Notes

Customer Examples for Macie Custom Data Identifiers
@https://aws.amazon.com/blogs/security/discover-sensitive-data-by-using-custom-data-identifiers-with-amazon-macie/

Macie FAQ
@https://w.amazon.com/bin/view/Amazon_Macie_Frequently_Asked_Questions

Macie Configuring Replication
@https://docs.aws.amazon.com/AmazonS3/latest/dev/replication-walkthrough-2.html

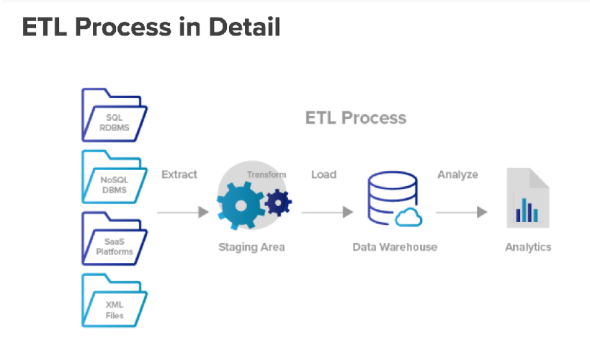
What is a Data Lake?
https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/
Services in AWS that creates Data Lake?separate permissioning
DL allows you to have unlimited data warehouse. easier for you to streamline where the data is going.
Data Sync: Lake is data goes to / Faucet is sources of data, manual processing batch uploads from. Could be chat history.. random kinds of data.

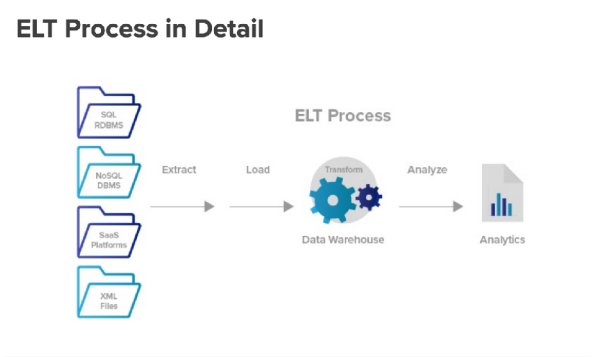
Upload to S3
Have that kick off a Macie Job. Triggered to run. *How to trigger services from S3.
Data is scanned and flagged.
Triggers an event bridge to be written to DynamoDB
Encrypt from reading it in S3, then encrypting it client side in DynamoDB

ETL vs. ELT

The five critical differences of ETL vs ELT:

1. ETL is the Extract, Transform, and Load process for data. ELT is Extract, Load, and Transform process for data.
2. In ETL, data moves from the data source to staging into the data warehouse.
3. ELT leverages the data warehouse to do basic transformations. There is no need for data staging.
4. ETL can help with data privacy and compliance by cleaning sensitive and secure data even before loading into the data warehouse.
5. ETL can perform sophisticated data transformations and can be more cost-effective than ELT.





Data transformation is the process where you extract data, sift through data, understand the data, and then transform it into something you can analyze. T

write Lambda to DynamoDB
https://sysadmins.co.za/interfacing-amazon-dynamodb-with-python-using-boto3/

AD examples
https://creately.com/blog/diagrams/aws-templates-for-architecture-diagrams/

Event Patterns
https://docs.aws.amazon.com/eventbridge/latest/userguide/filtering-examples-structure.html

Macie UserGuide
https://docs.aws.amazon.com/macie/latest/user/macie-user-guide.pdf#%5B%7B%22num%22%3A553%2C%22gen%22%3A0%7D%2C%7B%22name%22%3A%22XYZ%22%7D%2C72%2C503.447%2Cnull%5D

acloud.guru

so acloud.guru

log in with your @amazon email, and your Midway token (I believe)

and then take the aws certified Solutions architect course

it's very good

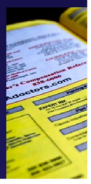
and has hands-on labs you can do at your own pace

IAM 101

IAM allows you to manage users and their level of access to the AWS Console.

Identity Access Management (IAM) offers the following features;

- Centralised control of your AWS account
- Shared Access to your AWS account
- Granular Permissions
- Identity Federation (including Active Directory, Facebook, Linkedin etc)
- Multifactor Authentication
- Provide temporary access for users/devices and services where necessary
- Allows you to set up your own password rotation policy
- Integrates with many different AWS services
- Supports PCI DSS Compliance



1 **Users**
End Users such as people, employees of an organization, etc.

2 **Groups**
A collection of users. Each user in the group will inherit the permissions of the group.

3 **Policies**
Policies are made up of documents, called Policy documents. These documents are in a format called JSON and they give permissions as to what a User/Group/Role is able to do.

4 **Roles**
You create roles and then assign them to AWS Resources.

What Have We Learnt So Far?

- **IAM is universal**. It does not apply to regions at this time.
- The **"root account"** is simply the account created when first setup your AWS account. It has complete Admin access.
- New Users have **NO permissions** when first created.
- New Users are assigned **Access Key ID** & **Secret Access Keys** when first created.
- **These are not the same as a password.** You cannot use the Access key ID & Secret Access Key to Login in to the console. You can use this to access AWS via the APIs and Command Line, however.
- **You only get to view these once.** If you lose them, you have to regenerate them. So, save them in a secure location.

