manual

- Developing against AWS has two components:
  - How to perform interactions with AWS without using the Online Console?
  - How to interact with AWS Proprietary services? (S3, DynamoDB, etc…)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Introduction

- Developing and performing AWS tasks against AWS can be done in several ways
  - Using the AWS CLI on our local computer
  - Using the AWS CLI on our EC2 machines
  - Using the AWS SDK on our local computer
  - Using the AWS SDK on our EC2 machines
  - Using the AWS Instance Metadata Service for EC2

- In this section, we'll learn:
  - How to do all of those
  - In the right & most secure way, adhering to best practices

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI Setup Windows

- We'll setup the CLI properly on Windows

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI Setup Mac OS X

- We'll setup the CLI properly on Mac OS X

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI Setup Linux

- We'll setup the CLI properly on Linux

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI Configuration

- Let's learn how to properly configure the CLI

- We'll learn how to get our access credentials and protect them
- Do not share your AWS Access Key and Secret key with anyone!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI ON EC2… THE **BAD** WAY

- We could run `aws configure` on EC2 just like we did (and it'll work)
- But… it's SUPER INSECURE
- NEVER EVER EVER PUT YOUR PERSONAL CREDENTIALS ON AN EC2
- Your **PERSONAL** credentials are **PERSONAL** and only belong on your **PERSONAL** computer

- If the EC2 is compromised, so is your personal account
- If the EC2 is shared, other people may perform AWS actions while impersonating you

- For EC2, there's a better way… it's called AWS IAM Roles

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CLI ON EC2… THE RIGHT WAY

- IAM Roles can be attached to EC2 instances
- IAM Roles can come with a policy authorizing exactly what the EC2 instance should be able to do

- EC2 Instances can then use these profiles automatically without any additional configurations
- This is the best practice on AWS and you should 100% do this.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS EC2 Instance Metadata

- AWS EC2 Instance Metadata is powerful but one of the least known features to developers
- It allows AWS EC2 instances to "learn about themselves" **without using an IAM Role for that purpose.**
- The URL is http://169.254.169.254/latest/meta-data
- You can retrieve the IAM Role name from the metadata, but you CANNOT retrieve the IAM Policy.
- Metadata = Info about the EC2 instance
- Userdata = launch script of the EC2 instance

- Let's practice and see what we can do with it!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SDK Overview

- What if you want to perform actions on AWS directly from your applications code ? (without using the CLI).
- You can use an SDK (software development kit) !
- Official SDKs are…
  - Java
  - .NET
  - Node.js
  - PHP
  - Python (named boto3 / botocore)
  - Go
  - Ruby
  - C++

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SDK Overview

- We have to use the AWS SDK when coding against AWS Services such as DynamoDB

- Fun fact… the AWS CLI uses the Python SDK (boto3)
- The exam expects you to know when you should use an SDK
- We'll practice the AWS SDK when we get to the Lambda functions

- Good to know: if you don't specify or configure a default region, then us-east-1 will be chosen by default

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SDK Credentials Security

- It's recommend to use the **default credential provider chain**
- The **default credential provider chain** works seamlessly with:
  - AWS credentials at ~/.aws/credentials (only on our computers or on premise)
  - Instance Profile Credentials using IAM Roles (for EC2 machines, etc…)
  - Environment variables (AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY)

- Overall, NEVER EVER STORE AWS CREDENTIALS IN YOUR CODE.
- Best practice is for credentials to be inherited from mechanisms above, and 100% IAM Roles if working from within AWS Services

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Exponential Backoff

- Any API that fails because of too many calls needs to be retried with Exponential Backoff
- These apply to rate limited API
- Retry mechanism included in SDK API calls

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Advanced S3

S3, Glacier, Athena

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 MFA-Delete

- MFA (multi factor authentication) forces user to generate a code on a device (usually a mobile phone or hardware) before doing important operations on S3

- To use MFA-Delete, enable Versioning on the S3 bucket
- You will need MFA to
  - permanently delete an object version
  - suspend versioning on the bucket
- You won't need MFA for
  - enabling versioning
  - listing deleted versions

- Only the bucket owner (root account) can enable/disable MFA-Delete
- MFA-Delete currently can only be enabled using the CLI

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Default Encryption vs Bucket Policies

- The old way to enable default encryption was to use a bucket policy and refuse any HTTP command without the proper headers:

```
{
    "Version": "2012-10-17",
    "Id": "PutObjPolicy",
    "Statement": [
          {
                "Sid": "DenyIncorrectEncryptionHeader",
                "Effect": "Deny",
                "Principal": "*",
                "Action": "s3:PutObject",
                "Resource": "arn:aws:s3:::<bucket_name>/*",
                "Condition": {
                        "StringNotEquals": {
                                "s3:x-amz-server-side-encryption": "AES256"
                         }
                }
          },
```

```
          {
                "Sid": "DenyUnEncryptedObjectUploads",
                "Effect": "Deny",
                "Principal": "*",
                "Action": "s3:PutObject",
                "Resource": "arn:aws:s3:::<bucket_name>/*",
                "Condition": {
                        "Null": {
                                "s3:x-amz-server-side-encryption": true
                        }
                }
          }
    ]
}
```

- The new way is to use the "default encryption" option in S3
- Note: Bucket Policies are evaluated before "default encryption"

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Access Logs

- For audit purpose, you may want to log all access to S3 buckets
- Any request made to S3, from any account, authorized or denied, will be logged into another S3 bucket
- That data can be analyzed using data analysis tools…
- Or Amazon Athena as we'll see later in this section!

- The log format is at: https://docs.aws.amazon.com/AmazonS3/latest/dev/LogFormat.html

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

S3 Cross Region Replication

- Must enable versioning (source and destination)
- Buckets must be in different AWS regions
- Can be in different accounts
- Copying is asynchronous
- Must give proper IAM permissions to S3

- Use cases: compliance, lower latency access, replication across accounts

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 pre-signed URLs

- Can generate pre-signed URLs using SDK or CLI
  - For downloads (easy, can use the CLI)
  - For uploads (harder, must use the SDK)
- Valid for a default of 3600 seconds, can change timeout with --expires-in [TIME_BY_SECONDS] argument
- Users given a pre-signed URL inherit the permissions of the person who generated the URL for GET / PUT

- Examples :
  - Allow only logged-in users to download a premium video on your S3 bucket
  - Allow an ever changing list of users to download files by generating URLs dynamically
  - Allow temporarily a user to upload a file to a precise location in our bucket

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Storage Classes

- Amazon S3 Standard - General Purpose
- Amazon S3 Standard-Infrequent Access (IA)
- Amazon S3 One Zone-Infrequent Access
- Amazon S3 Intelligent Tiering
- Amazon Glacier
- Amazon Glacier Deep Archive

- Amazon S3 Reduced Redundancy Storage (deprecated - omitted)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Standard – General Purpose

- High durability (99.999999999%) of objects across multiple AZ
- If you store 10,000,000 objects with Amazon S3, you can on average expect to incur a loss of a single object once every 10,000 years
- 99.99% Availability over a given year
- Sustain 2 concurrent facility failures

- Use Cases: Big Data analytics, mobile & gaming applications, content distribution…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Standard – Infrequent Access (IA)

- Suitable for data that is less frequently accessed, but requires rapid access when needed
- High durability (99.999999999%) of objects across multiple AZs
- 99.9% Availability
- Low cost compared to Amazon S3 Standard
- Sustain 2 concurrent facility failures

- Use Cases: As a data store for disaster recovery, backups…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 One Zone - Infrequent Access (IA)

- Same as IA but data is stored in a single AZ
- High durability (99.999999999%) of objects in a single AZ; data lost when AZ is destroyed
- 99.5% Availability
- Low latency and high throughput performance
- Supports SSL for data at transit and encryption at rest
- Low cost compared to IA (by 20%)

- Use Cases: Storing secondary backup copies of on-premise data, or storing data you can recreate

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Intelligent Tiering

- Same low latency and high throughput performance of S3 Standard
- Small monthly monitoring and auto-tiering fee
- Automatically moves objects between two access tiers based on changing access patterns
- Designed for durability of 99.999999999% of objects across multiple Availability Zones
- Resilient against events that impact an entire Availability Zone
- Designed for 99.9% availability over a given year

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon Glacier

- Low cost object storage meant for archiving / backup
- Data is retained for the longer term (10s of years)
- Alternative to on-premise magnetic tape storage
- Average annual durability is 99.999999999%
- Cost per storage per month ($0.004 / GB) + retrieval cost
- Each item in Glacier is called "**Archive**" (up to 40TB)
- Archives are stored in "**Vaults**"

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon Glacier & Glacier Deep Archive

- Amazon Glacier – 3 retrieval options:
  - Expedited (1 to 5 minutes)
  - Standard (3 to 5 hours)
  - Bulk (5 to 12 hours)
  - Minimum storage duration of 90 days

- Amazon Glacier Deep Archive – for long term storage – cheaper:
  - Standard (12 hours)
  - Bulk (48 hours)
  - Minimum storage duration of 180 days

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Storage Classes Comparison

	S3 Standard	S3 Intelligent-Tiering	S3 Standard-IA	S3 One Zone-IA	S3 Glacier	S3 Glacier Deep Archive
Designed for durability	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)	99.999999999% (11 9's)
Designed for availability	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability SLA	99.9%	99%	99%	99%	99.9%	99.9%
Availability Zones	≥3	≥3	≥3	1	≥3	≥3
Minimum capacity charge per object	N/A	N/A	128KB	128KB	40KB	40KB
Minimum storage duration charge	N/A	30 days	30 days	30 days	90 days	180 days
Retrieval fee	N/A	N/A	per GB retrieved	per GB retrieved	per GB retrieved	per GB retrieved

https://aws.amazon.com/s3/storage-classes/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Storage Classes – Price Comparison Example us-east-2

	S3 Standard	S3 Intelligent-Tiering	S3 Standard-IA	S3 One Zone-IA	S3 Glacier	S3 Glacier Deep Archive
Storage Cost (per GB per month)	$0.023	$0.0125 - $0.023	$0.0125	$0.01	$0.004 Minimum 90 days	$0.00099 Minimum 180 days
Retrieval Cost (per 1000 requests)	GET $0.0004	GET $0.0004	GET $0.001	GET $0.001	GET $0.0004 + Expedited - $10.00 Standard - $0.05 Bulk - $0.025	GET $0.0004 + Standard - $0.10 Bulk - $0.025
Time to retrieve	instantaneous	Instantaneous	Instantaneous	Instantaneous	Expedited (1 to 5 minutes) Standard (3 to 5 hours) Bulk (5 to 12 hours)	Standard (12 hours) Bulk (48 hours)
Monitoring Cost (per 1000 objects)		$0.0025				

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 – Moving between storage classes

- You can transition objects between storage classes
- For infrequently accessed object, move them to STANDARD_IA
- For archive objects you don't need in real-time, GLACIER or DEEP_ARCHIVE
- Moving objects can be automated using a **lifecycle configuration**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Lifecycle Rules

- **Transition actions**: It defines when objects are transitioned to another storage class.
  - Move objects to Standard IA class 60 days after creation
  - Move to Glacier for archiving after 6 months

- **Expiration actions:** configure objects to expire (delete) after some time
  - Access log files can be set to delete after a 365 days
  - Can be used to delete old versions of files (if versioning is enabled)
  - Can be used to delete incomplete multi-part uploads

- Rules can be created for a certain prefix (ex - s3://mybucket/mp3/*)
- Rules can be created for certain objects tags (ex - Department: Finance)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Lifecycle Rules – Scenario 1

- Your application on EC2 creates images thumbnails after profile photos are uploaded to Amazon S3. These thumbnails can be easily recreated, and only need to be kept for 45 days. The source images should be able to be immediately retrieved for these 45 days, and afterwards, the user can wait up to 6 hours. How would you design this?

- S3 source images can be on STANDARD, with a lifecycle configuration to transition them to GLACIER after 45 days.
- S3 thumbnails can be on ONEZONE_IA, with a lifecycle configuration to expire them (delete them) after 45 days.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Lifecycle Rules – Scenario 2

- A rule in your company states that you should be able to recover your deleted S3 objects immediately for 15 days, although this may happen rarely. After this time, and for up to 365 days, deleted objects should be recoverable within 48 hours.

- You need to enable S3 versioning in order to have object versions, so that "deleted objects" are in fact hidden by a "delete marker" and can be recovered
- You can transition these "noncurrent versions" of the object to S3_IA
- You can transition afterwards these "noncurrent versions" to DEEP_ARCHIVE

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 – Baseline Performance

- Amazon S3 automatically scales to high request rates, latency 100-200 ms
- Your application can achieve at least **3,500 PUT/COPY/POST/DELETE and 5,500 GET/HEAD requests per second per prefix in a bucket.**
- There are no limits to the number of prefixes in a bucket.
- Example (object path => prefix):
  - bucket/folder1/sub1/file => /folder1/sub1/
  - bucket/folder1/sub2/file => /folder1/sub2/
  - bucket/1/file => /1/
  - bucket/2/file => /2/
- If you spread reads across all four prefixes evenly, you can achieve 22,000 requests per second for GET and HEAD

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 – KMS Limitation

- If you use SSE-KMS, you may be impacted by the KMS limits
- When you upload, it calls the **GenerateDataKey** KMS API
- When you download, it calls the **Decrypt** KMS API
- Count towards the KMS quota per second (5500, 10000, 30000 req/s based on region)
- As of today, you cannot request a quota increase for KMS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Performance

- Multi-Part upload:
  - recommended for files > 100MB, must use for files > 5GB
  - Can help parallelize uploads (speed up transfers)

- S3 Transfer Acceleration (upload only)
  - Increase transfer speed by transferring file to an AWS edge location which will forward the data to the S3 bucket in the target region
  - Compatible with multi-part upload

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Performance – S3 Byte-Range Fetches

- Parallelize GETs by requesting specific byte ranges
- Better resilience in case of failures

Can be used to speed up downloads

Can be used to retrieve only partial data (for example the head of a file)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Select & Glacier Select

- Retrieve less data using SQL by performing **server side filtering**
- Can filter by rows & columns (simple SQL statements)
- Less network transfer, less CPU cost client-side

https://aws.amazon.com/blogs/aws/s3-glacier-select/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Athena

- **Serverless** service to perform analytics **directly against S3 files**
- Uses SQL language to query the files
- Has a JDBC / ODBC driver
- Charged per query and amount of data scanned
- Supports CSV, JSON, ORC, Avro, and Parquet (built on Presto)
- Use cases: Business intelligence / analytics / reporting, analyze & query VPC Flow Logs, ELB Logs, CloudTrail trails, etc...
- Exam Tip: Analyze data directly on S3 => use Athena

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Object Lock & Glacier Vault Lock

- S3 Object Lock
  - Adopt a WORM (Write Once Read Many) model
  - Block an object version deletion for a specified amount of time

- Glacier Vault Lock
  - Adopt a WORM (Write Once Read Many) model
  - Lock the policy for future edits (can no longer be changed)
  - Helpful for compliance and data retention

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudFront

- Content Delivery Network (CDN)
- Improves read performance, content is cached at the edge
- 216 Point of Presence globally (edge locations)
- DDoS protection, integration with Shield, AWS Web Application Firewall
- Can expose external HTTPS and can talk to internal HTTPS backends

Source: https://aws.amazon.com/cloudfront/features/?nc=sn&loc=2

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

CloudFront – Origins

- S3 bucket
  - For distributing files and caching them at the edge
  - Enhanced security with CloudFront **Origin Access Identity (OAI)**
  - CloudFront can be used as an ingress (to upload files to S3)
- S3 website
  - Must first enabled the bucket as a static S3 website
- Custom Origin (HTTP) – Must be publicly accessible
  - Application Load Balancer
  - EC2 instance
  - Any HTTP backend you want

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront – S3 as an Origin

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront – ALB or EC2 as an origin

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront Geo Restriction

- You can restrict who can access your distribution
  - **Whitelist:** Allow your users to access your content only if they're in one of the countries on a list of approved countries.
  - **Blacklist:** Prevent your users from accessing your content if they're in one of the countries on a blacklist of banned countries.

