

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
СТАВРОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ  
УНИВЕРСИТЕТ**

*Кафедра информационных систем*

**УТВЕРЖДАЮ**

Заведующий кафедрой

\_\_\_\_\_

« \_\_\_ » \_\_\_\_\_ 20\_\_ г.

**ЛЕКЦИЯ №6**

**по дисциплине «Облачные технологии»**

**Тема №2 Облачная обработка данных Amazon**

**Занятие №1 Amazon S3**

для студентов направления            38.03.05    «Бизнес-информатика»

ШИФР

наименование

*Рассмотрено УМК*

" " \_\_\_\_\_ 20\_\_ года

протокол N \_\_\_\_\_

**Ставрополь, 2023**

***Цели лекции:***

1. Сформировать информационно-наглядное представление об особенностях функционирования Amazon S3.
2. Показать актуальность и значимость современных облачных технологий в ведении бизнеса.

**Время:** \_\_\_\_\_ 90 мин.

***Учебно-материальное обеспечение:***

1. Опорная лекция.
2. ГОС ВО по направлению подготовки.
3. Рабочая программа дисциплины.
4. Основная и дополнительная литература.

***Распределение времени:***

<b>I. Вступительная часть</b>	5 мин.
<b>II. Основная часть:</b>	
1. Доступ к S3. Web-сервисы	35 мин.
2. Функционирование S3	45 мин.
<b>III. Заключительная часть</b>	5 мин.

## Вводная часть

Amazon Simple Storage Service (S3) — это долгосрочное хранилище данных на основе облачной обработки. Этот сервис работает независимо от остальных сервисов, предоставляемых Amazon. Фактически приложения, которые пользователь разрабатывает для хостинга его собственных серверов, могут использовать Amazon S3, без необходимости перемещения в облако иным образом. Когда компания Amazon называет S3 "простым хранилищем" (simple storage), то она имеет в виду набор функций, предоставляемых сервисом, а не простоту использования. Amazon S3 позволяет переместить данные в облако и извлечь их обратно. При этом нет необходимости знать что-либо о том, как именно и где именно в облаке хранятся ваши данные.

## Первый учебный вопрос – Доступ к S3. Web-сервисы.

Прежде чем получить доступ к S3, необходимо создать учетную запись Amazon Web Services. При этом можно запросить пространство для хранения данных на серверах, расположенных либо в США, либо в Европе, либо даже в Азиатско-Тихоокеанском регионе. Когда речь заходит о хранении данных, выбор местоположения серверов — это вопрос не только места жительства. На решение о месте хранения облачных данных влияют административные вопросы и различные аспекты, касающиеся конфиденциальности информации. Вопросы доступа к S3 рассматриваются исходя из предположения, что выбрано хранилище, ближайшее к пользователю.

Amazon предоставляет доступ к S3 как через SOAP API, так и через REST API. Хотя разработчики обычно лучше знакомы с разработкой Web-сервисов с помощью SOAP, REST представляет собой предпочтительный механизм для доступа к S3, вследствие того, что при обработке больших двоичных объектов (Binary Large Objects, BLOBs) через SOAP API возникают сложности. В частности, SOAP ограничивает размер объекта, которым можно манипулировать в S3, а также ограничивает любую обработку (например, индикатор статуса передачи), которую вам может потребоваться осуществлять над потоками данных по мере того, как они передаются в S3 и обратно.

Интерфейс прикладного программирования Amazon Web Services поддерживает следующие возможности:

- поиск корзин (buckets) и объектов;
- обнаружение их метаданных;
- создание новых корзин;
- загрузка новых объектов;
- удаление существующих корзин и объектов.

При манипулировании корзинами пользователя при желании возможно указать местоположение для хранения содержимого корзины. За исключением случаев, когда пользователю требуется по-настоящему

тонкоструктурированный контроль над взаимодействиями с S3, эксперты рекомендуют использовать "обертку" API (API wrapper) для предпочитаемого языка программирования, которая будет абстрагировать вас от S3 REST API. Например, разработчики программируют на Java с применением Jets3t1. Чтобы начать работу с Amazon S3, наверняка пригодится клиентское приложение командной строки s3cmd для Amazon S3 (<http://s3tools.logix.cz/s3cmd>). Это приложение представляет собой "обертку" с интерфейсом командной строки для доступа к Web-сервисам S3. Данный инструмент написан на Python, а это значит, что это дает возможность изучить его исходный код и он послужит отличным образцом.