S3 101

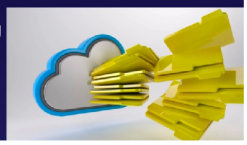
S3 provides developers and IT teams with secure, durable, highly-scalable object storage. Amazon S3 is easy to use, with a simple web services interface to store and retrieve any amount of data from anywhere on the web.

The basics of S3 are as follows;

- **S3 is a universal namespace**. That is, names must be unique globally.
- https://acloudguru.s3.amazonaws.com/
- https://acloudguru.eu-west-1.amazonaws.com/
- When you upload a file to S3, you will receive a
- **HTTP 200 code** if the upload was successful.

S3 is Object based. **Think of Objects just as files.**
Objects consist of the following:

- Key (This is simply the name of the object)
- Value (This is simply the data and is made up of a sequence of bytes).
- Version ID (Important for versioning)
- Metadata (Data about data you are storing
- Subresources;
  - Access Control Lists
  - Torrent



1

S3 Standard

99.99% availability
99.999999999% durability, stored redundantly across multiple devices in multiple facilities, and is designed to sustain the loss of 2 facilities concurrently.

2

S3 - IA

(Infrequently Accessed): For data that is accessed less frequently, but requires rapid access when needed. Lower fee than S3, but you are charged a retrieval fee.

3

S3 One Zone - IA

For where you want a lower-cost option for infrequently accessed data, but do not require the multiple Availability Zone data resilience.

4

S3 - Intelligent Tiering

Designed to optimize costs by automatically moving data to the most cost-effective access tier, without performance impact or operational overhead.

5

S3 Glacier

S3 Glacier is a secure, durable, and low-cost storage class for data archiving. You can reliably store any amount of data at costs that are competitive with or cheaper than on-premises solutions. Retrieval times configurable from minutes to hours.

6

S3 Glacier Deep Archive

S3 Glacier Deep Archive is Amazon S3's lowest-cost storage class where a retrieval time of 12 hours is acceptable.

	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%	N/A	N/A
Availability SLA	99.9%	99%	99%	99%	N/A	N/A
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
First byte latency	milliseconds	millseconds	milliseconds	milliseconds	select minutes or hours	select hours

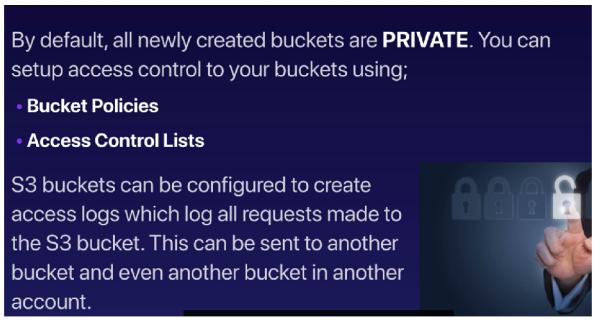
- Remember that S3 is **Object-based**: i.e. allows you to upload files.
- Files can be from 0 Bytes to 5 TB.
- There is unlimited storage.
- Files are stored in Buckets.
- **S3 is a universal namespace**. That is, names must be unique globally.
- https://acloudguru.s3.amazonaws.com/
- https://acloudguru.eu-west-1.amazonaws.com/

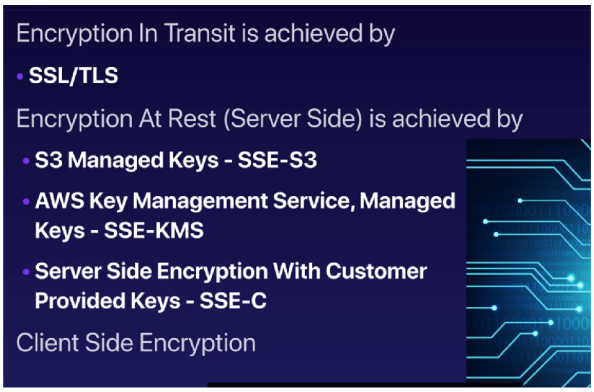
- **Not suitable to install an operating system on.**
- Successful uploads will generate a **HTTP 200** status code.
- You can turn on **MFA Delete**

The Key Fundamentals of S3 Are;

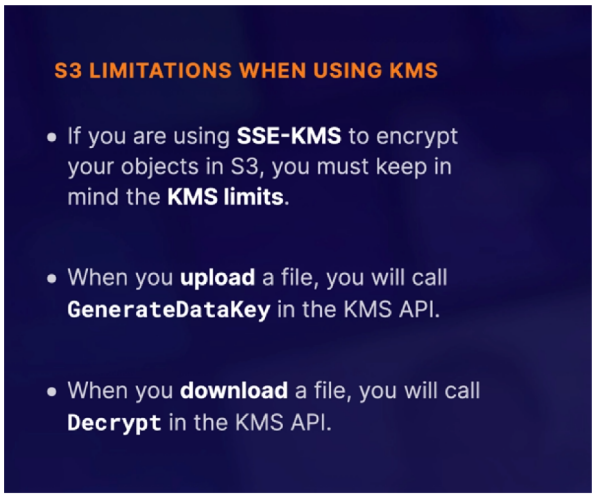
- Key (This is simply the name of the object)
- Value (This is simply the data and is made up of a sequence of bytes).
- Version ID (Important for versioning)
- Metadata (Data about data you are storing
- Subresources;
  - Access Control Lists
  - Torrent