- The "country" is determined using a 3rd party Geo-IP database
- Use case: Copyright Laws to control access to content

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront vs S3 Cross Region Replication

- CloudFront:
  - Global Edge network
  - Files are cached for a TTL (maybe a day)
  - Great for static content that must be available everywhere

- S3 Cross Region Replication:
  - Must be setup for each region you want replication to happen
  - Files are updated in near real-time
  - Great for dynamic content that needs to be available at low-latency in few regions

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudFront Hands On

- We'll create an S3 bucket
- We'll create a CloudFront distribution
- We'll create an Origin Access Identity
- We'll limit the S3 bucket to be accessed only using this identity

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront Signed URL / Signed Cookies

- You want to distribute paid shared content to premium users over the world
- We can use CloudFront Signed URL / Cookie. We attach a policy with:
  - Includes URL expiration
  - Includes IP ranges to access the data from
  - Trusted signers (which AWS accounts can create signed URLs)
- How long should the URL be valid for?
  - Shared content (movie, music): make it short (a few minutes)
  - Private content (private to the user): you can make it last for years

- Signed URL = access to individual files (one signed URL per file)
- Signed Cookies = access to multiple files (one signed cookie for many files)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront Signed URL Diagram

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFront Signed URL vs S3 Pre-Signed URL

- CloudFront Signed URL:
  - Allow access to a path, no matter the origin
  - Account wide key-pair, only the root can manage it
  - Can filter by IP, path, date, expiration
  - Can leverage caching features
- S3 Pre-Signed URL:
  - Issue a request as the person who pre-signed the URL
  - Uses the IAM key of the signing IAM principal
  - Limited lifetime

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Global users for our application

- You have deployed an application and have global users who want to access it directly.
- They go over the public internet, which can add a lot of latency due to many hops
- We wish to go as fast as possible through AWS network to minimize latency

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Unicast IP vs Anycast IP

- **Unicast IP:** one server holds one IP address

- **Anycast IP:** all servers hold the same IP address and the client is routed to the nearest one

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Global Accelerator

- Leverage the AWS internal network to route to your application
- 2 **Anycast IP** are created for your application
- The Anycast IP send traffic directly to Edge Locations
- The Edge locations send the traffic to your application

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Global Accelerator

- Works with **Elastic IP, EC2 instances, ALB, NLB, public or private**
- Consistent Performance
  - Intelligent routing to lowest latency and fast regional failover
  - No issue with client cache (because the IP doesn't change)
  - Internal AWS network
- Health Checks
  - Global Accelerator performs a health check of your applications
  - Helps make your application global (failover less than 1 minute for unhealthy)
  - Great for disaster recovery (thanks to the health checks)
- Security
  - only 2 external IP need to be whitelisted
  - DDoS protection thanks to AWS Shield

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Global Accelerator vs CloudFront

- They both use the AWS global network and its edge locations around the world
- Both services integrate with AWS Shield for DDoS protection.

- CloudFront
  - Improves performance for both cacheable content (such as images and videos)
  - Dynamic content (such as API acceleration and dynamic site delivery)
  - Content is served at the edge

- Global Accelerator
  - Improves performance for a wide range of applications over TCP or UDP
  - Proxying packets at the edge to applications running in one or more AWS Regions.
  - Good fit for non-HTTP use cases, such as gaming (UDP), IoT (MQTT), or Voice over IP
  - Good for HTTP use cases that require static IP addresses
  - Good for HTTP use cases that required deterministic, fast regional failover

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Snowball

- Physical data transport solution that helps moving TBs or PBs of data in or out of AWS
- Alternative to moving data over the network (and paying network fees)
- Secure, tamper resistant, uses KMS 256 bit encryption
- Tracking using SNS and text messages. E-ink shipping label
- Pay per data transfer job
- Use cases: large data cloud migrations, DC decommission, disaster recovery
- If it takes more than a week to transfer over the network, use Snowball devices!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Snowball Process

1. Request snowball devices from the AWS console for delivery
2. Install the snowball client on your servers
3. Connect the snowball to your servers and copy files using the client
4. Ship back the device when you're done (goes to the right AWS facility)
5. Data will be loaded into an S3 bucket
6. Snowball is completely wiped
7. Tracking is done using SNS, text messages and the AWS console

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Diagrams

- Direct upload to S3:

- With snowball

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Snowball Edge

- Snowball Edges add computational capability to the device
- 100 TB capacity with either:
  - Storage optimized – 24 vCPU
  - Compute optimized – 52 vCPU & optional GPU
- Supports a custom EC2 AMI so you can perform processing on the go
- Supports custom Lambda functions

- Very useful to pre-process the data while moving
- Use case: data migration, image collation, IoT capture, machine learning

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Snowmobile

- Transfer exabytes of data (1 EB = 1,000 PB = 1,000,000 TBs)
- Each Snowmobile has 100 PB of capacity (use multiple in parallel)
- Better than Snowball if you transfer more than 10 PB

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Solution Architecture: Snowball into Glacier

- Snowball cannot import to Glacier directly
- You have to use Amazon S3 first, and an S3 lifecycle policy

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Hybrid Cloud for Storage

- AWS is pushing for "hybrid cloud"
  - Part of your infrastructure is on the cloud
  - Part of your infrastructure is on-premise
- This can be due to
  - Long cloud migrations
  - Security requirements
  - Compliance requirements
  - IT strategy
- S3 is a proprietary storage technology (unlike EFS / NFS), so how do you expose the S3 data on-premise?
- AWS Storage Gateway!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Storage Cloud Native Options

BLOCK

Amazon EBS

EC2 Instance Store

FILE

Amazon EFS

OBJECT

S3

Glacier

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Storage Gateway

- Bridge between on-premise data and cloud data in S3
- Use cases: disaster recovery, backup & restore, tiered storage

- 3 types of Storage Gateway:
  - File Gateway
  - Volume Gateway
  - Tape Gateway

- Exam Tip: You need to know the differences between all 3!

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

File Gateway

- Configured S3 buckets are accessible using the NFS and SMB protocol
- Supports S3 standard, S3 IA, S3 One Zone IA
- Bucket access using IAM roles for each File Gateway
- Most recently used data is cached in the file gateway
- Can be mounted on many servers

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Volume Gateway

- Block storage using iSCSI protocol backed by S3
- Backed by EBS snapshots which can help restore on-premise volumes!
- **Cached volumes:** low latency access to most recent data
- **Stored volumes:** entire dataset is on premise, scheduled backups to S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Tape Gateway

- Some companies have backup processes using physical tapes (!)
- With Tape Gateway, companies use the same processes but in the cloud
- Virtual Tape Library (VTL) backed by Amazon S3 and Glacier
- Back up data using existing tape-based processes (and iSCSI interface)
- Works with leading backup software vendors

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Storage Gateway Summary

- Exam tip: Read the question well, it will hint at which gateway to use
- On premise data to the cloud => Storage Gateway

- File access / NFS => File Gateway
  (backed by S3)
- Volumes / Block Storage / iSCSI => Volume gateway
  (backed by S3 with EBS snapshots)
- VTL Tape solution / Backup with iSCSI = > Tape Gateway
  (backed by S3 and Glacier)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon FSx for Windows (File Server)

- *EFS is a shared POSIX system for Linux systems.*

- **FSx for Windows** is a fully managed **Windows** file system share drive
- Supports SMB protocol & Windows NTFS
- Microsoft Active Directory integration, ACLs, user quotas
- Built on SSD, scale up to 10s of GB/s, millions of IOPS, 100s PB of data
- Can be accessed from your on-premise infrastructure
- Can be configured to be Multi-AZ (high availability)
- Data is backed-up daily to S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon FSx for Lustre

FSx

- Lustre is a type of parallel distributed file system, for large-scale computing
- The name Lustre is derived from "Linux" and "cluster"

- Machine Learning, **High Performance Computing (HPC)**
- Video Processing, Financial Modeling, Electronic Design Automation
- Scales up to 100s GB/s, millions of IOPS, sub-ms latencies
- **Seamless integration with S3**
  - Can "read S3" as a file system (through FSx)
  - Can write the output of the computations back to S3 (through FSx)
- Can be used from on-premise servers

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Storage Comparison

- **S3:** Object Storage
- **Glacier:** Object Archival
- **EFS:** Network File System for Linux instances, POSIX filesystem
- **FSx for Windows:** Network File System for Windows servers
- **FSx for Lustre:** High Performance Computing Linux file system
- **EBS volumes:** Network storage for one EC2 instance at a time
- **Instance Storage:** Physical storage for your EC2 instance (high IOPS)
- **Storage Gateway:** File Gateway, Volume Gateway (cache & stored), Tape Gateway
- **Snowball / Snowmobile:** to move large amount of data to the cloud, physically
- **Database:** for specific workloads, usually with indexing and querying

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Integration & Messaging

SQS, SNS & Kinesis

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Introduction

- When we start deploying multiple applications, they will inevitably need to communicate with one another
- There are two patterns of application communication

1) Synchronous communications (application to application)

2) Asynchronous / Event based (application to queue to application)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Introduction

- Synchronous between applications can be problematic if there are sudden spikes of traffic
- What if you need to suddenly encode 1000 videos but usually it's 10?

- In that case, it's better to **decouple** your applications,
  - using SQS: queue model
  - using SNS: pub/sub model
  - using Kinesis: real-time streaming model
- These services can scale independently from our application!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SQS
What's a queue?

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS SQS – Standard Queue

- Oldest offering (over 10 years old)
- Fully managed
- Scales from 1 message per second to 10,000s per second
- Default retention of messages: 4 days, maximum of 14 days
- No limit to how many messages can be in the queue
- Low latency (<10 ms on publish and receive)
- Horizontal scaling in terms of number of consumers
- Can have duplicate messages (at least once delivery, occasionally)
- Can have out of order messages (best effort ordering)
- Limitation of 256KB per message sent

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SQS – Delay Queue

- Delay a message (consumers don't see it immediately) up to 15 minutes
- Default is 0 seconds (message is available right away)
- Can set a default at queue level
- Can override the default using the **DelaySeconds** parameter

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SQS – Producing Messages

- Define Body
- Add message attributes (metadata – optional)
- Provide Delay Delivery (optional)

- Get back
  - Message identifier
  - MD5 hash of the body

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SQS – Consuming Messages

- Consumers…
- Poll SQS for messages (receive up to 10 messages at a time)
- Process the message within the visibility timeout
- Delete the message using the message ID & receipt handle

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SQS – Visibility timeout

- When a consumer polls a message from a queue, the message is "invisible" to other consumers for a defined period… the **Visibility Timeout**:
  - Set between 0 seconds and 12 hours (default 30 seconds)
  - If too high (15 minutes) and consumer fails to process the message, you must wait a long time before processing the message again
  - If too low (30 seconds) and consumer needs time to process the message (2 minutes), another consumer will receive the message and the message will be processed more than once
- **ChangeMessageVisibility** API to change the visibility while processing a message
- **DeleteMessage** API to tell SQS the message was successfully processed

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SQS – Dead Letter Queue

- If a consumer fails to process a message within the Visibility Timeout…
  the message goes back to the queue!
- We can set a threshold of how many times a message can go back to the queue – it's called a "redrive policy"
- After the threshold is exceeded, the message goes into a dead letter queue (DLQ)
- We have to create a DLQ first and then designate it dead letter queue
- Make sure to process the messages in the DLQ before they expire!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SQS - Long Polling

- When a consumer requests message from the queue, it can optionally "wait" for messages to arrive if there are none in the queue
- This is called Long Polling
- **LongPolling decreases the number of API calls made to SQS while increasing the efficiency and latency of your application.**
- The wait time can be between 1 sec to 20 sec (20 sec preferable)
- Long Polling is preferable to Short Polling
- Long polling can be enabled at the queue level or at the API level using **WaitTimeSeconds**

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SQS Message consumption flow diagram

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SQS – FIFO Queue

- Newer offering (First In - First out) – not available in all regions!
- Name of the queue must end in .fifo
- Lower throughput (up to 3,000 per second with batching, 300/s without)
- Messages are processed in order by the consumer
- Messages are sent exactly once
- No per message delay (only per queue delay)
- Ability to do content based de-duplication
- 5-minute interval de-duplication using "Duplication ID"
- Message Groups:
  - Possibility to group messages for FIFO ordering using "Message GroupID"
  - Only one worker can be assigned per message group so that messages are processed in order
  - Message group is just an extra tag on the message!

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SQS with Auto Scaling Group (ASG)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SQS to decouple between application tiers

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SNS

- What if you want to send one message to many receivers?

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SNS

- The "event producer" only sends message to one SNS topic
- As many "event receivers" (subscriptions) as we want to listen to the SNS topic notifications
- Each subscriber to the topic will get all the messages (note: new feature to filter messages)
- Up to 10,000,000 subscriptions per topic
- 100,000 topics limit
- Subscribers can be:
  - SQS
  - HTTP / HTTPS (with delivery retries – how many times)
  - Lambda
  - Emails
  - SMS messages
  - Mobile Notifications

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SNS integrates with a lot of Amazon Products

- Some services can send data directly to SNS for notifications
- CloudWatch (for alarms)
- Auto Scaling Groups notifications
- Amazon S3 (on bucket events)
- CloudFormation (upon state changes => failed to build, etc)
- Etc…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SNS – How to publish

- Topic Publish (within your AWS Server – using the SDK)
  - Create a topic
  - Create a subscription (or many)
  - Publish to the topic

- Direct Publish (for mobile apps SDK)
  - Create a platform application
  - Create a platform endpoint
  - Publish to the platform endpoint
  - Works with Google GCM, Apple APNS, Amazon ADM…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SNS + SQS: Fan Out

- Push once in SNS, receive in many SQS
- Fully decoupled
- No data loss
- Ability to add receivers of data later
- SQS allows for delayed processing
- SQS allows for retries of work
- May have many workers on one queue and one worker on the other queue

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Kinesis Overview

- **Kinesis** is a managed alternative to Apache Kafka
- Great for application logs, metrics, IoT, clickstreams
- Great for "real-time" big data
- Great for streaming processing frameworks (Spark, NiFi, etc…)
- Data is automatically replicated to 3 AZ

- **Kinesis Streams:** low latency streaming ingest at scale
- **Kinesis Analytics:** perform real-time analytics on streams using SQL
- **Kinesis Firehose:** load streams into S3, Redshift, ElasticSearch…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Kinesis Streams Overview

- Streams are divided in ordered Shards / Partitions

- Data retention is 1 day by default, can go up to 7 days
- Ability to reprocess / replay data
- Multiple applications can consume the same stream
- Real-time processing with scale of throughput
- Once data is inserted in Kinesis, it can't be deleted (immutability)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis Streams Shards

- One stream is made of many different shards
- 1MB/s or 1000 messages/s at write PER SHARD
- 2MB/s at read PER SHARD
- Billing is per shard provisioned, can have as many shards as you want
- Batching available or per message calls.
- The number of shards can evolve over time (reshard / merge)
- **Records are ordered per shard**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Kinesis API – Put records

- PutRecord API + Partition key that gets hashed
- The same key goes to the same partition (helps with ordering for a specific key)
- Messages sent get a "sequence number"
- Choose a partition key that is highly distributed (helps prevent "hot partition")
  - user_id if many users
  - Not country_id if 90% of the users are in one country
- Use Batching with PutRecords to reduce costs and increase throughput
- ProvisionedThroughputExceeded if we go over the limits
- Can use CLI, AWS SDK, or producer libraries from various frameworks

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Kinesis API – Exceptions

- ProvisionedThroughputExceeded Exceptions
  - Happens when sending more data (exceeding MB/s or TPS for any shard)
  - Make sure you don't have a hot shard (such as your partition key is bad and too much data goes to that partition)

- Solution:
  - Retries with backoff
  - Increase shards (scaling)
  - Ensure your partition key is a good one

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Kinesis API – Consumers

- Can use a normal consumer (CLI, SDK, etc…)
- Can use Kinesis Client Library (in Java, Node, Python, Ruby, .Net)
  - KCL uses DynamoDB to checkpoint offsets
  - KCL uses DynamoDB to track other workers and share the work amongst shards

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis Security

- Control access / authorization using IAM policies
- Encryption in flight using HTTPS endpoints
- Encryption at rest using KMS
- Possibility to encrypt / decrypt data client side (harder)
- VPC Endpoints available for Kinesis to access within VPC

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Kinesis Data Firehose

- Fully Managed Service, no administration, automatic scaling, serverless
- Load data into Redshift / Amazon S3 / ElasticSearch / Splunk
- Near Real Time
  - 60 seconds latency minimum for non full batches
  - Or minimum 32 MB of data at a time
- Supports many data formats, conversions, transformations, compression

- Pay for the amount of data going through Firehose

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis Data Firehose Diagram

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Kinesis Data Streams vs Firehose

- Streams
  - Going to write custom code (producer / consumer)
  - Real time (~200 ms)
  - Must manage scaling (shard splitting / merging)
  - Data Storage for 1 to 7 days, replay capability, multi consumers

- Firehose
  - Fully managed, send to S3, Splunk, Redshift, ElasticSearch
  - Serverless data transformations with Lambda
  - **Near** real time (lowest buffer time is 1 minute)
  - Automated Scaling
  - No data storage

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis Analytics, Conceptually…

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Kinesis Data Analytics

- Perform real-time analytics on Kinesis Streams using SQL
- Kinesis Data Analytics:
  - Auto Scaling
  - Managed: no servers to provision
  - Continuous: real time
- Pay for actual consumption rate
- Can create streams out of the real-time queries

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Ordering data into Kinesis

- Imagine you have 100 trucks (truck_1, truck_2, … truck_100) on the road sending their GPS positions regularly into AWS.
- You want to consume the data in order for each truck, so that you can track their movement accurately.
- How should you send that data into Kinesis?

