WEBVTT

00:00:00.000 --> 00:00:07.360

Добрый день, уважаемые студенты. Мы начинаем с вами разбор вопроса Practice Test-а.

00:00:07.360 --> 00:00:15.600

Согласно вопросу, компания генерирует некоторое количество конфиденциальных данных, которые хранит на локальном дата-центре.

00:00:15.600 --> 00:00:25.120

Как бэкап решение, компания хочет также хранить эти данные в Amazon S3 Bucket.

00:00:25.120 --> 00:00:38.200

Согласно некоторым требованиям безопасности внутри компании, шифрование должно происходить перед тем, как данные будут отправлены в S3.

00:00:38.200 --> 00:00:47.320

Более того, компания должна самостоятельно заниматься управлением и обновлением ключей шифрования.

00:00:47.320 --> 00:00:56.440

Также эти ключи должны быть под контролем компании.

00:00:56.440 --> 00:01:03.040

Какие из предложенных методов помогают этого достичь?

00:01:03.040 --> 00:01:06.160

На этом слайде вы видите предложенные варианты ответов.

00:01:06.160 --> 00:01:09.280

Также видите, что 3 и 5 являются верными.

00:01:09.280 --> 00:01:12.800

Перед тем, как мы начнем разбирать вопросы, давайте вспомним теорию.

00:01:12.800 --> 00:01:22.640

Зная теорию, легче будет ответить на любой вопрос, связанный с шифрованием в S3.

00:01:22.640 --> 00:01:27.520

Как мы знаем, есть два различных вида шифрования.

00:01:27.520 --> 00:01:36.640

Это шифрование in-transit, когда данные из одного места передаются в другое, и at rest, когда данные хранятся.

00:01:36.640 --> 00:01:47.960

Если мы говорим касательно S3, in-transit подразумевается процесс передачи в и загрузки данных из S3.

00:01:47.960 --> 00:02:00.680

В случае, когда мы говорим at rest, имеется в виду шифрование, когда данные фактически находятся на S3 в дата-центрах AWS.

00:02:00.680 --> 00:02:12.440

Когда мы говорим шифрование in-transit, подразумевается использование SSL ключа или шифрование на стороне клиента.

00:02:12.440 --> 00:02:26.800

Самый популярный и явный пример это в браузере, когда вы открываете некоторый сайт, если видите, что подключение идет через HTTPS, то это защищенное подключение.

00:02:26.800 --> 00:02:41.400

Так вот, в этом случае был подключен специальный SSL сертификат, который помогает шифровать передачу данных от вас до веб-сайта, то есть in-transit.

00:02:41.400 --> 00:02:55.560

Когда мы говорим at rest касательно сервиса S3, есть два принципиально отличающихся группы, это Server-Side Encryption и Client-Side Encryption.

00:02:55.560 --> 00:03:09.200

Когда мы говорим Server-Side Encryption, идея в том, что мы передаем данные к S3, далее S3 шифрует данные у себя на стороне и сохраняет на жесткие диски.

00:03:09.200 --> 00:03:34.440

Когда мы говорим Client-Side Encryption, подразумевается, что данные перед передачей в S3 уже были зашифрованы некоторыми ключами, и S3 уже получает зашифрованный файл, ему нет смысла еще раз его шифровать, а оно в таком же виде сохраняет этот файл на жесткие диски.

00:03:34.440 --> 00:03:53.720

Если говорить про обратный процесс, в случае Server-Side Encryption, S3 извлекает файлы из жестких дисков, у себя на стороне, используя ключи шифрования, расшифровывает эти файлы и передает эти данные вам.

00:03:53.720 --> 00:04:09.160

Если же мы говорим Client-Side Encryption, при загрузке файла из S3 происходит следующее: S3 извлекает данные из жестких дисков, и не расшифровывая ее, передает в таком же виде вам.

00:04:09.160 --> 00:04:21.640

Вы уже получаете зашифрованный файл в таком же виде, как вы передавали ее ранее, и используя ключи шифрования, этот файл расшифровываете и дальше используете.

00:04:21.640 --> 00:04:25.120

Сам процесс расшифровки происходит уже на вашей стороне.

00:04:25.120 --> 00:04:46.600

Таким образом мы говорим правильными ответами является использование Client-Side Encryption с сервисом AWS KMS, и второй вариант также Client-Side Encryption, в котором используется мастер ключ, хранящийся на стороне клиента, то есть на нашей стороне.

00:04:46.600 --> 00:05:03.080

Если же мы говорим про оставшиеся варианты, они неверные, так как есть Server-Side Encryption в обоих неправильных вариантах, что противоречит самому требованию вопроса.

00:05:03.080 --> 00:05:10.760

И также есть еще вариант Set-up Client-Side encryption with Amazon S3 managed encryption keys.

00:05:10.760 --> 00:05:13.120

Этот вариант теоретически невозможен.

00:05:13.120 --> 00:05:34.360

Здесь имеется в виду managed encryption key, который недоступен для нас как для клиента, он управляется S3, используется для шифрования наших данных, и мы до него добраться, как-то менять его, обновлять его не можем.

00:05:34.360 --> 00:05:49.440

И когда говорим Client-Side encryption with S3 managed encryption key, она сама по себе нелогичная, то есть мы не добравшись, нет возможности эти ключи извлечь, чтобы зашифровать, а здесь говорится Client-Side encryption.

00:05:49.440 --> 00:05:52.040

Поэтому этот вариант сам по себе является некорректным.

00:05:52.040 --> 00:05:57.440

Таким образом мы разобрали вопрос Practice Test-а.

00:05:57.440 --> 00:06:00.440

Спасибо за внимание.