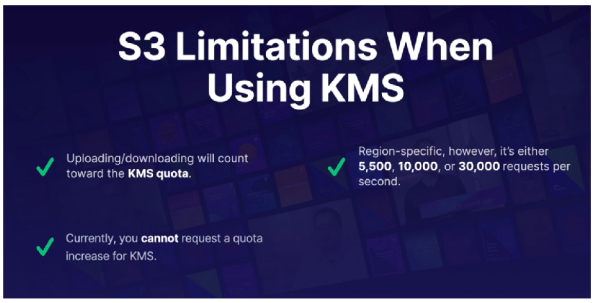
S3 Security and Encryption

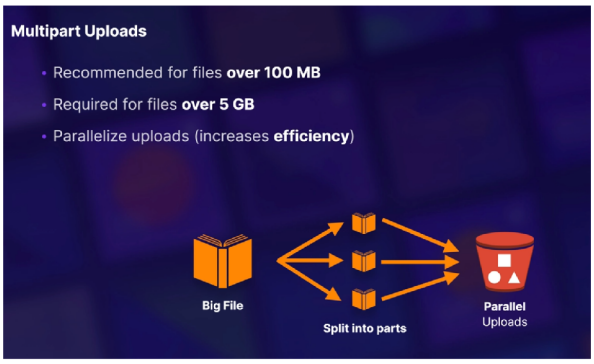




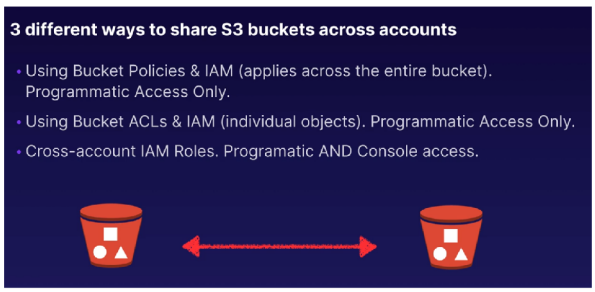
S3 Performance







Sharing S3 Buckets Across Accounts



S3 Transfer Acceleration

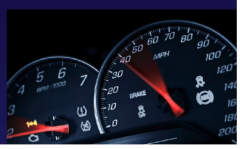
S3 Transfer Acceleration utilises the CloudFront Edge Network to accelerate your uploads to S3. Instead of uploading directly to your S3 bucket, you can use a distinct URL to upload directly to an edge location which will then transfer that file to S3. You will get a distinct URL to upload to:

Cloudfront

A content delivery network (CDN) is a system of distributed servers (network) that deliver webpages and other web content to a user based on the geographic locations of the user, the origin of the webpage, and a content delivery server.

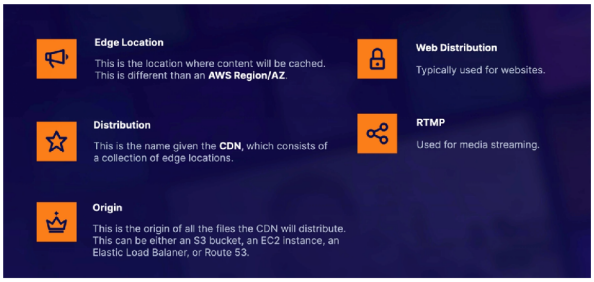
- **Edge Location** - This is the location where content will be cached. This is separate to an AWS Region/AZ.
- **Origin** - This is the origin of all the files that the CDN will distribute. This can be an S3 Bucket, an EC2 Instance, an Elastic Load Balancer, or Route53.
- **Distribution** - This is the name given the CDN which consists of a collection of Edge Locations.

Amazon CloudFront can be used to deliver your entire website, including dynamic, static, streaming, and interactive content using a global network of edge locations. Requests for your content are automatically routed to the nearest edge location, so content is delivered with the best possible performance.



- Web Distribution - Typically used for Websites.
- RTMP - Used for Media Streaming.

- Edge locations are not just READ only — you can write to them too. (ie put an object on to them.)
- Objects are cached for the life of the **TTL (Time To Live.)**
- You can clear cached objects, but you will be charged.



Create a CloudFront Signed URLs and Cookies

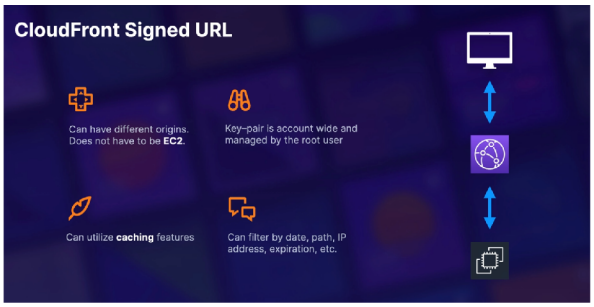
Use CloudFront Signed URLS or Signed Cookies

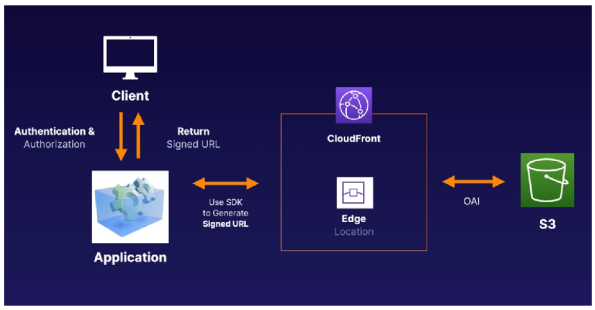
1. A signed URL is for individual files. **1 file = 1 URL.**
2. A signed cookie is for multiple files. **1 cookie = multiple files.**