- Answer: send using a "Partition Key" value of the "truck_id"
- The same key will always go to the same shard

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Ordering data into SQS

- For SQS standard, there is no ordering.
- For SQS FIFO, if you don't use a Group ID, messages are consumed in the order they are sent, **with only one consumer**

- You want to scale the number of consumers, but you want messages to be "grouped" when they are related to each other
- Then you use a Group ID (similar to Partition Key in Kinesis)

https://aws.amazon.com/blogs/compute/solving-complex-ordering-challenges-with-amazon-sqs-fifo-queues/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Kinesis vs SQS ordering

- Let's assume 100 trucks, 5 kinesis shards, 1 SQS FIFO
- Kinesis Data Streams:
  - On average you'll have 20 trucks per shard
  - Trucks will have their data ordered within each shard
  - The maximum amount of consumers in parallel we can have is 5
  - Can receive up to 5 MB/s of data
- SQS FIFO
  - You only have one SQS FIFO queue
  - You will have 100 Group ID
  - You can have up to 100 Consumers (due to the 100 Group ID)
  - You have up to 300 messages per second (or 3000 if using batching)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SQS vs SNS vs Kinesis

SQS:

- Consumer "pull data"
- Data is deleted after being consumed
- Can have as many workers (consumers) as we want
- No need to provision throughput
- No ordering guarantee (except FIFO queues)
- Individual message delay capability

SNS:

- Push data to many subscribers
- Up to 10,000,000 subscribers
- Data is not persisted (lost if not delivered)
- Pub/Sub
- Up to 100,000 topics
- No need to provision throughput
- Integrates with SQS for fan-out architecture pattern

Kinesis:

- Consumers "pull data"
- As many consumers as we want
- Possibility to replay data
- Meant for real-time big data, analytics and ETL
- Ordering at the shard level
- Data expires after X days
- Must provision throughput

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Amazon MQ

- SQS, SNS are "cloud-native" services, and they're using proprietary protocols from AWS.
- Traditional applications running from on-premise may use open protocols such as: MQTT, AMQP, STOMP, Openwire, WSS
- **When migrating to the cloud**, instead of re-engineering the application to use SQS and SNS, we can use Amazon MQ
- **Amazon MQ = managed Apache ActiveMQ**

- Amazon MQ doesn't "scale" as much as SQS / SNS
- Amazon MQ runs on a dedicated machine, can run in HA with failover
- Amazon MQ has both queue feature (~SQS) and topic features (~SNS)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Serverless Overview

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What's serverless?

- Serverless is a new paradigm in which the developers don't have to manage servers anymore…
- They just deploy code
- They just deploy… functions !
- Initially... Serverless == FaaS (Function as a Service)
- Serverless was pioneered by AWS Lambda but now also includes anything that's managed: "databases, messaging, storage, etc."
- **Serverless does not mean there are no servers…**
  it means you just don't manage / provision / see them

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Serverless in AWS

- AWS Lambda & Step Functions
- DynamoDB
- AWS Cognito
- AWS API Gateway
- Amazon S3
- AWS SNS & SQS
- AWS Kinesis
- Aurora Serverless

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Why AWS Lambda

Amazon EC2

- Virtual Servers in the Cloud
- Limited by RAM and CPU
- Continuously running
- Scaling means intervention to add / remove servers

Amazon Lambda

- Virtual **functions** – no servers to manage!
- Limited by time - **short executions**
- Run **on-demand**
- **Scaling is automated**!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Benefits of AWS Lambda

- Easy Pricing:
  - Pay per request and compute time
  - Free tier of 1,000,000 AWS Lambda requests and 400,000 GBs of compute time

- Integrated with the whole AWS Stack
- Integrated with many programming languages
- Easy monitoring through AWS CloudWatch
- Easy to get more resources per functions (up to 3GB of RAM!)
- Increasing RAM will also improve CPU and network!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Lambda language support

- Node.js (JavaScript)
- Python
- Java (Java 8 compatible)
- C# (.NET Core)
- Golang
- C# / Powershell

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Lambda Integrations
Main ones

API Gateway

Kinesis

DynamoDB

AWS S3 –
Simple Storage Service

AWS IoT
Internet of Things

CloudWatch Events

CloudWatch Logs

AWS SNS

AWS Cognito

Amazon
SQS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example: Serverless Thumbnail creation

New image in S3

trigger

AWS Lambda Function Creates a Thumbnail

push

New thumbnail in S3

push

Metadata in DynamoDB

Image name
Image size
Creation date
etc…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example: Serverless CRON Job

CloudWatch Events

Trigger
Every 1 hour

AWS Lambda Function
Perform a task

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Lambda Pricing (as of June 2018, us-east-1 region)

- You can find overall pricing information here: https://aws.amazon.com/lambda/pricing/
- Pay per **calls**:
  - First 1,000,000 requests are free
  - $0.20 per 1 million requests thereafter ($0.0000002 per request)
- Pay per **duration**: (in increment of 100ms)
  - 400,000 GB-seconds of compute time per month if FREE
  - == 400,000 seconds if function is 1GB RAM
  - == 3,200,000 seconds if function is 128 MB RAM
  - After that $1.00 for 600,000 GB-seconds
- It is usually very cheap to run AWS Lambda so it's very popular

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Lambda Configuration

- Timeout: default 3 seconds, max of 15 minutes (900 seconds)
- Environment variables
- Allocated memory (128M to 3G)
- Ability to deploy within a VPC + assign security groups
- IAM execution role must be attached to the Lambda function

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Lambda Limits to Know

- Execution:
  - Memory allocation: 128 MB – 3008 MB (64 MB increments)
  - Maximum execution time: 300 seconds (5 minutes), now 15 minutes but 5 for exam
  - Disk capacity in the "function container" (in /tmp): 512 MB
  - Concurrency limits: 1000
- Deployment:
  - Lambda function deployment size (compressed .zip): 50 MB
  - Size of uncompressed deployment (code + dependencies): 250 MB
  - Can use the /tmp directory to load other files at startup
  - Size of environment variables: 4 KB

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Lambda@Edge

- You have deployed a CDN using CloudFront
- What if you wanted to run a global AWS Lambda alongside?
- Or how to implement request filtering before reaching your application?

- For this, you can use **Lambda@Edge**:
  deploy Lambda functions alongside your CloudFront CDN
  - Build more responsive applications
  - You don't manage servers, Lambda is deployed globally
  - Customize the CDN content
  - Pay only for what you use

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Lambda@Edge

- You can use Lambda to change CloudFront requests and responses:
  - After CloudFront receives a request from a viewer (viewer request)
  - Before CloudFront forwards the request to the origin (origin request)
  - After CloudFront receives the response from the origin (origin response)
  - Before CloudFront forwards the response to the viewer (viewer response)

- You can also generate responses to viewers without ever sending the request to the origin

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Lambda@Edge: Global application

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Lambda@Edge: Use Cases

- Website Security and Privacy
- Dynamic Web Application at the Edge
- Search Engine Optimization (SEO)
- Intelligently Route Across Origins and Data Centers
- Bot Mitigation at the Edge
- Real-time Image Transformation
- A/B Testing
- User Authentication and Authorization
- User Prioritization
- User Tracking and Analytics

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB

- Fully Managed, Highly available with replication across 3 AZ
- NoSQL database - not a relational database
- Scales to massive workloads, distributed database
- Millions of requests per seconds, trillions of row, 100s of TB of storage
- Fast and consistent in performance (low latency on retrieval)
- Integrated with IAM for security, authorization and administration
- Enables event driven programming with DynamoDB Streams
- Low cost and auto scaling capabilities

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB - Basics

- DynamoDB is made of **tables**
- Each table has a **primary key** (must be decided at creation time)
- Each table can have an infinite number of items (= rows)
- Each item has **attributes** (can be added over time – can be null)
- Maximum size of a item is 400KB
- Data types supported are:
  - Scalar Types: String, Number, Binary, Boolean, Null
  - Document Types: List, Map
  - Set Types: String Set, Number Set, Binary Set

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB – Provisioned Throughput

- Table must have provisioned read and write capacity units
- **Read Capacity Units (RCU):** throughput for reads ($0.00013 per RCU)
  - 1 RCU = 1 strongly consistent read of 4 KB per second
  - 1 RCU = 2 eventually consistent read of 4 KB per second
- **Write Capacity Units (WCU):** throughput for writes ($0.00065 per WCU)
  - 1 WCU = 1 write of 1 KB per second
- Option to setup auto-scaling of throughput to meet demand
- Throughput can be exceeded temporarily using "burst credit"
- If burst credit are empty, you'll get a "ProvisionedThroughputException".
- It's then advised to do an exponential back-off retry

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB - DAX

- DAX = DynamoDB Accelerator
- Seamless cache for DynamoDB, no application re-write
- Writes go through DAX to DynamoDB
- Micro second latency for cached reads & queries
- Solves the Hot Key problem (too many reads)
- 5 minutes TTL for cache by default
- Up to 10 nodes in the cluster
- Multi AZ (3 nodes minimum recommended for production)
- Secure (Encryption at rest with KMS, VPC, IAM, CloudTrail…)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB Streams

- Changes in DynamoDB (Create, Update, Delete) can end up in a DynamoDB Stream
- This stream can be read by AWS Lambda, and we can then do:
  - React to changes in real time (welcome email to new users)
  - Analytics
  - Create derivative tables / views
  - Insert into ElasticSearch
- Could implement cross region replication using Streams
- Stream has 24 hours of data retention

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB - New Features

- Transactions (new from Nov 2018)
  - All or nothing type of operations
  - Coordinated Insert, Update & Delete across multiple tables
  - Include up to 10 unique items or up to 4 MB of data

- On Demand (new from Nov 2018)
  - No capacity planning needed (WCU / RCU) – scales automatically
  - 2.5x more expensive than provisioned capacity (use with care)
  - Helpful when spikes are un-predictable or the application is very low throughput

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB – Security & Other Features

- Security:
  - VPC Endpoints available to access DynamoDB without internet
  - Access fully controlled by IAM
  - Encryption at rest using KMS
  - Encryption in transit using SSL / TLS
- Backup and Restore feature available
  - Point in time restore like RDS
  - No performance impact
- Global Tables
  - Multi region, fully replicated, high performance
- Amazon DMS can be used to migrate to DynamoDB (from Mongo, Oracle, MySQL, S3, etc…)
- You can launch a local DynamoDB on your computer for development purposes

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB – Other features

- **Global Tables:** (cross region replication)
  - Active Active replication, many regions
  - Must enable DynamoDB Streams
  - Useful for low latency, DR purposes

- **Capacity planning:**
  - Planned capacity: provision WCU & RCU, can enable auto scaling
  - On-demand capacity: get unlimited WCU & RCU, no throttle, more expensive

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Building a Serverless API

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS API Gateway

- AWS Lambda + API Gateway: No infrastructure to manage
- Handle API versioning (v1, v2…)
- Handle different environments (dev, test, prod…)
- Handle security (Authentication and Authorization)
- Create API keys, handle request throttling
- Swagger / Open API import to quickly define APIs
- Transform and validate requests and responses
- Generate SDK and API specifications
- Cache API responses

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

API Gateway Integrations

- Outside of VPC:
  - AWS Lambda (most popular / powerful)
  - Endpoints on EC2
  - Load Balancers
  - Any AWS Service
  - External and publicly accessible HTTP endpoints

- Inside of VPC:
  - AWS Lambda in your VPC
  - EC2 endpoints in your VPC

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

API Gateway – Security
IAM Permissions

- Create an IAM policy authorization and attach to User / Role
- API Gateway verifies IAM permissions passed by the calling application
- Good to provide access within your own infrastructure
- Leverages "Sig v4" capability where IAM credential are in headers

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

API Gateway – Security
Lambda Authorizer (formerly Custom Authorizers)

- Uses AWS Lambda to validate the token in header being passed
- Option to cache result of authentication
- Helps to use OAuth / SAML / 3rd party type of authentication
- Lambda must return an IAM policy for the user

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

API Gateway – Security
Cognito User Pools

- Cognito fully manages user lifecycle
- API gateway verifies identity automatically from AWS Cognito
- No custom implementation required
- Cognito only helps with authentication, not authorization

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

API Gateway – Security – Summary

- IAM:
  - Great for users / roles already within your AWS account
  - Handle authentication + authorization
  - Leverages Sig v4
- Custom Authorizer:
  - Great for 3rd party tokens
  - Very flexible in terms of what IAM policy is returned
  - Handle Authentication + Authorization
  - Pay per Lambda invocation
- Cognito User Pool:
  - You manage your own user pool (can be backed by Facebook, Google login etc…)
  - No need to write any custom code
  - Must implement authorization in the backend

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Cognito

- We want to give our users an identity so that they can interact with our application.
- Cognito User Pools:
  - Sign in functionality for app users
  - Integrate with API Gateway
- Cognito Identity Pools (Federated Identity):
  - Provide AWS credentials to users so they can access AWS resources directly
  - Integrate with Cognito User Pools as an identity provider
- Cognito Sync:
  - Synchronize data from device to Cognito.
  - May be deprecated and replaced by AppSync

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Cognito User Pools (CUP)

- Create a serverless database of user for your mobile apps
- Simple login: Username (or email) / password combination
- Possibility to verify emails / phone numbers and **add MFA**
- Can enable Federated Identities (Facebook, Google, SAML…)
- Sends back a JSON Web Tokens (JWT)
- **Can be integrated with API Gateway for authentication**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Cognito – Federated Identity Pools

- Goal:
  - Provide direct access to AWS Resources from the Client Side
- How:
  - Log in to federated identity provider – or remain anonymous
  - Get temporary AWS credentials back from the Federated Identity Pool
  - These credentials come with a pre-defined IAM policy stating their permissions
- Example:
  - provide (temporary) access to write to S3 bucket using Facebook Login

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Cognito Sync

- **Deprecated – use AWS AppSync now**

- Store preferences, configuration, state of app
- Cross device synchronization (any platform – iOS, Android, etc…)
- Offline capability (synchronization when back online)
- **Requires Federated Identity Pool in Cognito (not User Pool)**
- Store data in datasets (up to 1MB)
- Up to 20 datasets to synchronise

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS SAM - Serverless Application Model

- SAM = Serverless Application Model
- Framework for developing and deploying serverless applications
- All the configuration is YAML code
  - Lambda Functions
  - DynamoDB tables
  - API Gateway
  - Cognito User Pools
- SAM can help you to run Lambda, API Gateway, DynamoDB locally
- SAM can use CodeDeploy to deploy Lambda functions

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Serverless Architectures

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Mobile application: MyTodoList

- We want to create a mobile application with the following requirements

- Expose as REST API with HTTPS
- Serverless architecture
- Users should be able to directly interact with their own folder in S3
- Users should authenticate through a managed serverless service
- The users can write and read to-dos, but they mostly read them
- The database should scale, and have some high read throughput

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Mobile app: REST API layer

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Mobile app: giving users access to S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Mobile app: high read throughput, static data

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Mobile app: caching at the API Gateway

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

In this lecture

- Serverless REST API: HTTPS, API Gateway, Lambda, DynamoDB
- Using Cognito to generate temporary credentials with STS to access S3 bucket with restricted policy. App users can directly access AWS resources this way. Pattern can be applied to DynamoDB, Lambda…
- Caching the reads on DynamoDB using DAX
- Caching the REST requests at the API Gateway level
- Security for authentication and authorization with Cognito, STS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Serverless hosted website: MyBlog.com

- This website should scale globally
- Blogs are rarely written, but often read
- Some of the website is purely static files, the rest is a dynamic REST API
- Caching must be implement where possible
- Any new users that subscribes should receive a welcome email
- Any photo uploaded to the blog should have a thumbnail generated

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Serving static content, globally

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Serving static content, globally, securely

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Adding a public serverless REST API

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Leveraging DynamoDB Global Tables

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

User Welcome email flow

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Thumbnail Generation flow

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Hosted Website Summary

- We've seen static content being distributed using CloudFront with S3
- The REST API was serverless, didn't need Cognito because public
- We leveraged a Global DynamoDB table to serve the data globally
- (we could have used Aurora Global Tables)
- We enabled DynamoDB streams to trigger a Lambda function
- The lambda function had an IAM role which could use SES
- SES (Simple Email Service) was used to send emails in a serverless way
- S3 can trigger SQS / SNS / Lambda to notify of events

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Micro Services architecture

- We want to switch to a micro service architecture
- Many services interact with each other directly using a REST API
- Each architecture for each micro service may vary in form and shape

- We want a micro-service architecture so we can have a leaner development lifecycle for each service

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Micro Services Environment

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Discussions on Micro Services

- You are free to design each micro-service the way you want
- Synchronous patterns: API Gateway, Load Balancers
- Asynchronous patterns: SQS, Kinesis, SNS, Lambda triggers (S3)
- Challenges with micro-services:
  - repeated overhead for creating each new microservice,
  - issues with optimizing server density/utilization
  - complexity of running multiple versions of multiple microservices simultaneously
  - proliferation of client-side code requirements to integrate with many separate services.
- Some of the challenges are solved by Serverless patterns:
  - API Gateway, Lambda scale automatically and you pay per usage
  - You can easily clone API, reproduce environments
  - Generated client SDK through Swagger integration for the API Gateway

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Distributing paid content

- We sell videos online and users have to paid to buy videos
- Each videos can be bought by many different customers
- We only want to distribute videos to users who are premium users
- We have a database of premium users
- Links we send to premium users should be short lived
- Our application is global
- We want to be fully serverless

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Start simple, premium user service

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Add authentication

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Add Videos Storage Service

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Distribute Globally and Secure

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Distribute Content only to premium users

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Premium User Video service

- We have implemented a fully serverless solution:
  - Cognito for authentication
  - DynamoDB for storing users that are premium
  - 2 serverless applications
    - Premium User registration
    - CloudFront Signed URL generator
  - Content is stored in S3 (serverless and scalable)
  - Integrated with CloudFront with OAI for security (users can't bypass)
  - CloudFront can only be used using Signed URLs to prevent unauthorized users
  - What about S3 Signed URL? They're not efficient for global access

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Software updates offloading

- We have an application running on EC2, that distributes software updates once in a while
- When a new software update is out, we get a lot of request and the content is distributed in mass over the network. It's very costly
- We don't want to change our application, but want to optimize our cost and CPU, how can we do it?

