WEBVTT

00:00:00.000 --> 00:00:02.640

Добрый день, уважаемые студенты.

00:00:02.640 --> 00:00:07.480

Мы с вами начинаем разбор вопроса Practice Test-а.

00:00:07.480 --> 00:00:11.840

Согласно вопросу, у компании во владении несколько AWS аккаунтов.

00:00:11.840 --> 00:00:15.840

Они все подключены к AWS Organizations.

00:00:15.840 --> 00:00:27.640

Вам необходимо скопировать несколько S3 объектов в другой S3 bucket, который находится на другом AWS аккаунте.

00:00:27.640 --> 00:00:31.240

Оба аккаунта находятся в вашем владении.

00:00:31.240 --> 00:00:38.120

Вам как Solutions Architect было дано задание настроить все необходимые права доступа, чтобы выполнить это задание.

00:00:38.120 --> 00:00:49.800

Также вам необходимо обеспечить следующие скопированные объекты.

00:00:49.800 --> 00:01:02.440

Они должны быть доступны для пользователей в Destination аккаунте, а для Source аккаунта должны быть недоступны.

00:01:02.440 --> 00:01:10.640

Нам необходимо выбрать вариант ответа, который помогает эту задачу выполнить.

00:01:10.640 --> 00:01:16.920

На этом слайде вы видите варианты ответов.

00:01:16.920 --> 00:01:20.360

Также видите, что первый вариант является верным.

00:01:20.360 --> 00:01:32.320

Я бы здесь хотел еще раз отметить, что с теми знаниями, которые у вас есть по AWS, вы с легкостью можете исключать один или несколько вариантов ответа, заведомо ложных.

00:01:32.320 --> 00:01:38.680

Давайте попробуем поступить следующим образом, а потом после разберем верный вариант ответа.

00:01:38.680 --> 00:01:53.000

Если мы говорим про второй вариант, здесь предлагается использовать функционал Requester Pays для того, чтобы получить некоторые скидки или выплаты и ни слова не говорится про permission.

00:01:53.000 --> 00:01:57.480

Исходя из этого, мы можем исключить этот вариант ответа.

00:01:57.480 --> 00:02:17.120

Если мы говорим про третий вариант ответа, здесь предлагается использовать сервис Amazon WorkDocs и далее произвести настройки до S3 bucket-ов, сделать файлы общими и таким образом скопировать на destination bucket.

00:02:17.120 --> 00:02:25.640

Это не сработает по той простой причине, так как WorkDocs предоставляется по модели SaaS, то есть Software As A Service.

00:02:25.640 --> 00:02:35.560

Идея в том, что это готовый сервис, где вы можете менять его минимально, только некоторые user-based настройки.

00:02:35.560 --> 00:02:43.600

А производить интеграцию с S3 невозможно, поэтому этот вариант мы тоже можем исключить.

00:02:43.600 --> 00:02:48.400

Если мы говорим про четвертый вариант ответа, то здесь говорится про CORS.

00:02:48.400 --> 00:02:56.040

Я напомню, что CORS это термин, не относящийся напрямую к AWS, не относящийся к облачным технологиям.

00:02:56.040 --> 00:03:00.760

CORS пришел к нам из веб-разработки, там часто используется.

00:03:00.760 --> 00:03:06.640

И здесь упоминается, так как у S3 есть возможность настраивать CORS policy.

00:03:06.640 --> 00:03:09.640

Что такое CORS policy, вкратце напомню.

00:03:09.640 --> 00:03:27.120

Это некоторый документ, который прописывается в первом веб-сайте для того, чтобы из второго веб-сайта можно было использовать данные из первого веб-сайта.

00:03:27.120 --> 00:03:36.920

То есть CORS он как бы прописывает те веб-сайты, которые могут обращаться к вашему веб-сайту.

00:03:36.920 --> 00:03:51.880

Так как здесь идет больше речь про веб-сайты, а в вопросе у нас речь идет про S3 bucket и передачу данных в рамках глобальной инфраструктуры AWS, при этом не выходя в интернет, то этот вариант мы тоже можем исключить.

00:03:51.880 --> 00:03:58.680

Остается первый вариант, который является верным.

00:03:58.680 --> 00:04:04.080

Давайте теперь остановимся на верном варианте ответа и почему оно таковым является.

00:04:04.080 --> 00:04:08.840

Здесь нам предлагается настроить policy для source bucket.

00:04:08.840 --> 00:04:19.800

Оно разрешает второму аккаунту получать объекты из этого bucket.

00:04:19.800 --> 00:04:45.040

Далее нам необходимо на стороне второго аккаунта настроить пользователя либо роль, у которого есть возможность получать данные из source bucket и загружать файлы в destination bucket, который находится в том же AWS аккаунте, что и этот user или роль.

00:04:45.040 --> 00:05:04.520

После того как мы это все настроим, мы фактически сидя в destination аккаунте, cross-account запросим нужные нам объекты и в свой родной аккаунт положим наши объекты.

00:05:04.520 --> 00:05:10.000

Почему важно, чтобы роль или пользователь был в destination bucket?

00:05:10.000 --> 00:05:23.320

Идея в том, что аккаунтом-владельцем файлов при записи в s3 bucket является тот аккаунт, который записывает.

00:05:23.320 --> 00:05:44.640

Если мы это делаем cross-account, то при записи из аккаунта A, что является source, запишем в аккаунт B, cross-account, то на стороне destination bucket владельцем файлов внутри этого bucket будет другой аккаунт.

00:05:44.640 --> 00:05:57.480

И даже если это ваш аккаунт, это ваш bucket, вы не сможете обратиться к объекту внутри bucket, владельцем которого вы не являетесь.

00:05:57.480 --> 00:06:10.480

Поэтому нам необходимо, чтобы пользователь был на стороне destination аккаунта, таким образом при записи он относится к тому же аккаунту, соответственно проблем с доступом не возникнет.

00:06:10.480 --> 00:06:23.280

Единственная здесь задача, правильно прописать policy на стороне source аккаунта, source bucket, чтобы эти данные можно было cross-account получить.

00:06:23.280 --> 00:06:39.800

На следующих слайдах вы можете посмотреть детальный разбор вопроса.

00:06:39.800 --> 00:06:42.800

На этом мы завершаем разбор вопроса Practice Test-а.

00:06:42.800 --> 00:06:45.800

Спасибо за внимание.