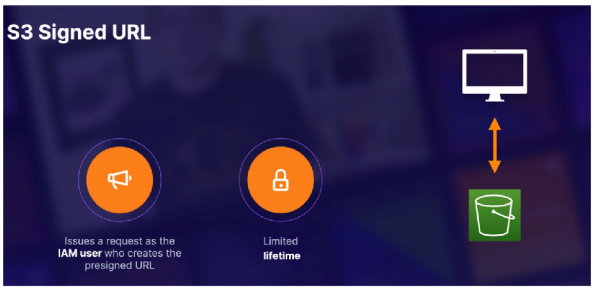
When we create a signed URL or signed cookie, we attach a policy.

The policy can include:

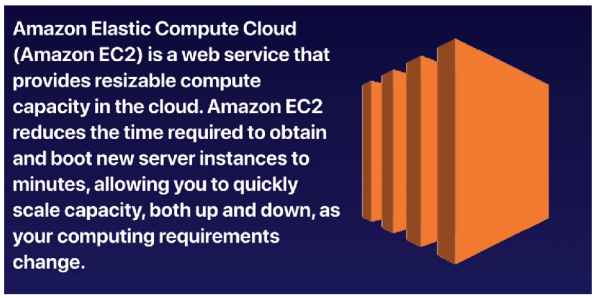
- URL expiration
- IP ranges
- Trusted signers (which AWS accounts can create **signed URLS**)

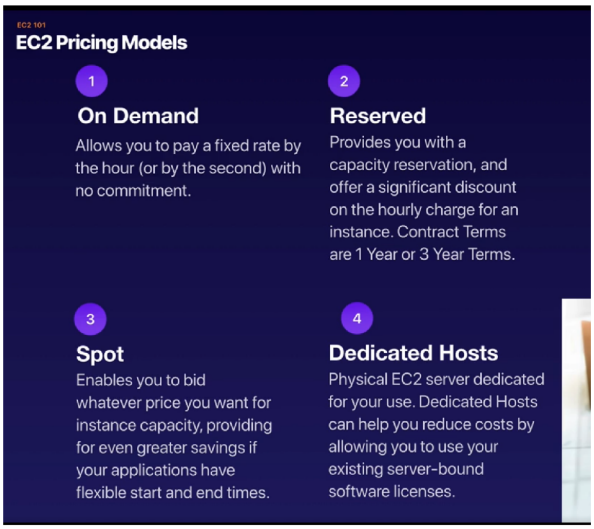


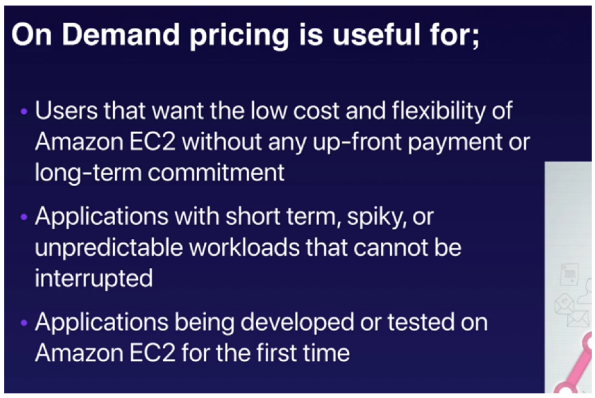




EC2 101 Elastic Compute Cloud







Lambda

Lambda Is The Ultimate Abstraction Layer;

- Data Centres
- Hardware
- Assembly Code/Protocols
- High Level Languages
- Operating Systems
- Application Layer/AWS APIs
- AWS Lambda

You can use Lambda in the following ways;

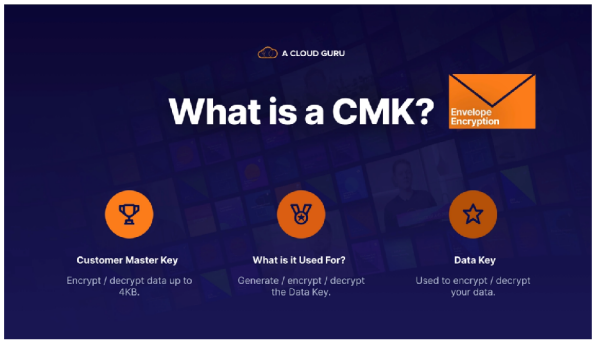
- As an event-driven compute service where AWS Lambda runs your code in response to events. These events could be changes to data in an Amazon S3 bucket or an Amazon DynamoDB table.
- As a compute service to run your code in response to HTTP requests using Amazon API Gateway or API calls made using AWS SDKs. This is what we use at A Cloud Guru.