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Our application current state

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Easy way to fix things!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why CloudFront?

- No changes to architecture
- Will cache software update files at the edge
- Software update files are not dynamic, they're static (never changing)
- Our EC2 instances aren't serverless
- But CloudFront is, and will scale for us
- Our ASG will not scale as much, and we'll save tremendously in EC2
- We'll also save in availability, network bandwidth cost, etc
- Easy way to make an existing application more scalable and cheaper!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Big Data Ingestion Pipeline

- We want the ingestion pipeline to be fully serverless
- We want to collect data in real time
- We want to transform the data
- We want to query the transformed data using SQL
- The reports created using the queries should be in S3
- We want to load that data into a warehouse and create dashboards

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Big Data Ingestion Pipeline

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Big Data Ingestion Pipeline discussion

- IoT Core allows you to harvest data from IoT devices
- Kinesis is great for real-time data collection
- Firehose helps with data delivery to S3 in near real-time (1 minute)
- Lambda can help Firehose with data transformations
- Amazon S3 can trigger notifications to SQS
- Lambda can subscribe to SQS (we could have connecter S3 to Lambda)
- Athena is a serverless SQL service and results are stored in S3
- The reporting bucket contains analyzed data and can be used by reporting tool such as AWS QuickSight, Redshift, etc…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Databases

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Choosing the Right Database

- We have a lot of managed databases on AWS to choose from
- Questions to choose the right database based on your architecture:
  - Read-heavy, write-heavy, or balanced workload? Throughput needs? Will it change, does it need to scale or fluctuate during the day?
  - How much data to store and for how long? Will it grow? Average object size? How are they accessed?
  - Data durability? Source of truth for the data ?
  - Latency requirements? Concurrent users?
  - Data model? How will you query the data? Joins? Structured? Semi-Structured?
  - Strong schema? More flexibility? Reporting? Search? RDBMS / NoSQL?
  - License costs? Switch to Cloud Native DB such as Aurora?

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Database Types

- **RDBMS (= SQL / OLTP):** RDS, Aurora – great for joins
- **NoSQL database:** DynamoDB (~JSON), ElastiCache (key / value pairs), Neptune (graphs) – no joins, no SQL
- **Object Store:** S3 (for big objects) / Glacier (for backups / archives)
- **Data Warehouse** (= SQL Analytics / BI): Redshift (OLAP), Athena
- **Search:** ElasticSearch (JSON) – free text, unstructured searches
- **Graphs:** Neptune – displays relationships between data

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

RDS Overview

- Managed PostgreSQL / MySQL / Oracle / SQL Server
- Must provision an EC2 instance & EBS Volume type and size
- Support for Read Replicas and Multi AZ
- Security through IAM, Security Groups, KMS , SSL in transit
- Backup / Snapshot / Point in time restore feature
- Managed and Scheduled maintenance
- Monitoring through CloudWatch

- **Use case:** Store relational datasets (RDBMS / OLTP), perform SQL queries, transactional inserts / update / delete is available

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RDS for Solutions Architect

- **Operations:** small downtime when failover happens, when maintenance happens, scaling in read replicas / ec2 instance / restore EBS implies manual intervention, application changes
- **Security:** AWS responsible for OS security, we are responsible for setting up KMS, security groups, IAM policies, authorizing users in DB, using SSL
- **Reliability:** Multi AZ feature, failover in case of failures
- **Performance:** depends on EC2 instance type, EBS volume type, ability to add Read Replicas. Doesn't auto-scale
- **Cost:** Pay per hour based on provisioned EC2 and EBS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Aurora Overview

- Compatible API for PostgreSQL / MySQL
- Data is held in 6 replicas, across 3 AZ
- Auto healing capability
- Multi AZ, Auto Scaling Read Replicas
- Read Replicas can be Global
- Aurora database can be Global for DR or latency purposes
- Auto scaling of storage from 10GB to 64 TB
- Define EC2 instance type for aurora instances
- Same security / monitoring / maintenance features as RDS
- "Aurora Serverless" option

- **Use case:** same as RDS, but with less maintenance / more flexibility / more performance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Aurora for Solutions Architect

- **Operations:** less operations, auto scaling storage
- **Security:** AWS responsible for OS security, we are responsible for setting up KMS, security groups, IAM policies, authorizing users in DB, using SSL
- **Reliability:** Multi AZ, highly available, possibly more than RDS, Aurora Serverless option.
- **Performance:** 5x performance (according to AWS) due to architectural optimizations. Up to 15 Read Replicas (only 5 for RDS)
- **Cost:** Pay per hour based on EC2 and storage usage. Possibly lower costs compared to Enterprise grade databases such as Oracle

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElastiCache Overview

- Managed Redis / Memcached (similar offering as RDS, but for caches)
- In-memory data store, sub-millisecond latency
- Must provision an EC2 instance type
- Support for Clustering (Redis) and Multi AZ, Read Replicas (sharding)
- Security through IAM, Security Groups, KMS, Redis Auth
- Backup / Snapshot / Point in time restore feature
- Managed and Scheduled maintenance
- Monitoring through CloudWatch

- **Use Case:** Key/Value store, Frequent reads, less writes, cache results for DB queries, store session data for websites, cannot use SQL.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElastiCache for Solutions Architect

- **Operations:** same as RDS
- **Security:** AWS responsible for OS security, we are responsible for setting up KMS, security groups, IAM policies, users (Redis Auth), using SSL
- **Reliability:** Clustering, Multi AZ
- **Performance:** Sub-millisecond performance, in memory, read replicas for sharding, very popular cache option
- **Cost:** Pay per hour based on EC2 and storage usage

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DynamoDB Overview

- AWS proprietary technology, managed NoSQL database
- Serverless, provisioned capacity, auto scaling, on demand capacity (Nov 2018)
- Can replace ElastiCache as a key/value store (storing session data for example)
- Highly Available, Multi AZ by default, Read and Writes are decoupled, DAX for read cache
- Reads can be eventually consistent or strongly consistent
- Security, authentication and authorization is done through IAM
- DynamoDB Streams to integrate with AWS Lambda
- Backup / Restore feature, Global Table feature
- Monitoring through CloudWatch
- Can only query on primary key, sort key, or indexes

- **Use Case:** Serverless applications development (small documents 100s KB), distributed serverless cache, doesn't have SQL query language available, has transactions capability from Nov 2018

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

DynamoDB for Solutions Architect

- **Operations:** no operations needed, auto scaling capability, serverless
- **Security:** full security through IAM policies, KMS encryption, SSL in flight
- **Reliability:** Multi AZ, Backups
- **Performance:** single digit millisecond performance, DAX for caching reads, performance doesn't degrade if your application scales
- **Cost:** Pay per provisioned capacity and storage usage (no need to guess in advance any capacity – can use auto scaling)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

S3 Overview

- S3 is a… key / value store for objects
- Great for big objects, not so great for small objects
- Serverless, scales infinitely, max object size is 5 TB
- Eventually consistency for overwrites and deletes
- Tiers: S3 Standard, S3 IA, S3 One Zone IA, Glacier for backups
- Features: Versioning, Encryption, Cross Region Replication, etc…
- Security: IAM, Bucket Policies, ACL
- Encryption: SSE-S3, SSE-KMS, SSE-C, client side encryption, SSL in transit

- **Use Case:** static files, key value store for big files, website hosting

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

S3 for Solutions Architect

- **Operations:** no operations needed
- **Security:** IAM, Bucket Policies, ACL, Encryption (Server/Client), SSL
- **Reliability:** 99.999999999% durability / 99.99% availability, Multi AZ, CRR
- **Performance:** scales to thousands of read / writes per second, transfer acceleration / multi-part for big files
- **Cost:** pay per storage usage, network cost, requests number

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Athena Overview

- Fully Serverless database with SQL capabilities
- Used to query data in S3
- Pay per query
- Output results back to S3
- Secured through IAM

- **Use Case:** one time SQL queries, serverless queries on S3, log analytics

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Athena for Solutions Architect

- **Operations:** no operations needed, serverless
- **Security:** IAM + S3 security
- **Reliability:** managed service, uses Presto engine, highly available
- **Performance:** queries scale based on data size
- **Cost:** pay per query / per TB of data scanned, serverless

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Redshift Overview

- Redshift is based on PostgreSQL, but **it's not used for OLTP**
- **It's OLAP – online analytical processing (analytics and data warehousing)**
- 10x better performance than other data warehouses, scale to PBs of data
- **Columnar** storage of data (instead of row based)
- Massively Parallel Query Execution (MPP), highly available
- Pay as you go based on the instances provisioned
- Has a SQL interface for performing the queries
- BI tools such as AWS Quicksight or Tableau integrate with it

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Redshift Continued…

- Data is loaded from S3, DynamoDB, DMS, other DBs…
- From 1 node to 128 nodes, up to 160 GB of space per node
- Leader node: for query planning, results aggregation
- Compute node: for performing the queries, send results to leader
- Redshift Spectrum: perform queries directly against S3 (no need to load)
- Backup & Restore, Security VPC / IAM / KMS, Monitoring
- Redshift Enhanced VPC Routing: COPY / UNLOAD goes through VPC

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Redshift – Snapshots & DR

- Snapshots are point-in-time backups of a cluster, stored internally in S3
- Snapshots are incremental (only what has changed is saved)
- You can restore a snapshot into a **new cluster**
- Automated: every 8 hours, every 5 GB, or on a schedule. Set retention
- Manual: snapshot is retained until you delete it

- You can configure Amazon Redshift to automatically copy snapshots (automated or manual) of a cluster to another AWS Region

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Redshift Spectrum

- Query data that is already in S3 without loading it
- **Must have a Redshift cluster available to start the query**
- The query is then submitted to thousands of Redshift Spectrum nodes

https://aws.amazon.com/blogs/big-data/amazon-redshift-spectrum-extends-data-warehousing-out-to-exabytes-no-loading-required/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Redshift for Solutions Architect

- **Operations:** similar to RDS
- **Security:** IAM, VPC, KMS, SSL (similar to RDS)
- **Reliability:** highly available, auto healing features
- **Performance:** 10x performance vs other data warehousing, compression
- **Cost:** pay per node provisioned, 1/10th of the cost vs other warehouses

- Remember: Redshift = Analytics / BI / Data Warehouse

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Neptune

- Fully managed **graph** database
- When do we use Graphs?
  - High relationship data
  - Social Networking: Users friends with Users, replied to comment on post of user and likes other comments.
  - Knowledge graphs (Wikipedia)
- Highly available across 3 AZ, with up to 15 read replicas
- Point-in-time recovery, continuous backup to Amazon S3
- Support for KMS encryption at rest + HTTPS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Neptune for Solutions Architect

- **Operations:** similar to RDS
- **Security:** IAM, VPC, KMS, SSL (similar to RDS) + IAM Authentication
- **Reliability:** Multi-AZ, clustering
- **Performance:** best suited for graphs, clustering to improve performance
- **Cost:** pay per node provisioned (similar to RDS)

- Remember: Neptune = Graphs

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ElasticSearch

- Example: In DynamoDB, you can only find by primary key or indexes.
- With ElasticSearch, you can **search any field**, even partially matches
- It's common to use ElasticSearch as a complement to another database
- ElasticSearch also has some usage for Big Data applications
- You can provision a cluster of instances
- Built-in integrations: Amazon Kinesis Data Firehose, AWS IoT, and Amazon CloudWatch Logs for data ingestion
- Security through Cognito & IAM, KMS encryption, SSL & VPC
- Comes with Kibana (visualization) & Logstash (log ingestion) – ELK stack

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

ElasticSearch for Solutions Architect

- **Operations:** similar to RDS
- **Security:** Cognito, IAM, VPC, KMS, SSL
- **Reliability:** Multi-AZ, clustering
- **Performance:** based on ElasticSearch project (open source), petabyte scale
- **Cost:** pay per node provisioned (similar to RDS)

- Remember: ElasticSearch = Search / Indexing

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Monitoring, Audit and Performance

CloudWatch, CloudTrail & AWS Config

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Metrics

- CloudWatch provides metrics for *every* services in AWS
- **Metric** is a variable to monitor (CPUUtilization, NetworkIn…)
- Metrics belong to **namespaces**
- **Dimension** is an attribute of a metric (instance id, environment, etc…).
- Up to 10 dimensions per metric
- Metrics have **timestamps**
- Can create CloudWatch dashboards of metrics

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch EC2 Detailed monitoring

- EC2 instance metrics have metrics "every 5 minutes"
- With detailed monitoring (for a cost), you get data "every 1 minute"
- Use detailed monitoring if you want to more prompt scale your ASG!

- The AWS Free Tier allows us to have 10 detailed monitoring metrics

- Note: EC2 Memory usage is by default not pushed (must be pushed from inside the instance as a custom metric)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Custom Metrics

- Possibility to define and send your own custom metrics to CloudWatch
- Ability to use dimensions (attributes) to segment metrics
  - Instance.id
  - Environment.name
- Metric resolution (**StorageResolution** API parameter – two possible value):
  - Standard: 1 minute (60 seconds)
  - High Resolution: 1 second – Higher cost
- Use API call **PutMetricData**
- Use exponential back off in case of throttle errors

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudWatch Dashboards

- Great way to setup dashboards for quick access to keys metrics
- Dashboards are global
- Dashboards can include graphs from different regions
- You can change the time zone & time range of the dashboards
- You can setup automatic refresh (10s, 1m, 2m, 5m, 15m)

- Pricing:
  - 3 dashboards (up to 50 metrics) for free
  - $3/dashboard/month afterwards

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Logs

- Applications can send logs to CloudWatch using the SDK
- CloudWatch can collect log from:
  - Elastic Beanstalk: collection of logs from application
  - ECS: collection from containers
  - AWS Lambda: collection from function logs
  - VPC Flow Logs: VPC specific logs
  - API Gateway
  - CloudTrail based on filter
  - CloudWatch log agents: for example on EC2 machines
  - Route53: Log DNS queries
- CloudWatch Logs can go to:
  - Batch exporter to S3 for archival
  - Stream to ElasticSearch cluster for further analytics

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Logs

- Logs storage architecture:
  - Log groups: arbitrary name, usually representing an application
  - Log stream: instances within application / log files / containers
- Can define log expiration policies (never expire, 30 days, etc..)
- Using the AWS CLI we can tail CloudWatch logs
- To send logs to CloudWatch, make sure IAM permissions are correct!
- Security: encryption of logs using KMS at the Group Level

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudWatch Logs Metric Filter & Insights

- CloudWatch Logs can use filter expressions
  - For example, find a specific IP inside of a log
  - Metric filters can be used to trigger alarms

- CloudWatch Logs Insights (new – Nov 2018) can be used to query logs and add queries to CloudWatch Dashboards

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Alarms

- Alarms are used to trigger notifications for any metric
- Alarms can go to Auto Scaling, EC2 Actions, SNS notifications
- Various options (sampling, %, max, min, etc…)
- Alarm States:
  - OK
  - INSUFFICIENT_DATA
  - ALARM
- Period:
  - Length of time in seconds to evaluate the metric
  - High resolution custom metrics: can only choose 10 sec or 30 sec

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudWatch Events

- Source + Rule => Target

- Schedule: Cron jobs
- Event Pattern: Event rules to react to a service doing something
  - Ex: CodePipeline state changes!
- Triggers to Lambda functions, SQS/SNS/Kinesis Messages
- CloudWatch Event creates a small JSON document to give information about the change