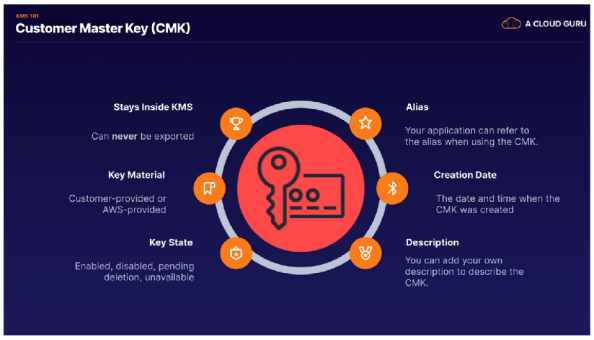
AWS Lambda is a compute service where you can upload your code and create a Lambda function. AWS Lambda takes care of provisioning and managing the servers that you use to run the code. You don't have to worry about operating systems, patching, scaling, etc.

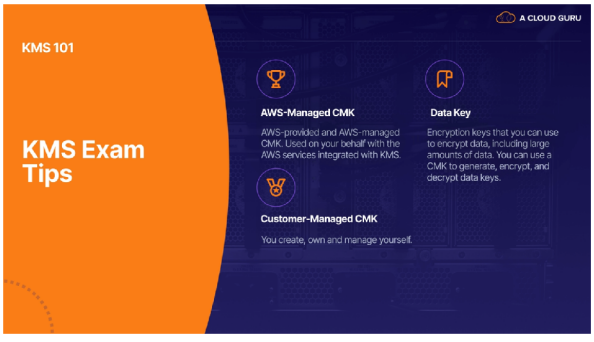
Lambda Exam Tips

- Lambda scales out (not up) automatically
- Lambda functions are independent, 1 event = 1 function
- Lambda is serverless
- Know what services are serverless!
- Lambda functions can trigger other lambda functions, 1 event can = x functions if functions trigger other functions

KMS 101







@https://docs.aws.amazon.com/kms/latest/developerguide/symm-asymm-choose.html#key-spec-rsa

Accolades

https://accolades.corp.amazon.com/

Architecture Accelerating

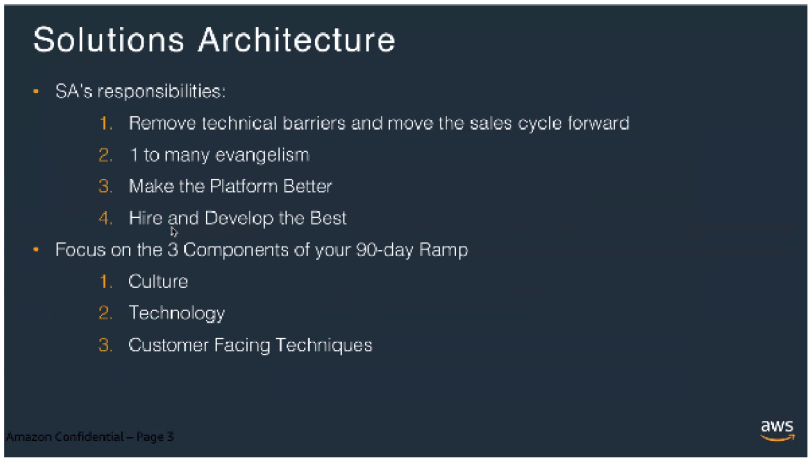
https://kiku.aws.training/SessionSearch?pageNumber=1&courseId=37002&trainingProviderId=1

AWS Cloud Economics (3.5 hrs)

https://kiku.aws.training/Details/eLearning?id=36904

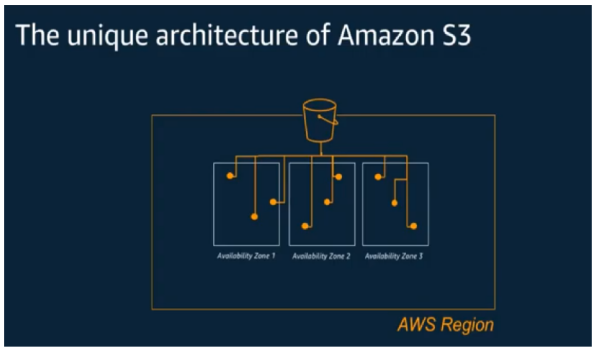
AWS Technical Professional (3 hrs)

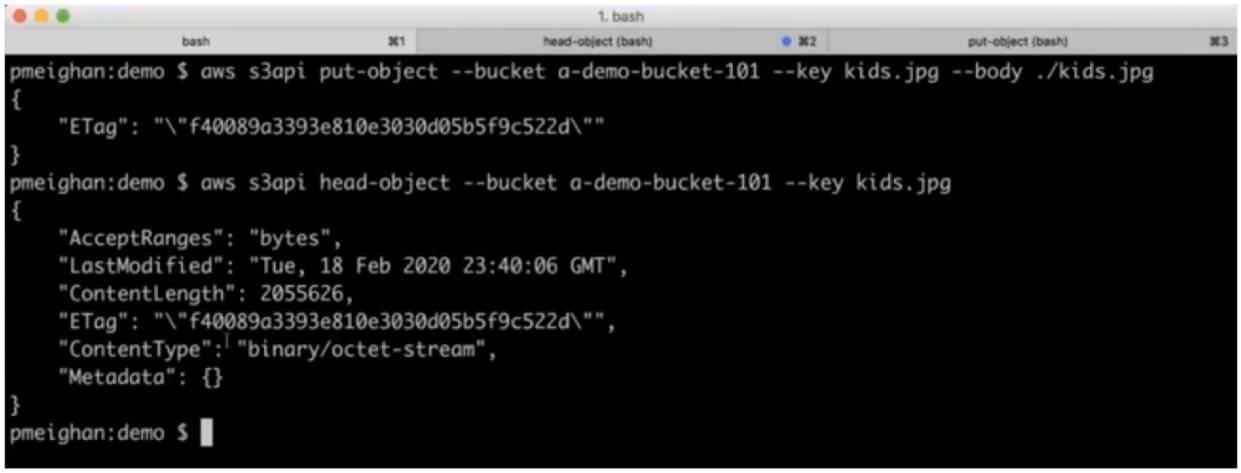
https://kiku.aws.training/Details/Curriculum?id=45423



Ramp up

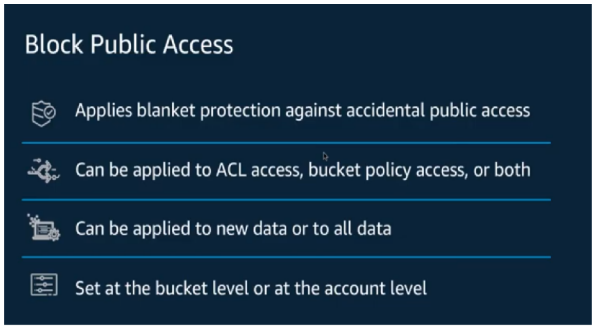
- Getting started with S3: https://www.youtube.com/watch?v=vFfY_-TL-pc

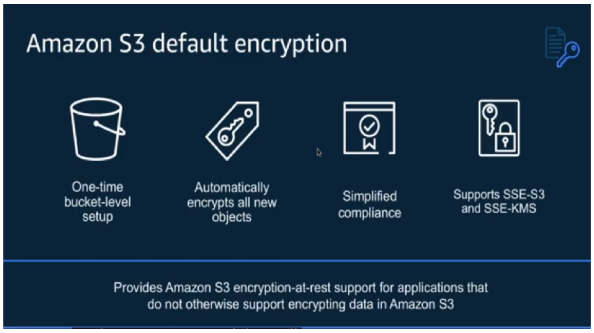



1. bash
bash ⌘1
head-object (bash) ⌘2
put-object (bash) ⌘3
pmeighan:demo $ aws s3api put-object --bucket a-demo-bucket-101 --key kids.jpg --body ./kids.jpg
{
"ETag": "\"f40089a3393e810e3030d05b5f9c522d\""
}
pmeighan:demo $ aws s3api head-object --bucket a-demo-bucket-101 --key kids.jpg
{
"AcceptRanges": "bytes",
"LastModified": "Tue, 18 Feb 2020 23:40:06 GMT",
"ContentLength": 2055626,
"ETag": "\"f40089a3393e810e3030d05b5f9c522d\"",
"ContentType": "binary/octet-stream",
"Metadata": {}
}
pmeighan:demo $


Object tags work with lifecycle policies
Perform automated actions on a subset of your data with object tags
Lifecycle
Specify a tag filter to transition or expire objects
Use S3 batch operations to apply object tags at scale
Ex: Transition all objects tagged "Project : Delta" to S3 Glacier


Lifecycle policies use rules to manage your storage
Use lifecycle policies to transition objects to another storage class
Lifecycle rules take action based on object age. Here's an example:
1. Move objects older than 30 days to S3 Standard — Infrequent Access
2. Move objects older than 365 days to S3 Glacier Deep Archive
S3 Standard
S3 Standard — Infrequent Access
S3 Glacier Deep Archive


Block Public Access
Applies blanket protection against accidental public access
Can be applied to ACL access, bucket policy access, or both
Can be applied to new data or to all data
Set at the bucket level or at the account level


Amazon S3 default encryption
One-time bucket-level setup
Automatically encrypts all new objects
Simplified compliance
Supports SSE-S3 and SSE-KMS
Provides Amazon S3 encryption-at-rest support for applications that do not otherwise support encrypting data in Amazon S3

Amazon S3 inventory

A managed alternative to using the LIST API

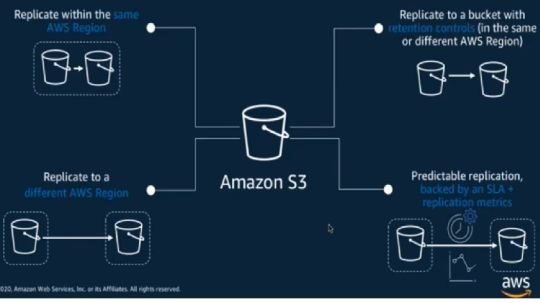
Regularly generates a list of objects for analytics and auditing.

- Encryption status
- Storage class
- Creation date
- Replication status
- Object size, and more
- S3 Intelligent-Tiering access tier new!

S3 Data Protection capabilities

GOAL	AMAMZON S3 FEATURES
Replicate data for compliance and bad actor protection	Use S3 Replication with Replication Time Control and ownership override
Protect data from accidental deletes	Use bucket versioning while reducing cost with Lifecycle policies
Protect data for governance and compliance purposes	Use S3 Object Lock to store objects as write-once-read-many (WORM)

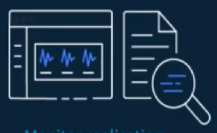
S3 replication



Amazon S3 Replication time control new!