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS CloudTrail

- Provides governance, compliance and audit for your AWS Account
- CloudTrail is enabled by default!
- Get an history of events / API calls made within your AWS Account by:
  - Console
  - SDK
  - CLI
  - AWS Services
- Can put logs from CloudTrail into CloudWatch Logs
- If a resource is deleted in AWS, look into CloudTrail first!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Config

- Helps with auditing and recording **compliance** of your AWS resources
- Helps record configurations and changes over time
- Possibility of storing the configuration data into S3 (analyzed by Athena)
- Questions that can be solved by AWS Config:
  - Is there unrestricted SSH access to my security groups?
  - Do my buckets have any public access?
  - How has my ALB configuration changed over time?
- You can receive alerts (SNS notifications) for any changes
- AWS Config is a per-region service
- Can be aggregated across regions and accounts

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Config Resource

- View compliance of a resource over time

- View configuration of a resource over time

- View CloudTrail API calls if enabled

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Config Rules

- Can use AWS managed config rules (over 75)
- Can make custom config rules (must be defined in AWS Lambda)
  - Evaluate if each EBS disk is of type gp2
  - Evaluate if each EC2 instance is t2.micro
- Rules can be evaluated / triggered:
  - For each config change
  - And / or: at regular time intervals
  - Can trigger CloudWatch Events if the rule is non-compliant (and chain with Lambda)
- Rules can have auto remediations:
  - If a resource is not compliant, you can trigger an auto remediation
  - Ex: stop instances with non-approved tags
- AWS Config Rules does not prevent actions from happening (no deny)
- **Pricing:** no free tier, $2 per active rule per region per month

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudWatch vs CloudTrail vs Config

- CloudWatch
  - Performance monitoring (metrics, CPU, network, etc…) & dashboards
  - Events & Alerting
  - Log Aggregation & Analysis
- CloudTrail
  - Record API calls made within your Account by everyone
  - Can define trails for specific resources
  - Global Service
- Config
  - Record configuration changes
  - Evaluate resources against compliance rules
  - Get timeline of changes and compliance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS STS – Security Token Service

- Allows to grant limited and temporary access to AWS resources.
- Token is valid for up to one hour (must be refreshed)
- AssumeRole
  - Within your own account: for enhanced security
  - Cross Account Access: assume role in target account to perform actions there
- AssumeRoleWithSAML
  - return credentials for users logged with SAML
- AssumeRoleWithWebIdentity
  - return creds for users logged with an IdP (Facebook Login, Google Login, OIDC compatible…)
  - AWS recommends against using this, and using **Cognito** instead
- GetSessionToken
  - for MFA, from a user or AWS account root user

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Using STS to Assume a Role

- Define an IAM Role within your account or cross-account
- Define which principals can access this IAM Role
- Use AWS STS (Security Token Service) to retrieve credentials and impersonate the IAM Role you have access to (**AssumeRole API**)
- Temporary credentials can be valid between 15 minutes to 1 hour

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Cross account access with STS

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_common-scenarios_aws-accounts.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Identity Federation in AWS

- Federation lets users outside of AWS to assume temporary role for accessing AWS resources.
- These users assume identity provided access role.

- Federations can have many flavors:
  - SAML 2.0
  - Custom Identity Broker
  - Web Identity Federation with Amazon Cognito
  - Web Identity Federation without Amazon Cognito
  - Single Sign On
  - Non-SAML with AWS Microsoft AD

- Using federation, you don't need to create IAM users (user management is outside of AWS)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SAML 2.0 Federation

- To integrate Active Directory / ADFS with AWS (or any SAML 2.0)
- Provides access to AWS Console or CLI (through temporary creds)
- No need to create an IAM user for each of your employees

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_saml.html

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_enable-console-saml.html

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SAML 2.0 Federation – Active Directory FS

- Same process as with any SAML 2.0 compatible IdP

https://aws.amazon.com/blogs/security/aws-federated-authentication-with-active-directory-federation-services-ad-fs/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SAML 2.0 Federation

- Needs to setup a trust between AWS IAM and SAML (both ways)
- SAML 2.0 enables web-based, cross domain SSO
- Uses the STS API: AssumeRoleWithSAML

- Note federation through SAML is the "old way" of doing things
- **Amazon Single Sign On (SSO)** Federation is the new managed and simpler way
  - Read more here: https://aws.amazon.com/blogs/security/enabling-federation-to-aws-using-windows-active-directory-adfs-and-saml-2-0/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Custom Identity Broker Application

- Use only if identity provider is not compatible with SAML 2.0
- **The identity broker must determine the appropriate IAM policy**
- Uses the STS API: **AssumeRole** or **GetFederationToken**

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_common-scenarios_federated-users.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Web Identity Federation – AssumeRoleWithWebIdentity

- Not recommended by AWS – use Cognito instead (allows for anonymous users, data synchronization, MFA)

https://docs.amazonaws.cn/en_us/amazondynamodb/latest/developerguide/WIF.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Cognito

- Goal:
  - Provide direct access to AWS Resources from the Client Side (mobile, web app)
- Example:
  - provide (temporary) access to write to S3 bucket using Facebook Login
- Problem:
  - We don't want to create IAM users for our app users
- How:
  - Log in to federated identity provider – or remain anonymous
  - Get temporary AWS credentials back from the Federated Identity Pool
  - These credentials come with a pre-defined IAM policy stating their permissions

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

What is Microsoft Active Directory (AD)?

- Found on any Windows Server with AD Domain Services
- Database of **objects**: User Accounts, Computers, Printers, File Shares, Security Groups
- Centralized security management, create account, assign permissions
- Objects are organized in **trees**
- A group of trees is a **forest**

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Directory Services

- AWS Managed Microsoft AD
  - Create your own AD in AWS, manage users locally, supports MFA
  - Establish "trust" connections with your on-premise AD
- AD Connector
  - Directory Gateway (proxy) to redirect to on-premise AD
  - Users are managed on the on-premise AD
- Simple AD
  - AD-compatible managed directory on AWS
  - Cannot be joined with on-premise AD

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Organizations

- Global service
- Allows to manage multiple AWS accounts
- The main account is the master account – you can't change it
- Other accounts are member accounts
- Member accounts can only be part of one organization
- Consolidated Billing across all accounts - single payment method
- Pricing benefits from aggregated usage (volume discount for EC2, S3…)
- API is available to automate AWS account creation

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Multi Account Strategies

- Create accounts per **department**, per **cost center**, per **dev / test / prod**, based on **regulatory restrictions** (using SCP), for **better resource isolation** (ex: VPC), to have **separate per-account service limits**, isolated account for **logging**

- Multi Account vs One Account Multi VPC
- Use tagging standards for billing purposes
- Enable CloudTrail on all accounts, send logs to central S3 account
- Send CloudWatch Logs to central logging account
- Establish Cross Account Roles for Admin purposes

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Organizational Units (OU) - Examples

Business Unit

Environmental Lifecycle

Project-based

https://aws.amazon.com/answers/account-management/aws-multi-account-billing-strategy/

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Organization

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Service Control Policies (SCP)

- Whitelist or blacklist IAM actions
- Applied at the **Root**, **OU** or **Account** level

- SCP is applied to all the **Users and Roles** of the Account, including Root
- The SCP does not affect service-linked roles
  - Service-linked roles enable other AWS services to integrate with AWS Organizations and can't be restricted by SCPs.
- SCP must have an explicit Allow (does not allow anything by default)
- Use cases:
  - Restrict access to certain services (for example: can't use EMR)
  - Enforce PCI compliance by explicitly disabling services

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SCP Hierarchy

- Master Account
  - Can do anything
  - EXCEPT access Athena
- Account A
  - Can do anything
  - EXCEPT access Redshift (explicit Deny from OU)
- Account B
  - Can do anything
  - EXCEPT access Redshift (explicit Deny from Prod OU)
  - EXCEPT access Lambda (explicit Deny from HR OU)
- Account C
  - Can do anything
  - EXCEPT access Redshift (explicit Deny from Prod OU)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

SCP Examples
Blacklist and Whitelist strategies

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowsAllActions",
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        },
        {
            "Sid": "DenyDynamoDB",
            "Effect": "Deny",
            "Action": "dynamodb:*",
            "Resource": "*"
        }
    ]
}
```

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:*",
                "cloudwatch:*"
            ],
            "Resource": "*"
        }
    ]
}
```

More examples: https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_example-scps.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Organization – Moving Accounts

To migrate accounts from one organization to another

1. Remove the member account from the old organization
2. Send an invite to the new organization
3. Accept the invite to the new organization from the member account

If you want the master account of the old organization to also join the new organization, do the following:

1. Remove the member accounts from the organizations using procedure above
2. Delete the old organization
3. Repeat the process above to invite the old master account to the new org

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Conditions

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Deny",
        "Action": "*",
        "Resource": "*",
        "Condition": {
            "NotIpAddress": {
                "aws:SourceIp": [
                    "192.0.2.0/24",
                    "203.0.113.0/24"
                ]
            },
```

aws:SourceIP: restrict the client IP **from** which the API calls are being made

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowOnlyInsideEU",
            "Effect": "Allow",
            "Action": [
                "ec2:*",
                "rds:*",
                "dynamodb:*"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestedRegion": [
                        "eu-central-1",
                        "eu-west-1"
                    ]
                }
            }
        }
    ]
}
```

Aws:RequestedRegion: restrict the region The API calls are made **to**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Conditions

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "StartStopIfTags",
            "Effect": "Allow",
            "Action": [
                "ec2:StartInstances",
                "ec2:StopInstances",
                "ec2:DescribeTags"
            ],
            "Resource": "arn:aws:ec2:region:account-id:instance/*",
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/Project": "DataAnalytics",
                    "aws:PrincipalTag/Department": "Data"
                }
            }
        }
    ]
}
```

Restrict based on tags

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllActionsForEC2",
            "Effect": "Allow",
            "Action": "ec2:*",
            "Resource": "*"
        },
        {
            "Sid": "DenyStopAndTerminateWhenMFAIsNotPresent",
            "Effect": "Deny",
            "Action": [
                "ec2:StopInstances",
                "ec2:TerminateInstances"
            ],
            "Resource": "*",
            "Condition": {
                "BoolIfExists": {"aws:MultiFactorAuthPresent": false}
            }
        }
    ]
}
```

Force MFA

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

IAM for S3

- ListBucket permission applies to

  arn:aws:s3:::test

- => bucket level permission

- GetObject, PutObject, DeleteObject applies to

  arn:awn:s3:::test/*

- => object level permission

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::test"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": ["arn:aws:s3:::test/*"]
    }
  ]
}
```

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Roles vs Resource Based Policies

- Attach a policy to a resource (example: S3 bucket policy) versus attaching of a using a role as a proxy

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Roles vs Resource Based Policies

- When you assume a role (user, application or service), you give up your original permissions and take the permissions assigned to the role
- When using a resource based policy, the principal doesn't have to give up his permissions
- Example: User in account A needs to scan a DynamoDB table in Account A and dump it in an S3 bucket in Account B.
- Supported by: Amazon S3 buckets, SNS topics, SQS queues

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Permission Boundaries

- IAM Permission Boundaries are supported for users and roles (not groups)
- Advanced feature to use a managed policy to set the maximum permissions an IAM entity can get.

Example:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:*",
                "cloudwatch:*",
                "ec2:*"
            ],
            "Resource": "*"
        }
    ]
}
```

IAM Permission Boundary

+

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "iam:CreateUser",
    "Resource": "*"
  }
}
```

IAM Permissions Through IAM Policy

=

No Permissions

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

IAM Permission Boundaries

- Can be used in combinations of AWS Organizations SCP

https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_boundaries.html

Use cases

- Delegate responsibilities to non administrators within their permission boundaries, for example create new IAM users
- Allow developers to self-assign policies and manage their own permissions, while making sure they can't "escalate" their privileges (= make themselves admin)
- Useful to restrict one specific user (instead of a whole account using Organizations & SCP)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

IAM Policy Evaluation Logic

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_evaluation-logic.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example IAM Policy

- Can you perform sqs:CreateQueue?
- Can you perform sqs:DeleteQueue?
- Can you perform ec2:DescribeInstances?

```
{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Action": "sqs:*",
        "Effect": "Deny",
        "Resource": "*"
      },
      {
        "Action": [
          "sqs:DeleteQueue"
        ],
        "Effect": "Allow",
        "Resource": "*"
      }
    ]
}
```

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Resource Access Manager (RAM)

- Share AWS resources that you own with other AWS accounts
- Share with any account or within your Organization
- Avoid resource duplication!
- VPC Subnets:
  - allow to have all the resources launched in the same subnets
  - must be from the same AWS Organizations.
  - Cannot share security groups and default VPC
  - Participants can manage their own resources in there
  - Participants can't view, modify, delete resources that belong to other participants or the owner
- AWS Transit Gateway
- Route53 Resolver Rules
- License Manager Configurations

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Resource Access Manager – VPC example

- Each account…
  - is responsible for its own resources
  - cannot view, modify or delete other resources in other accounts
- Network is shared so…
  - Anything deployed in the VPC can talk to other resources in the VPC
  - Applications are accessed easily across accounts, using private IP!
  - Security groups from other accounts can be referenced for maximum security

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Single Sign-On (SSO)

- Centrally manage Single Sign-On to access **multiple accounts** and **3rd-party business applications.**
- Integrated with **AWS Organizations**
- **Supports SAML 2.0** markup
- Integration with on-premise **Active Directory**
- Centralized permission management
- Centralized auditing with CloudTrail

https://aws.amazon.com/blogs/security/introducing-aws-single-sign-on/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Single Sign-On (SSO) – Setup with AD

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

SSO – vs AssumeRoleWithSAML

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Security & Encryption

KMS, Encryption SDK, SSM Parameter Store

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why encryption?
Encryption in flight (SSL)

- Data is encrypted before sending and decrypted after receiving
- SSL certificates help with encryption (HTTPS)
- Encryption in flight ensures no MITM (man in the middle attack) can happen

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Why encryption?
Server side encryption at rest

- Data is encrypted after being received by the server
- Data is decrypted before being sent
- It is stored in an encrypted form thanks to a key (usually a data key)
- The encryption / decryption keys must be managed somewhere and the server must have access to it

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Why encryption?
Client side encryption

- Data is encrypted by the client and never decrypted by the server
- Data will be decrypted by a receiving client
- The server should not be able to decrypt the data
- Could leverage Envelope Encryption

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS KMS (Key Management Service)

- Anytime you hear "encryption" for an AWS service, it's most likely KMS
- Easy way to control access to your data, AWS manages keys for us
- Fully integrated with IAM for authorization
- Seamlessly integrated into:
  - Amazon EBS: encrypt volumes
  - Amazon S3: Server side encryption of objects
  - Amazon Redshift: encryption of data
  - Amazon RDS: encryption of data
  - Amazon SSM: Parameter store
  - Etc…
- But you can also use the CLI / SDK

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS KMS 101

- Anytime you need to share sensitive information… use KMS
  - Database passwords
  - Credentials to external service
  - Private Key of SSL certificates
- The value in KMS is that the CMK used to encrypt data can never be retrieved by the user, and the CMK can be rotated for extra security
- **Never ever store your secrets in plaintext, especially in your code!**
- Encrypted secrets can be stored in the code / environment variables
- **KMS can only help in encrypting up to 4KB of data per call**
- If data > 4 KB, use envelope encryption
- To give access to KMS to someone:
  - Make sure the Key Policy allows the user
  - Make sure the IAM Policy allows the API calls

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS KMS (Key Management Service)

- Able to fully manage the keys & policies:
  - Create
  - Rotation policies
  - Disable
  - Enable
- Able to audit key usage (using CloudTrail)
- Three types of Customer Master Keys (CMK):
  - AWS Managed Service Default CMK: **free**
  - User Keys created in KMS: **$1 / month**
  - User Keys imported (must be 256-bit symmetric key): **$1 / month**
- + pay for API call to KMS ($0.03 / 10000 calls)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How does KMS work?
API – Encrypt and Decrypt

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Encryption in AWS Services

- Requires migration (through Snapshot / Backup):
  - EBS Volumes
  - RDS databases
  - ElastiCache
  - EFS network file system

- In-place encryption:
  - S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Parameter Store

- Secure storage for configuration and secrets
- Optional Seamless Encryption using KMS
- Serverless, scalable, durable, easy SDK, free
- Version tracking of configurations / secrets
- Configuration management using path & IAM
- Notifications with CloudWatch Events
- Integration with CloudFormation

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Parameter Store Hierarchy

- /my-department/
  - my-app/
    - dev/
      - db-url
      - db-password
    - prod/
      - db-url
      - db-password
  - other-app/
- /other-department/

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Secrets Manager

- Newer service, meant for storing secrets
- Capability to force **rotation of secrets** every X days
- Automate generation of secrets on rotation (uses Lambda)
- Integration with **Amazon RDS** (MySQL, PostgreSQL, Aurora)
- Secrets are encrypted using KMS

- Mostly meant for RDS integration

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudHSM

- KMS => AWS manages the software for encryption
- CloudHSM => AWS provisions encryption **hardware**
- Dedicated Hardware (HSM = Hardware Security Module)
- You manage your own encryption keys entirely (not AWS)
- HSM device is tamper resistant, FIPS 140-2 Level 3 compliance
- **CloudHSM clusters are spread across Multi AZ (HA) – must setup**
- Supports both symmetric and **asymmetric** encryption (SSL/TLS keys)
- No free tier available
- Must use the CloudHSM Client Software
- Redshift supports CloudHSM for database encryption and key management
- Good option to use with SSE-C encryption

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudHSM Diagram

IAM permissions:

- CRUD an HSM Cluster

CloudHSM Software:

- Manage the Keys
- Manage the Users

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Shield

- AWS Shield Standard:
  - Free service that is activated for every AWS customer
  - Provides protection from attacks such as SYN/UDP Floods, Reflection attacks and other layer 3/layer 4 attacks
- AWS Shield Advanced:
  - Optional DDoS mitigation service ($3,000 per month per organization)
  - Protect against more sophisticated attack on Amazon EC2, Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator, and Route 53
  - 24/7 access to AWS DDoS response team (DRP)
  - Protect against higher fees during usage spikes due to DDoS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS WAF – Web Application Firewall

- Protects your web applications from common web exploits (Layer 7)
- **Layer 7 is HTTP** (vs Layer 4 is TCP)
- Deploy on **Application Load Balancer, API Gateway, CloudFront**

- Define Web ACL (Web Access Control List):
  - Rules can include: **IP addresses,** HTTP headers, HTTP body, or URI strings
  - Protects from common attack - **SQL injection** and **Cross-Site Scripting (XSS)**
  - Size constraints, **geo-match (block countries)**
  - **Rate-based rules** (to count occurrences of events) – **for DDoS protection**

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS Firewall Manager

- Manage rules in all accounts of an AWS Organization

- Common set of security rules
- WAF rules (Application Load Balancer, API Gateways, CloudFront)
- AWS Shield Advanced (ALB, CLB, Elastic IP, CloudFront)
- Security Groups for EC2 and ENI resources in VPC

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Sample Reference Architecture for DDoS Protection

https://aws.amazon.com/answers/networking/aws-ddos-attack-mitigation/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Shared Responsibility Model

- AWS responsibility - Security **of** the Cloud
  - Protecting infrastructure (hardware, software, facilities, and networking) that runs all of the AWS services
  - Managed services like S3, DynamoDB, RDS etc

- Customer responsibility - Security **in** the Cloud
  - For EC2 instance, customer is responsible for management of the guest OS (including security patches and updates), firewall & network configuration, IAM etc

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example, for RDS

- AWS responsibility:
  - Manage the underlying EC2 instance, disable SSH access
  - Automated DB patching
  - Automated OS patching
  - Audit the underlying instance and disks & guarantee it functions
- Your responsibility:
  - Check the ports / IP / security group inbound rules in DB's SG
  - In-database user creation and permissions
  - Creating a database with or without public access
  - Ensure parameter groups or DB is configured to only allow SSL connections
  - Database encryption setting

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example, for S3

- AWS responsibility:
  - Guarantee you get unlimited storage
  - Guarantee you get encryption
  - Ensure separation of the data between different customers
  - Ensure AWS employees can't access your data
- Your responsibility:
  - Bucket configuration
  - Bucket policy / public setting
  - IAM user and roles
  - Enabling encryption

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Shared Responsibility Model diagram

https://aws.amazon.com/compliance/shared-responsibility-model/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Networking - VPC

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Diagram

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Understanding CIDR - IPv4 (Classless Inter-Domain Routing)

- CIDR are used for Security Groups rules, or AWS networking in general

Source
0.0.0.0/0
122.149.196.85/32

- They help to define an IP address range
  - We've seen WW.XX.YY.ZZ/32 == one IP
  - We've seen 0.0.0.0/0 == all IPs
  - But we can define for ex: 192.168.0.0/26: 192.168.0.0 – 192.168.0.63 (64 IP)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Understanding CIDR

- A CIDR has two components:
  - The base IP (XX.XX.XX.XX)
  - The Subnet Mask (/26)

- The base IP represents an IP contained in the range
- The subnet masks defines how many bits can change in the IP

- The subnet mask can take two forms. Examples:
  - 255.255.255.0 ← less common
  - /24 ← more common

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Understanding CIDRs Subnet Masks

- The subnet masks basically allows part of the underlying IP to get additional next values from the base IP

- /32 allows for 1 IP = 2^0
- /31 allows for 2 IP = 2^1
- /30 allows for 4 IP = 2^2
- /29 allows for 8 IP = 2^3
- /28 allows for 16 IP = 2^4
- /27 allows for 32 IP = 2^5
- /26 allows for 64 IP = 2^6
- /25 allows for 128 IP = 2^7
- /24 allows for 256 IP = 2^8
- /16 allows for 65,536 IP = 2^16
- /0 allows for all IPs = 2^32

- Quick memo:
- /32 – no IP number can change
- /24 - last IP number can change
- /16 – last IP two numbers can change
- /8 – last IP three numbers can change
- /0 – all IP numbers can change

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Understanding CIDRs
Little exercise

- 192.168.0.0/24 = … ?
  - 192.168.0.0 – 192.168.0.255 (256 IP)
- 192.168.0.0/16 = … ?
  - 192.168.0.0 – 192.168.255.255 (65,536 IP)
- 134.56.78.123/32 = … ?
  - Just 134.56.78.123
- 0.0.0.0/0
  - All IP!
- When in doubt, use this website: https://www.ipaddressguide.com/cidr

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Private vs Public IP (IPv4) Allowed ranges

- The Internet Assigned Numbers Authority (IANA) established certain blocks of IPV4 addresses for the use of private (LAN) and public (Internet) addresses.

- Private IP can only allow certain values
  - 10.0.0.0 – 10.255.255.255 (10.0.0.0/8) <= in big networks
  - 172.16.0.0 – 172.31.255.255 (172.16.0.0/12) <= default AWS one
  - 192.168.0.0 – 192.168.255.255 (192.168.0.0/16) <= example: home networks

- All the rest of the IP on the internet are public IP

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Default VPC Walkthrough

- All new accounts have a default VPC
- New instances are launched into default VPC if no subnet is specified
- Default VPC have internet connectivity and all instances have public IP
- We also get a public and a private DNS name

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC in AWS – IPv4

- VPC = Virtual Private Cloud
- You can have multiple VPCs in a region (max 5 per region – soft limit)
- Max CIDR per VPC is 5. For each CIDR:
  - Min size is /28 = 16 IP Addresses
  - Max size is /16 = 65536 IP Addresses
- Because VPC is private, only the Private IP ranges are allowed:
  - 10.0.0.0 – 10.255.255.255 (10.0.0.0/8)
  - 172.16.0.0 – 172.31.255.255 (172.16.0.0/12)
  - 192.168.0.0 – 192.168.255.255 (192.168.0.0/16)
- <u>Your VPC CIDR should not overlap with your other networks (ex: corporate)</u>

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

State of Hands On

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Adding Subnets

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Subnets - IPv4

- AWS reserves 5 IPs address (first 4 and last 1 IP address) in each Subnet
- These 5 IPs are not available for use and cannot be assigned to an instance
- Ex, if CIDR block 10.0.0.0/24, reserved IP are:
  - 10.0.0.0: Network address
  - 10.0.0.1: Reserved by AWS for the VPC router
  - 10.0.0.2: Reserved by AWS for mapping to Amazon-provided DNS
  - 10.0.0.3: Reserved by AWS for future use
  - 10.0.0.255: Network broadcast address. AWS does not support broadcast in a VPC, therefore the address is reserved
- Exam Tip:
  - If you need 29 IP addresses for EC2 instances, you can't choose a Subnet of size /27 (32 IP)
  - You need at least 64 IP, Subnet size /26 (64-5 = 59 > 29, but 32-5 = 27 < 29)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Internet Gateways

- Internet gateways helps our VPC instances connect with the internet
- It scales horizontally and is HA and redundant
- Must be created separately from VPC
- One VPC can only be attached to one IGW and vice versa
- Internet Gateway is also a NAT for the instances that have a public IPv4

- Internet Gateways on their own do not allow internet access…
- Route tables must also be edited!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Adding IGW

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Editing Route Tables

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Instances – Network Address Translation (outdated but still at the exam)

- Allows instances in the private subnets to connect to the internet
- Must be launched in a public subnet
- Must disable EC2 flag: Source / Destination Check
- Must have Elastic IP attached to it
- Route table must be configured to route traffic from private subnets to NAT Instance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Instance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Instances – Comments

- Amazon Linux AMI pre-configured are available
- Not highly available / resilient setup out of the box
- => Would need to create ASG in multi AZ + resilient user-data script
- Internet traffic bandwidth depends on EC2 instance performance
- Must manage security groups & rules:
  - Inbound:
    - Allow HTTP / HTTPS Traffic coming from Private Subnets
    - Allow SSH from your home network (access is provided through Internet Gateway)
  - Outbound:
    - Allow HTTP / HTTPS traffic to the internet

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Gateway

- AWS managed NAT, higher bandwidth, better availability, no admin
- Pay by the hour for usage and bandwidth
- NAT is created in a specific AZ, uses an EIP
- Cannot be used by an instance in that subnet (only from other subnets)
- Requires an IGW (Private Subnet => NAT => IGW)
- 5 Gbps of bandwidth with automatic scaling up to 45 Gbps
- No security group to manage / required

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Gateway

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

NAT Gateway with High Availability

- NAT Gateway is resilient within a single-AZ
- Must create **multiple NAT Gateway** in **multiple AZ** for fault-tolerance
- There is no cross AZ failover needed because if an AZ goes down it doesn't need NAT

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

NAT Instance vs Gateway

- See: https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DNS Resolution in VPC

- enableDnsSupport: (= DNS Resolution setting)
  - Default True
  - Helps decide if DNS resolution is supported for the VPC
  - If True, queries the AWS DNS server at 169.254.169.253
- enableDnsHostname: (= DNS Hostname setting)
  - False by default for newly created VPC, True by default for Default VPC
  - Won't do anything unless enableDnsSupport=true
  - If True, Assign public hostname to EC2 instance if it has a public
- If you use custom DNS domain names in a private zone in Route 53, you must set both these attributes to true

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Network ACLs & Security Group Incoming Request

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Network ACLs & Security Group Outgoing Request

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Network ACLs

- NACL are like a firewall which control traffic from and to subnet
- Default NACL allows everything outbound and everything inbound
- **One NACL per Subnet, new Subnets are assigned the Default NACL**
- Define NACL rules:
  - Rules have a number (1-32766) and higher precedence with a lower number
  - E.g. If you define #100 ALLOW <IP> and #200 DENY <IP> , IP will be allowed
  - Last rule is an asterisk (*) and denies a request in case of no rule match
  - AWS recommends adding rules by increment of 100
- Newly created NACL will deny everything
- NACL are a great way of blocking a specific IP at the subnet level

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Network ACLs vs Security Groups

Security Group	Network ACL
Operates at the instance level	Operates at the subnet level
Supports allow rules only	Supports allow rules and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (therefore, you don't have to rely on users to specify the security group)

https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Security.html#VPC_Security_Comparison

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Example Network ACL with Ephemeral Ports

- https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Peering

- Connect two VPC, privately using AWS' network
- Make them behave as if they were in the same network
- Must not have overlapping CIDR
- VPC Peering connection is **not transitive** (must be established for each VPC that need to communicate with one another)
- You can do VPC peering with another AWS account
- You must update route tables in each VPC's subnets to ensure instances can communicate

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Peering – Good to know

- VPC peering can work **inter-region, cross-account**
- You can reference a security group of a peered VPC (works cross account)

Type	Protocol	Port Range	Source
HTTP	TCP	80	sg-00d2b0f5fd6de757e
HTTP	TCP	80	sg-013347765f7a63aae/12356788

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Peering

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Endpoints

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

VPC Endpoints

- Endpoints allow you to connect to AWS Services using a private network instead of the public www network
- They scale horizontally and are redundant
- They remove the need of IGW, NAT, etc… to access AWS Services
- Interface: provisions an ENI (private IP address) as an entry point (must attach security group) – most AWS services
- Gateway: provisions a target and must be used in a route table – S3 and DynamoDB
- In case of issues:
  - Check DNS Setting Resolution in your VPC
  - Check Route Tables

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Flow Logs

- Capture information about IP traffic going into your interfaces:
  - VPC Flow Logs
  - Subnet Flow Logs
  - Elastic Network Interface Flow Logs
- Helps to monitor & troubleshoot connectivity issues
- Flow logs data can go to S3 / CloudWatch Logs
- Captures network information from AWS managed interfaces too: ELB, RDS, ElastiCache, Redshift, WorkSpaces

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Flow Logs

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Flow Log Syntax

- *<version> <account-id> <interface-id> <srcaddr> <dstaddr> <srcport> <dstport> <protocol> <packets> <bytes> <start> <end> <action> <log-status>*
- Srcaddr, dstaddr help identify problematic IP
- Srcport, dstport help identity problematic ports
- Action : success or failure of the request due to Security Group / NACL
- Can be used for analytics on usage patterns, or malicious behavior
- Flow logs example: https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html#flow-log-records
- Query VPC flow logs using Athena on S3 or CloudWatch Logs Insights

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Flow Logs: Looking at "ACTION" field
How to troubleshoot SG vs NACL issue?

For incoming requests
Inbound REJECT: NACL or SG
Inbound ACCEPT, outbound REJECT: NACL

For outgoing requests
Outbound REJECT: NACL or SG
Outbound ACCEPT, inbound REJECT: NACL

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Bastion Hosts

- We can use a Bastion Host to SSH into our private instances
- The bastion is in the public subnet which is then connected to all other private subnets
- **Bastion Host security group must be tightened**

- Exam Tip: Make sure the bastion host only has port 22 traffic from the IP you need, not from the security groups of your other instances

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Site to Site VPN

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Site to Site VPN

- Virtual Private Gateway:
  - VPN concentrator on the AWS side of the VPN connection
  - VGW is created and attached to the VPC from which you want to create the Site-to-Site VPN connection
  - Possibility to customize the ASN
- Customer Gateway:
  - Software application or physical device on customer side of the VPN connection
  - https://docs.aws.amazon.com/vpc/latest/adminguide/Introduction.html#DevicesTested
  - IP Address:
    - Use static, internet-routable IP address for your customer gateway device.
    - If behind a CGW behind NAT (with NAT-T), use the public IP address of the NAT

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Direct Connect

- Provides a dedicated **private** connection from a remote network to your VPC
- Dedicated connection must be setup between your DC and AWS Direct Connect locations
- You need to setup a Virtual Private Gateway on your VPC
- Access public resources (S3) and private (EC2) on same connection
- Use Cases:
  - Increase bandwidth throughput - working with large data sets – lower cost
  - More consistent network experience - applications using real-time data feeds
  - Hybrid Environments (on prem + cloud)
- Supports both IPv4 and IPv6

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Direct Connect Diagram

https://docs.aws.amazon.com/directconnect/latest/UserGuide/images/direct_connect_overview.png

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Direct Connect Gateway

- If you want to setup a Direct Connect to one or more VPC in many different regions (same account), you must use a Direct Connect Gateway

https://docs.aws.amazon.com/directconnect/latest/UserGuide/direct-connect-gateways.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Direct Connect – Connection Types