Designed to replicate 99.99% of objects within 15 minutes

15 minute replication time backed by an AWS Service Level Agreement (SLA)



Monitor replication using Amazon CloudWatch metrics and event notifications

Amazon S3 Replication time control

Designed to replicate 99.99% of objects within 15 minutes

Monitor your replication with 3 new CloudWatch metrics

Optional: Set up alarms on your metrics

Replication latency
The maximum number of seconds by which the destination region is behind the source region for a given replication rule

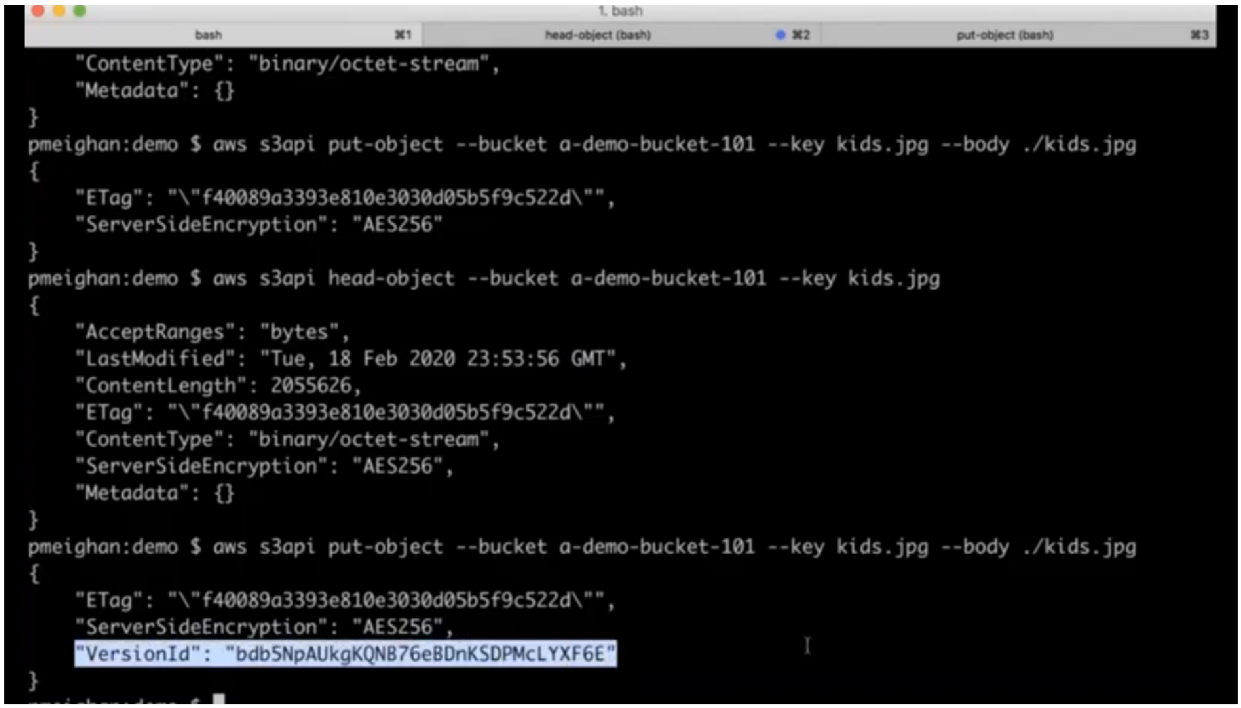
Bytes pending replication
The total number of bytes of objects pending replication for a given replication rule

Operations pending replication
The number of operations pending replication for a given replication rule

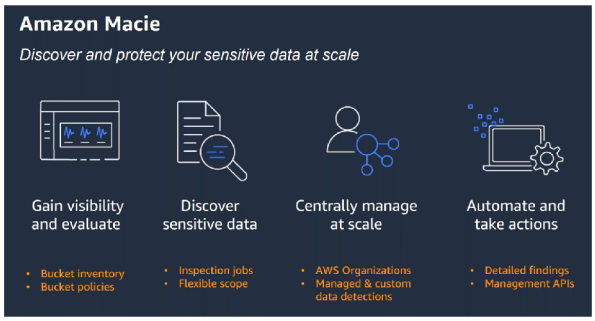
Amazon S3 Object Lock

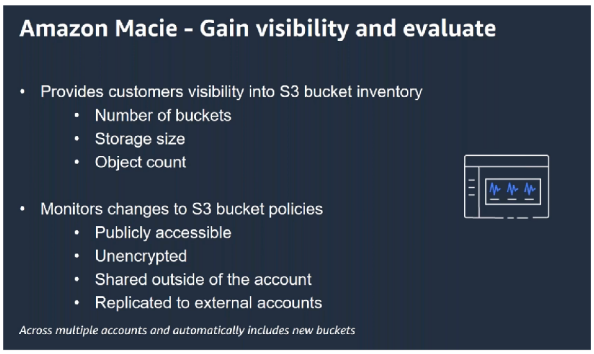
Use Object Lock to store objects as write-once-read-many (WORM)