- **Dedicated Connections:** 1Gbps and 10 Gbps capacity
  - Physical ethernet port dedicated to a customer
  - Request made to AWS first, then completed by AWS Direct Connect Partners

- **Hosted Connections:** 50Mbps, 500 Mbps, to 10 Gbps
  - Connection requests are made via AWS Direct Connect Partners
  - Capacity can be **added or removed on demand**
  - 1, 2, 5, 10 Gbps available at select AWS Direct Connect Partners

- Lead times are often longer than 1 month to establish a new connection

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Direct Connect – Encryption

- Data in transit is <u>not encrypted</u> but is private
- AWS Direct Connect + VPN provides an IPsec-encrypted private connection
- Good for an extra level of security, but slightly more complex to put in place

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Egress Only Internet Gateway

- Egress only Internet Gateway is for IPv6 only
- Similar function as a NAT, but a NAT is for IPv4
- Good to know: IPv6 are all public addresses
- Therefore all our instances with IPv6 are publicly accessibly
- Egress Only Internet Gateway gives our IPv6 instances access to the internet, but they won't be directly reachable by the internet
- After creating an Egress Only Internet Gateway, edit the route tables

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Exposing services in your VPC to other VPC

- Option 1: make it public
  - Goes through the public www
  - Tough to manage access

- Option 2: VPC peering
  - Must create many peering relations
  - Opens the **whole** network

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS PrivateLink (VPC Endpoint Services)

- Most secure & scalable way to expose a service to 1000s of VPC (own or other accounts)
- Does not require VPC peering, internet gateway, NAT, route tables…
- Requires a network load balancer (Service VPC) and ENI (Customer VPC)
- If the NLB is in multiple AZ, and the ENI in multiple AZ, the solution is fault tolerant!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

EC2-Classic & AWS ClassicLink (deprecated)

- **EC2-Classic:** instances run in a single network shared with other customers
- **Amazon VPC:** your instances run logically isolated to your AWS account

- **ClassicLink** allows you to link EC2-Classic instances to a VPC in your account
  - Must associate a security group
  - Enables communication using private IPv4 addresses
  - Removes the need to make use of public IPv4 addresses or Elastic IP addresses

- Likely to be distractors at the exam

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS VPN CloudHub

- Provide secure communication between sites, if you have multiple VPN connections
- Low cost hub-and-spoke model for primary or secondary network connectivity between locations
- It's a VPN connection so it goes over the public internet

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Diagram

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

VPC Section Summary (1/3)

- **CIDR:** IP Range
- **VPC:** Virtual Private Cloud => we define a list of IPv4 & IPv6 CIDR
- **Subnets:** Tied to an AZ, we define a CIDR
- **Internet Gateway:** at the VPC level, provide IPv4 & IPv6 Internet Access
- **Route Tables:** must be edited to add routes from subnets to the IGW, VPC Peering Connections, VPC Endpoints, etc…
- **NAT Instances:** gives internet access to instances in private subnets. Old, must be setup in a public subnet, disable Source / Destination check flag
- **NAT Gateway:** managed by AWS, provides scalable internet access to private instances, IPv4 only
- **Private DNS + Route 53:** enable DNS Resolution + DNS hostnames (VPC)
- **NACL:** Stateless, subnet rules for inbound and outbound, don't forget ephemeral ports
- **Security Groups:** Stateful, operate at the EC2 instance level

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Section Summary (2/3)

- **VPC Peering:** Connect two VPC with non overlapping CIDR, non transitive
- **VPC Endpoints:** Provide private access to AWS Services (S3, DynamoDB, CloudFormation, SSM) within VPC
- **VPC Flow Logs:** Can be setup at the VPC / Subnet / ENI Level, for ACCEPT and REJECT traffic, helps identifying attacks, analyze using Athena or CloudWatch Log Insights
- **Bastion Host:** Public instance to SSH into, that has SSH connectivity to instances in private subnets
- **Site to Site VPN:** setup a Customer Gateway on DC, a Virtual Private Gateway on VPC, and site-to-site VPN over public internet
- **Direct Connect:** setup a Virtual Private Gateway on VPC, and establish a direct private connection to an AWS Direct Connect Location
- **Direct Connect Gateway:** setup a Direct Connect to many VPC in different regions
- **Internet Gateway Egress:** like a NAT Gateway, but for IPv6

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

VPC Section Summary (3/3)

- **Private Link / VPC Endpoint Services:**
  - connect services privately from your service VPC to customers VPC
  - Doesn't need VPC peering, public internet, NAT gateway, route tables
  - Must be used with Network Load Balancer & ENI
- **ClassicLink:** connect EC2-Classic instances privately to your VPC
- **VPN CloudHub:** hub-and-spoke VPN model to connect your sites

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Networking Costs in AWS per GB - Simplified

- Use Private IP instead of Public IP for good savings and better network performance
- Use same AZ for maximum savings (at the cost of high availability)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Disaster Recovery Overview

- Any event that has a negative impact on a company's business continuity or finances is a disaster
- Disaster recovery (DR) is about preparing for and recovering from a disaster
- What kind of disaster recovery?
  - On-premise => On-premise: traditional DR, and very expensive
  - On-premise => AWS Cloud: hybrid recovery
  - AWS Cloud Region A => AWS Cloud Region B
- Need to define two terms:
  - RPO: Recovery Point Objective
  - RTO: Recovery Time Objective

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

RPO and RTO

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Disaster Recovery Strategies

- Backup and Restore
- Pilot Light
- Warm Standby
- Hot Site / Multi Site Approach

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Backup and Restore (High RPO)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Disaster Recovery – Pilot Light

- A small version of the app is always running in the cloud
- Useful for the critical core (pilot light)
- Very similar to Backup and Restore
- Faster than Backup and Restore as critical systems are already up

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Warm Standby

- Full system is up and running, but at minimum size
- Upon disaster, we can scale to production load

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Multi Site / Hot Site Approach

- Very low RTO (minutes or seconds) – very expensive
- Full Production Scale is running AWS and On Premise

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

All AWS Multi Region

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Disaster Recovery Tips

- Backup
  - EBS Snapshots, RDS automated backups / Snapshots, etc…
  - Regular pushes to S3 / S3 IA / Glacier, Lifecycle Policy, Cross Region Replication
  - From On-Premise: Snowball or Storage Gateway
- High Availability
  - Use Route53 to migrate DNS over from Region to Region
  - RDS Multi-AZ, ElastiCache Multi-AZ, EFS, S3
  - Site to Site VPN as a recovery from Direct Connect
- Replication
  - RDS Replication (Cross Region), AWS Aurora + Global Databases
  - Database replication from on-premise to RDS
  - Storage Gateway
- Automation
  - CloudFormation / Elastic Beanstalk to re-create a whole new environment
  - Recover / Reboot EC2 instances with CloudWatch if alarms fail
  - AWS Lambda functions for customized automations
- Chaos
  - Netflix has a "simian-army" randomly terminating EC2

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

DMS – Database Migration Service

- Quickly and securely migrate databases to AWS, resilient, self healing
- The source database remains available during the migration
- Supports:
  - Homogeneous migrations: ex Oracle to Oracle
  - Heterogeneous migrations: ex Microsoft SQL Server to Aurora
- Continuous Data Replication using CDC
- You must create an EC2 instance to perform the replication tasks

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

DMS Sources and Targets

SOURCES:

- On-Premise and EC2 instances databases: *Oracle, MS SQL Server, MySQL, MariaDB, PostgreSQL, MongoDB, SAP, DB2*
- Azure: *Azure SQL Database*
- Amazon RDS: all including Aurora
- Amazon S3

TARGETS:

- On-Premise and EC2 instances databases: Oracle, MS SQL Server, MySQL, MariaDB, PostgreSQL, SAP
- Amazon RDS
- Amazon Redshift
- Amazon DynamoDB
- Amazon S3
- ElasticSearch Service
- Kinesis Data Streams
- DocumentDB

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Schema Conversion Tool (SCT)

- Convert your Database's Schema from one engine to another
- Example OLTP: (SQL Server or Oracle) to MySQL, PostgreSQL, Aurora
- Example OLAP: (Teradata or Oracle) to Amazon Redshift

- You do not need to use SCT if you are migrating the same DB engine
  - Ex: On-Premise PostgreSQL => RDS PostgreSQL
  - The DB engine is still PostgreSQL (RDS is the platform)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

On-Premise strategy with AWS

- Ability to download Amazon Linux 2 AMI as a VM (.iso format)
  - VMWare, KVM, VirtualBox (Oracle VM), Microsoft Hyper-V
- VM Import / Export
  - Migrate existing applications into EC2
  - Create a DR repository strategy for your on-premise VMs
  - Can export back the VMs from EC2 to on-premise
- AWS Application Discovery Service
  - Gather information about your on-premise servers to plan a migration
  - Server utilization and dependency mappings
  - Track with AWS Migration Hub
- AWS Database Migration Service (DMS)
  - replicate On-premise => AWS , AWS => AWS, AWS => On-premise
  - Works with various database technologies (Oracle, MySQL, DynamoDB, etc..)
- AWS Server Migration Service (SMS)
  - Incremental replication of on-premise live servers to AWS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS DataSync

- Move large amount of data from on-premise to AWS
- Can synchronize to: **Amazon S3, Amazon EFS, Amazon FSx for Windows**
- Move data from your NAS or file system via **NFS** or **SMB**
- Replication tasks can be scheduled hourly, daily, weekly
- Leverage the DataSync agent to connect to your systems

NFS / SMB to AWS (S3, EFS, FSx for Windows)

EFS to EFS

https://docs.aws.amazon.com/datasync/latest/userguide/how-datasync-works.html

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Transferring large amount of data into AWS

- Example: transfer 200 TB of data in the cloud. We have a 100 Mbps internet connection.
- **Over the internet / Site-to-Site VPN:**
  - Immediate to setup
  - Will take 200(TB)*1000(GB)*1000(MB)*8(Mb)/100 Mbps = 16,000,000s = 185d
- **Over direct connect 1Gbps:**
  - Long for the one-time setup (over a month)
  - Will take 200(TB)*1000(GB)*8(Gb)/1 Gbps = 1,600,000s = 18.5d
- **Over Snowball:**
  - Will take 2 to 3 snowballs in parallel
  - Takes about 1 week for the end-to-end transfer
  - Can be combined with DMS
- **For on-going replication / transfers:** Site-to-Site VPN or DX with DMS or DataSync

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Extra Solution Architecture discussions

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Lambda, SNS & SQS

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Fan Out Pattern: deliver to multiple SQS

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

S3 Events

- S3:ObjectCreated, S3:ObjectRemoved, S3:ObjectRestore, S3:Replication…
- Object name filtering possible (*.jpg)
- Use case: generate thumbnails of images uploaded to S3
- Can create as many "S3 events" as desired

- S3 event notifications typically deliver events in seconds but can sometimes take a minute or longer
- If two writes are made to a single non-versioned object at the same time, it is possible that only a single event notification will be sent
- If you want to ensure that an event notification is sent for every successful write, you can enable versioning on your bucket.

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Caching Strategies

Caching, TTL, Network, Computation, Cost, Latency

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Blocking an IP address

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Blocking an IP address – with an ALB

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Blocking an IP address – with an NLB

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Blocking an IP address – ALB + WAF

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Blocking an IP address – ALB, CloudFront WAF

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

High Performance Computing (HPC)

- The cloud is the perfect place to perform HPC
- You can create a very high number of resources in no time
- You can speed up time to results by adding more resources
- You can pay only for the systems you have used

- Perform genomics, computational chemistry, financial risk modeling, weather prediction, machine learning, deep learning, autonomous driving

- Which services help perform HPC?

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Data Management & Transfer

- AWS Direct Connect:
  - Move GB/s of data to the cloud, over a private secure network
- Snowball & Snowmobile
  - Move PB of data to the cloud
- AWS DataSync
  - Move large amount of data between on-premise and S3, EFS, FSx for Windows

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Compute and Networking

- EC2 Instances:
  - CPU optimized, GPU optimized
  - Spot Instances / Spot Fleets for cost savings + Auto Scaling

- EC2 Placement Groups: **Cluster** for good network performance

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Compute and Networking

- EC2 Enhanced Networking (SR-IOV)
  - Higher bandwidth, higher PPS (packet per second), lower latency
  - Option 1: **Elastic Network Adapter (ENA)** up to 100 Gbps
  - Option 2: Intel 82599 VF up to 10 Gbps – LEGACY

- Elastic Fabric Adapter (EFA)
  - Improved ENA for **HPC**, only works for **Linux**
  - Great for inter-node communications, **tightly coupled workloads**
  - Leverages Message Passing Interface (MPI) standard
  - Bypasses the underlying Linux OS to provide low-latency, reliable transport

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Storage

- Instance-attached storage:
  - **EBS:** scale up to 64000 IOPS with io1 Provisioned IOPS
  - **Instance Store:** scale to millions of IOPS, linked to EC2 instance, low latency

- Network storage:
  - **Amazon S3:** large blob, not a file system
  - **Amazon EFS:** scale IOPS based on total size, or use provisioned IOPS
  - **Amazon FSx for Lustre:**
    - HPC optimized distributed file system, millions of IOPS
    - Backed by S3

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Automation and Orchestration

- AWS Batch
  - **AWS Batch** supports multi-node parallel jobs, which enables you to run single jobs that span multiple **EC2** instances.
  - Easily schedule jobs and launch EC2 instances accordingly

- AWS ParallelCluster
  - Open source cluster management tool to deploy HPC on AWS
  - Configure with text files
  - Automate creation of VPC, Subnet, cluster type and instance types

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Creating a highly available EC2 instance

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Creating a highly available EC2 instance With an Auto Scaling Group

ASG Settings

1 min
1 max
1 desired
>= 2 AZ

EC2 user data to attach
The Elastic IP

EC2 instance role to
Allow API calls to attach
The Elastic IP

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Creating a highly available EC2 instance With ASG + EBS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

High Availability for a Bastion Host

- HA options for the bastion host
  - Run 2 across 2 AZ
  - Run 1 across 2 AZ with 1 ASG 1:1:1
- Routing to the bastion host
  - If 1 bastion host, use an elastic IP with ec2 user-data script to access it
  - If 2 bastion hosts, use an Network Load Balancer (layer 4) deployed in multiple AZ
  - If NLB, the bastion hosts can live in the private subnet directly
- Note: Can't use ALB as the ALB is layer 7 (HTTP protocol)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Other Services

Overview of Services that might come up in a few questions

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Continuous Integration

- Developers push the code to a code repository often (GitHub / CodeCommit / Bitbucket / etc…)
- A testing / build server checks the code as soon as it's pushed (CodeBuild / Jenkins CI / etc…)
- The developer gets feedback about the tests and checks that have passed / failed

- Find bugs early, fix bugs
- Deliver faster as the code is tested
- Deploy often
- Happier developers, as they're unblocked

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Continuous Delivery

- Ensure that the software can be released reliably whenever needed.
- Ensures deployments happen often and are quick
- Shift away from "one release every 3 months" to "5 releases a day"
- That usually means automated deployment
  - CodeDeploy
  - Jenkins CD
  - Spinnaker
  - Etc…

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Technology Stack for CICD

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Infrastructure as Code

- Currently, we have been doing a lot of manual work
- All this manual work will be very tough to reproduce:
  - In another region
  - in another AWS account
  - Within the same region if everything was deleted

- Wouldn't it be great, if all our infrastructure was… code?
- That code would be deployed and create / update / delete our infrastructure

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

What is CloudFormation

- CloudFormation is a declarative way of outlining your AWS Infrastructure, for any resources (most of them are supported).
- For example, within a CloudFormation template, you say:
  - I want a security group
  - I want two EC2 machines using this security group
  - I want two Elastic IPs for these EC2 machines
  - I want an S3 bucket
  - I want a load balancer (ELB) in front of these machines
- Then CloudFormation creates those for you, in the **right order**, with the **exact configuration** that you specify

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Benefits of AWS CloudFormation (1/2)

- Infrastructure as code
  - No resources are manually created, which is excellent for control
  - The code can be version controlled for example using git
  - Changes to the infrastructure are reviewed through code

- Cost
  - Each resources within the stack is tagged with an identifier so you can easily see how much a stack costs you
  - You can estimate the costs of your resources using the CloudFormation template
  - Savings strategy: In Dev, you could automation deletion of templates at 5 PM and recreated at 8 AM, safely

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Benefits of AWS CloudFormation (2/2)

- Productivity
  - Ability to destroy and re-create an infrastructure on the cloud on the fly
  - Automated generation of Diagram for your templates!
  - Declarative programming (no need to figure out ordering and orchestration)

- Separation of concern: create many stacks for many apps, and many layers. Ex:
  - VPC stacks
  - Network stacks
  - App stacks

- Don't re-invent the wheel
  - Leverage existing templates on the web!
  - Leverage the documentation

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How CloudFormation Works

- Templates have to be uploaded in S3 and then referenced in CloudFormation
- To update a template, we can't edit previous ones. We have to re-upload a new version of the template to AWS
- Stacks are identified by a name
- Deleting a stack deletes every single artifact that was created by CloudFormation.

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Deploying CloudFormation templates

- Manual way:
  - Editing templates in the CloudFormation Designer
  - Using the console to input parameters, etc

- Automated way:
  - Editing templates in a YAML file
  - Using the AWS CLI (Command Line Interface) to deploy the templates
  - Recommended way when you fully want to automate your flow

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFormation Building Blocks

Templates components

1. **Resources: your AWS resources declared in the template (MANDATORY)**
2. Parameters: the dynamic inputs for your template
3. Mappings: the static variables for your template
4. Outputs: References to what has been created
5. Conditionals: List of conditions to perform resource creation
6. Metadata

Templates helpers:

1. References
2. Functions

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Note:
This is an introduction to CloudFormation

- It can take over 3 hours to properly learn and master CloudFormation
- This lecture is meant so you get a good idea of how it works
- The exam expects you to understand how to read CloudFormation

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

CloudFormation - StackSets

- Create, update, or delete stacks across **multiple accounts and regions** with a single operation
- Administrator account to create StackSets
- Trusted accounts to create, update, delete stack instances from StackSets
- When you update a stack set, *all* associated stack instances are updated throughout all accounts and regions.

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS ECS – Elastic Container Service

- ECS is a container orchestration service
- ECS helps you run Docker containers on EC2 machines
- ECS is complicated, and made of:
  - "ECS Core": Running ECS on user-provisioned EC2 instances
  - Fargate: Running ECS tasks on AWS-provisioned compute (serverless)
  - EKS: Running ECS on AWS-powered Kubernetes (running on EC2)
  - ECR: Docker Container Registry hosted by AWS
- ECS & Docker are very popular for **microservices**
- **For now, for the exam, only "ECS Core" & ECR is in scope**
- IAM security and roles at the ECS task level

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

What's Docker?

- Docker is a "container technology"
- Run a containerized application on any machine with Docker installed
- **Containers allows our application to work the same way anywhere**
- Containers are isolated from each other
- Control how much memory / CPU is allocated to your container
- Ability to restrict network rules
- More efficient than Virtual machines
- Scale containers up and down very quickly (seconds)

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS ECS – Use cases

- Run microservices
  - Ability to run multiple docker containers on the same machine
  - Easy service discovery features to enhance communication
  - Direct integration with Application Load Balancers
  - Auto scaling capability
- Run batch processing / scheduled tasks
  - Schedule ECS containers to run on On-demand / Reserved / Spot instances
- Migrate applications to the cloud
  - Dockerize legacy applications running on premise
  - Move Docker containers to run on ECS

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS ECS – Concepts

- **ECS cluster:** set of EC2 instances
- **ECS service:** applications definitions running on ECS cluster
- **ECS tasks + definition:** containers running to create the application
- **ECS IAM roles:** roles assigned to tasks to interact with AWS

NOT FOR DISTRIBUTION @ Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS ECS – ALB integration

- Application Load Balancer (ALB) has a direct integration feature with ECS called "port mapping"
- This allows you to run multiple instances of the same application on the same EC2 machine
- Use cases:
  - Increased resiliency even if running on one EC2 instance
  - Maximize utilization of CPU / cores
  - Ability to perform rolling upgrades without impacting application uptime

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

AWS ECS – ECS Setup & Config file

- Run an EC2 instance, install the ECS agent with ECS config file
- Or use an ECS-ready Linux AMI (still need to modify config file)

- ECS Config file is at **`/etc/ecs/ecs.config`**

```
ECS_CLUSTER=MyCluster      #Assign EC2 instance to an ECS cluster
ECS_ENGINE_AUTH_DATA={…}  #to pull images from private registries
ECS_AVAILABLE_LOGGING_DRIVERS=[…]   #CloudWatch container logging
ECS_ENABLE_TASK_IAM_ROLE=true       #Enable IAM roles for ECS tasks
```

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS ECR – Elastic Container Registry

- Store, managed and deploy your containers on AWS
- Fully integrated with IAM & ECS
- Sent over HTTPS (encryption in flight) and encrypted at rest

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

ECS - IAM Task Roles

- The EC2 instance should have an IAM role allowing it to access the ECS service (for the ECS agent)
- Each ECS task should have an ECS IAM task role to perform their API calls
- Use the "taskRoleArn" parameter in a task definition

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Fargate

- When launching an ECS Cluster, we have to create our EC2 instances
- If we need to scale, we need to add EC2 instances
- So we manage infrastructure…

- With Fargate, it's all Serverless!
- We don't provision EC2 instances
- We just create task definitions, and AWS will run our containers for us
- To scale, just increase the task number. Simple! No more EC2 ☺

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Step Functions

- Build serverless visual workflow to orchestrate your Lambda functions
- Represent flow as a JSON **state machine**
- Features: sequence, parallel, conditions, timeouts, error handling…
- Can also integrate with EC2, ECS, On premise servers, API Gateway
- Maximum execution time of 1 year
- Possibility to implement human approval feature
- Use cases:
  - Order fulfillment
  - Data processing
  - Web applications
  - Any workflow

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Visual workflow in Step Functions

NOT FOR DISTRIBU
rmulus.com

© Stephane Maarek

AWS SWF – Simple Workflow Service

- Coordinate work amongst applications
- Code runs on EC2 (not serverless)
- 1 year max runtime
- Concept of "activity step" and "decision step"
- Has built-in "human intervention" step
- Example: order fulfilment from web to warehouse to delivery
- Step Functions is recommended to be used for new applications, except:
  - If you need external signals to intervene in the processes
  - If you need child processes that return values to parent processes

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Amazon EMR

- EMR stands for "Elastic MapReduce"
- EMR helps creating **Hadoop clusters (Big Data)** to analyze and process vast amount of data
- The clusters can be made of hundreds of EC2 instances
- Also supports Apache Spark, HBase, Presto, Flink…
- EMR takes care of all the provisioning and configuration
- Auto-scaling and integrated with Spot instances
- Use cases: data processing, machine learning, web indexing, big data…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Glue

- **Fully-managed ETL (Extract, Transform & Load) service**
- Automating time consuming steps of data preparation for analytics
- **Serverless**, pay as you go, fully managed, provisions Apache Spark
- Crawls data sources and identifies data formats (schema inference)
- Automated **Code Generation**
- Sources: Amazon Aurora, RDS, Redshift & S3
- Sinks: S3, Redshift, etc…
- **Glue Data Catalog:** Metadata (definition & schema) of the Source Tables

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Opsworks

- Chef & Puppet help you perform server configuration automatically, or repetitive actions
- They work great with EC2 & On Premise VM
- AWS Opsworks = Managed Chef & Puppet
- It's an alternative to AWS SSM

- No hands on here, no knowledge of chef and puppet needed
- In the exam: Chef & Puppet needed => AWS Opsworks

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

Quick word on Chef / Puppet

- They help with managing configuration as code
- Helps in having consistent deployments
- Works with Linux / Windows
- Can automate: user accounts, cron, ntp, packages, services…

- They leverage "Recipes" or "Manifests"

- Chef / Puppet have similarities with SSM / Beanstalk / CloudFormation but they're open-source tools that work cross-cloud

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Elastic Transcoder

- **Convert media files (video + music)** stored in S3 into various formats for tablets, PC, Smartphone, TV, etc
- Features: bit rate optimization, thumbnail, watermarks, captions, DRM, progressive download, encryption
- 4 Components:
  - Jobs: what does the work of the transcoder
  - Pipeline: Queue that manages the transcoding job
  - Presets: Template for converting media from one format to another
  - Notifications: SNS for example
- Pay for what you use, scales automatically, fully managed

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS Organizations

- Global service
- Allows to manage multiple AWS accounts
- The main account is the master account – you can't change it
- Other accounts are member accounts
- Member accounts can only be part of one organization
- Consolidated Billing across all accounts - single payment method
- Pricing benefits from aggregated usage (volume discount for EC2, S3…)
- API is available to automate AWS account creation

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

OU & Service Control Policies (SCPs)

- Organize accounts in Organizational Unit (OU)
  - Can be anything: dev / test / prod or Finance / HR / IT
  - Can nest OU within OU
- Apply Service Control Policies (SCPs) to OU
  - Permit / Deny access to AWS services
  - SCP has a similar syntax to IAM
  - It's a filter to IAM
- Helpful for sandbox account creation
- Helpful to separate dev and prod resources
- Helpful to only allow approved services

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS WorkSpaces

- Managed, Secure Cloud Desktop
- Great to eliminate management of on-premise VDI (Virtual Desktop Infrastructure)
- On Demand, pay per by usage
- Secure, Encrypted, Network Isolation
- Integrated with Microsoft Active Directory

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

AWS AppSync

- Store and sync data across mobile and web apps in real-time
- Makes use of GraphQL (mobile technology from Facebook)
- Client Code can be generated automatically
- Integrations with DynamoDB / Lambda
- Real-time subscriptions
- Offline data synchronization (replaces Cognito Sync)
- Fine Grained Security

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

White Papers and Architectures

Well Architected Framework, Disaster Recovery, etc…

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Section Overview

- Well Architected Framework Whitepaper
- Well Architected Tool
- AWS Trusted Advisor
- Reference architectures resources (for real-world)
- Disaster Recovery on AWS Whitepaper

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Well Architected Framework General Guiding Principles

- Stop guessing your capacity needs
- Test systems at production scale
- Automate to make architectural experimentation easier
- Allow for evolutionary architectures
  - Design based on changing requirements
- Drive architectures using data
- Improve through game days
  - Simulate applications for flash sale days

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Well Architected Framework
5 Pillars

- 1) Operational Excellence
- 2) Security
- 3) Reliability
- 4) Performance Efficiency
- 5) Cost Optimization

- They are not something to balance, or trade-offs, they're a synergy

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Well Architected Framework

- It's also **questions**!
- Let's look into the Well-Architected Tool
- https://console.aws.amazon.com/wellarchitected

AWS Well-Architected Tool

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

1) Operational Excellence

- Includes the ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures
- Design Principles
  - **Perform operations as code** - Infrastructure as code
  - **Annotate documentation** - Automate the creation of annotated documentation after every build
  - **Make frequent, small, reversible changes** - So that in case of any failure, you can reverse it
  - **Refine operations procedures frequently** - And ensure that team members are familiar with it
  - **Anticipate failure**
  - **Learn from all operational failures**

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Operational Excellence AWS Services

- Prepare

AWS CloudFormation

AWS Config

- Operate

AWS CloudFormation

AWS Config

AWS CloudTrail

Amazon CloudWatch

AWS X-Ray

- Evolve

AWS CloudFormation

AWS CodeBuild

AWS CodeCommit

AWS CodeDeploy

AWS CodePipeline

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

2) Security

- Includes the ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies
- Design Principles
  - **Implement a strong identity foundation** - Centralize privilege management and reduce (or even eliminate) reliance on long-term credentials - Principle of least privilege - IAM
  - **Enable traceability** - Integrate logs and metrics with systems to automatically respond and take action
  - **Apply security at all layers** - Like edge network, VPC, subnet, load balancer, every instance, operating system, and application
  - **Automate security best practices**
  - **Protect data in transit and at rest** - Encryption, tokenization, and access control
  - **Keep people away from data** - Reduce or eliminate the need for direct access or manual processing of data
  - **Prepare for security events** - Run incident response simulations and use tools with automation to increase your speed for detection, investigation, and recovery

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Security
AWS Services

- Identity and Access Management

IAM

AWS-STS

MFA token

AWS Organizations

- Detective Controls

AWS Config

AWS CloudTrail

Amazon CloudWatch

- Infrastructure Protection

Amazon CloudFront

Amazon VPC

AWS Shield

AWS WAF

Amazon Inspector

- Data Protection:

KMS

S3

Elastic Load Balancing (ELB)

Amazon EBS

Amazon RDS

- Incident Response

IAM

AWS CloudFormation

Amazon CloudWatch Events

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

3) Reliability

- Ability of a system to recover from infrastructure or service disruptions, dynamically acquire computing resources to meet demand, and mitigate disruptions such as misconfigurations or transient network issues
- Design Principles
  - **Test recovery procedures -** Use automation to simulate different failures or to recreate scenarios that led to failures before
  - **Automatically recover from failure -** Anticipate and remediate failures before they occur
  - **Scale horizontally to increase aggregate system availability -** Distribute requests across multiple, smaller resources to ensure that they don't share a common point of failure
  - **Stop guessing capacity -** Maintain the optimal level to satisfy demand without over or under provisioning - Use Auto Scaling
  - **Manage change in automation -** Use automation to make changes to infrastructure

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Reliability
AWS Services

- Foundations

IAM

Amazon VPC

Service Limits

AWS Trusted Advisor

- Change Management

AWS Auto Scaling

Amazon CloudWatch

AWS CloudTrail

AWS Config

- Failure Management

Backups

AWS CloudFormation

Amazon S3

Amazon S3 Glacier

Amazon Route 53

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

4) Performance Efficiency

- Includes the ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve
- Design Principles
  - **Democratize advanced technologies** - Advance technologies become services and hence you can focus more on product development
  - **Go global in minutes** - Easy deployment in multiple regions
  - **Use serverless architectures** - Avoid burden of managing servers
  - **Experiment more often** - Easy to carry out comparative testing
  - **Mechanical sympathy** - Be aware of all AWS services

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Performance Efficiency AWS Services

- Selection
- Review
- Monitoring
- Tradeoffs

AWS Auto Scaling

AWS Lambda

Amazon Elastic Block Store (EBS)

Amazon Simple Storage Service (S3)

Amazon RDS

AWS CloudFormation

AWS News Blog

Amazon CloudWatch

AWS Lambda

Amazon RDS

Amazon ElastiCache

AWS Snowball

Amazon CloudFront

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

5) Cost Optimization

- Includes the ability to run systems to deliver business value at the lowest price point
- Design Principles
  - **Adopt a consumption mode** - Pay only for what you use
  - **Measure overall efficiency** - Use CloudWatch
  - **Stop spending money on data center operations** - AWS does the infrastructure part and enables customer to focus on organization projects
  - **Analyze and attribute expenditure** - Accurate identification of system usage and costs, helps measure return on investment (ROI) - Make sure to use tags
  - **Use managed and application level services to reduce cost of ownership** - As managed services operate at cloud scale, they can offer a lower cost per transaction or service

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Cost Optimization AWS Services

- Expenditure Awareness

AWS Budgets

AWS Cost and Usage Report

AWS Cost Explorer

Reserved Instance Reporting

- Cost-Effective Resources

Spot instance

Reserved instance

Amazon S3 Glacier

- Matching supply and demand

AWS Auto Scaling

AWS Lambda

- Optimizing Over Time

AWS Trusted Advisor

AWS Cost and Usage Report

AWS News Blog

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Trusted Advisor

- No need to install anything – high level AWS account assessment
- Analyze your AWS accounts and provides recommendation:
  - Cost Optimization
  - Performance
  - Security
  - Fault Tolerance
  - Service Limits
- Core Checks and recommendations – all customers
- Can enable weekly email notification from the console
- Full Trusted Advisor – Available for Business & Enterprise support plans
  - Ability to set CloudWatch alarms when reaching limits

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

© Stephane Maarek

More Architecture Examples

- We've explored the most important architectural patterns:
  - **Classic:** EC2, ELB, RDS, ElastiCache, etc…
  - **Serverless:** S3, Lambda, DynamoDB, CloudFront, API Gateway, etc…

- If you want to see more AWS architectures:
- https://aws.amazon.com/architecture/
- https://aws.amazon.com/solutions/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Exam Review & Tips

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

State of learning checkpoint

- Let's look how far we've gone on our learning journey

- https://aws.amazon.com/certification/certified-solutions-architect-associate/

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Practice makes perfect

- If you're new to AWS, take a bit of AWS practice thanks to this course before rushing to the exam
- The exam recommends you to have one or more years of hands-on experience on AWS
- Practice makes perfect!

- If you feel overwhelmed by the amount of knowledge you just learned, just go through it one more time

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Proceed by elimination

- Most questions are going to be scenario based
- For all the questions, rule out answers that you know for sure are wrong
- For the remaining answers, understand which one makes the most sense

- There are very few trick questions
- Don't over-think it
- If a solution seems feasible but highly complicated, it's probably wrong

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Skim the AWS Whitepapers

- You can read about some AWS White Papers here:
  - Architecting for the Cloud: AWS Best Practices
  - AWS Well-Architected Framework
  - AWS Disaster Recovery (https://aws.amazon.com/disaster-recovery/)

- Overall we've explored all the most important concepts in the course
- It's never bad to have a look at the whitepapers you think are interesting!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Read each service's FAQ

- FAQ = Frequently asked questions
- Example: https://aws.amazon.com/vpc/faqs/

- FAQ cover a lot of the questions asked at the exam
- They help confirm your understanding of a service

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Get into the AWS Community

- Help out and discuss with other people in the course Q&A
- Review questions asked by other people in the Q&A
- Do the practice test in this section

- Read forums online
- Read online blogs
- Attend local meetups and discuss with other AWS engineers
- Watch re-invent videos on Youtube (AWS Conference)

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

How will the exam work?

- You'll have to register online at https://www.aws.training/
- Fee for the exam is 150 USD
- Provide two identity documents (ID, Credit Card, details are in emails sent to you)
- No notes are allowed, no pen is allowed, no speaking
- 65 questions will be asked in 130 minutes
- At the end you can optionally review all the questions / answers

- You will know right away if you passed / failed the exams
- You will not know which answers were right / wrong
- You will know the overall score a few days later (email notification)
- To pass you need a score of a least 720 out of 1000
- If you fail, you can retake the exam again 14 days later

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Congratulations!

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com

Congratulations!

- Congrats on finishing the course!
- I hope you will pass the exam without a hitch ☺

- If you haven't done so yet, I'd love a review from you!
- If you passed, I'll be more than happy to know I've helped
  - Post it in the Q&A to help & motivate other students. Share your tips!
  - Post it on LinkedIn and tag me!

- Overall, I hope you learned how to use AWS and that you will be a tremendously good AWS Solutions Architect

© Stephane Maarek

NOT FOR DISTRIBUTION © Stephane Maarek www.datacumulus.com