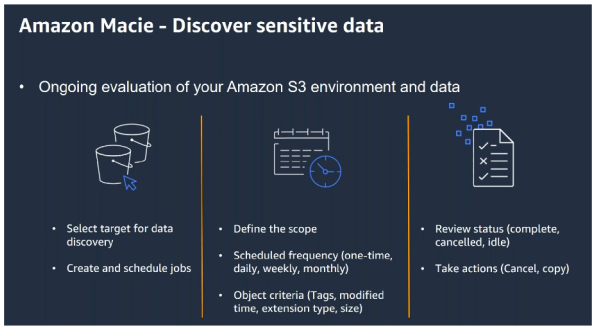
Compliance mode	Governance mode	Legal hold
Store compliant data	Store data in WORM format; privileged users can modify retention controls	If you're unsure how long you want your objects to stay immutable

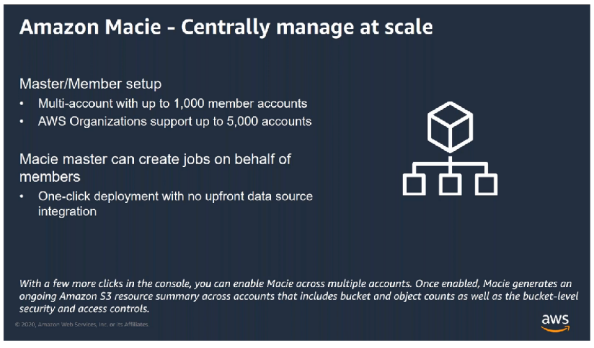


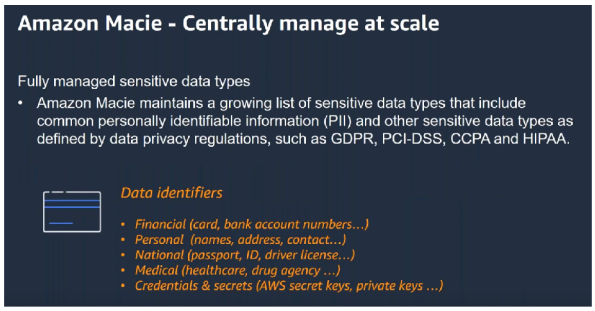
- Introducing the New Macie: https://www.youtube.com/watch?v=I-ewoQekdXE (the Macie service you are using is new, we had an older version which is now called Macie Classic)

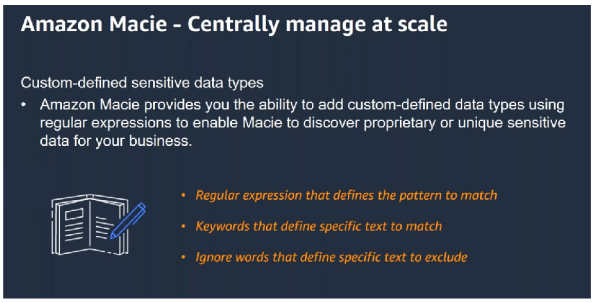


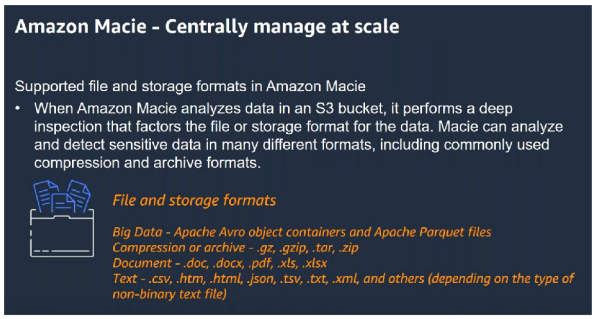


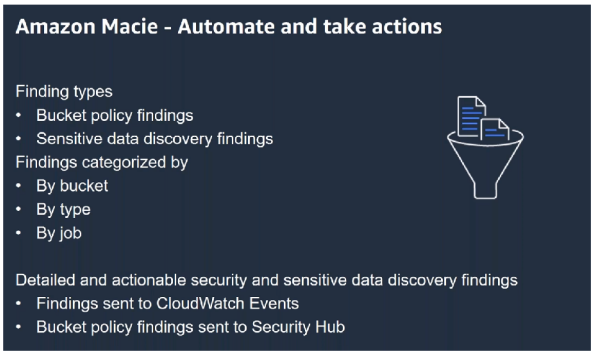


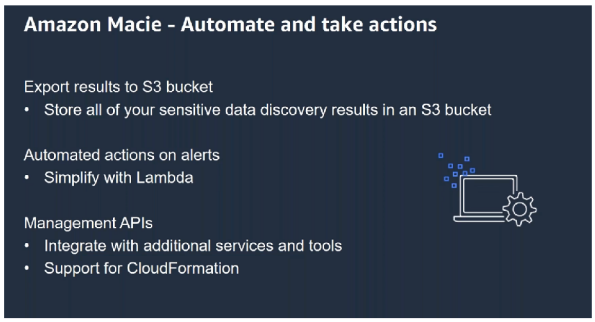












NOTES:

- Built-in Data Identifier, Custom-Defined Sensitive Data Types
- Classification Findings
- Detailed Discovery Reports to Separate S3 bucket for customers to confogure
- Delegated Administrator

- Getting started with Identity & Access Management (IAM): https://www.youtube.com/watch?v=Zvz-qYYhvMk
- Using KMS for data protection: https://www.youtube.com/watch?v=hxWvbNvj2Ig
- Introduction to Cloud9: https://pages.awscloud.com/Introducing-AWS-Cloud9_0106-DEV_OD.html
- Attached sample encryption requirements doc