Кроме того, Amazon предоставляет доступ к Amazon S3 по протоколу BitTorrent. BitTorrent представляет собой пиринговый (peer-to-peer, P2P) протокол для кооперативного доступа к файлам. Поскольку BitTorrent является стандартным протоколом для совместного доступа к большим двоичным объектам, на рынке предлагается целый ряд клиентов и приложений, предназначенных для потребления и публикации данных с помощью BitTorrent. Если приложение может использовать эту встроенную инфраструктуру, то возможно, имеет смысл воспользоваться преимуществами, предоставляемыми поддержкой протокола BitTorrent со стороны Amazon S3. В общем случае, однако, транзакционные Web-приложения не используют BitTorrent для взаимодействия с S3.

## Второй учебный вопрос – Функционирование S3

Чтобы проиллюстрировать практическое использование S3, воспользуемся утилитой `s3cmd` для передачи файлов в хранилище S3 и обратно. Команды, поддерживаемые этой утилитой, отражают функции API Web-сервисов, на которых она основывается. После того как будет закачена эта утилита, пользователю потребуется сконфигурировать ее, указав ключ доступа к S3 и секретный ключ S3. Вне зависимости от того, какой инструментарий используется — эта утилита или какое-нибудь другое средство, все равно будут нужны эти ключи для доступа к частным корзинам пользователя в S3.

Первое, что потребуется пользователю сделать в S3 — это создать корзину, в которой он будет хранить объекты. Это делается следующей командой:

```
s3cmd mb s3://BUCKET
```

Данная команда создает корзину и присваивает ей указанное вами имя. Как я уже заметил ранее в этой главе, пространство имен для вашей корзины доступно всем клиентам Amazon. Чтобы команда выполнялась успешно, необходимо заменить имя BUCKET на любое другое, уникальное для пользователя. Многие пользователи с тем, чтобы сделать имена корзин уникальными для себя, примерно так, как это делается для доменных имен, предваряют имена корзин префиксами. Тем не менее, необходимо иметь в виду, что какой бы стандарт именования пользователь ни принял, ничто не может помешать другим пользователям нарушать ваше соглашение об именовании.

Существуют правила именования. Имена могут состоять только из строчных букв (символов в нижнем регистре клавиатуры), цифр, точек, символов подчеркивания и дефисов. Начальным символом имени должна быть буква или цифра.

Имя не должно составляться в стиле IP-адреса (иными словами, не

допускаются имена наподобие 10.0.0.1). Длина имени должна составлять не менее 3 и не более 255 символов.

Возможно, пользователю захочется называть свои корзины таким образом, чтобы они образовали собственное виртуальное пространство имен корзины. Например, `com.imaginary.mybucketis`, с учетом имени домена `imaginary.com`. Важно отметить, что ничто не гарантирует того, что никакой другой пользователь не станет использовать домен, выбранный пользователем в качестве префикса имен своих корзины.

Amazon предлагает следующие правила именования корзины, которые позволят пользователям создавать действительные и корректные URL для объектов S3.

1. Не следует создавать имена корзины, в состав которых входят символы подчеркивания (несмотря на то, что официально это разрешено).

2. В идеальном варианте лучше всего ограничить длину имени 63 символами.

3. Не следует создавать имена корзины, завершающиеся дефисом, а также следует избегать ситуаций, когда в составе имени за дефисом следует точка.

После того как корзина будет создана, ее можно начинать заполнять объектами: `s3cmd put LOCAL_FILE s3://BUCKET/S3FILE`

**Например:**

```
s3cmd put home_movie.mp4 s3://com.imaginary.movies/home_movie.mp4
```

Утилита `s3cmd` ограничивает размеры файлов до 5 Гбайт, что является следствием ранее упомянутого ограничения S3. Когда объект потребуется, возможно извлечь его из облака:

```
s3cmd get s3://BUCKET/S3FILE LOCAL_FILE
```

**Например:**

```
s3cmd get s3://com.imaginary.movies/home_movie.mp4 home_movies3.mp4
```

В приведенном примере файл с видео извлечен обратно на настольный компьютер. Существует еще ряд популярных команд, которые возможно использовать.

- Вывод списка всех корзин: `s3cmd ls`
- Вывод списка содержимого конкретной корзины: `s3cmd ls s3://BUCKET`
- Удаление объекта из корзины: `s3cmd del s3://BUCKET/S3FILE`
- Удаление корзины: `s3cmd rb s3://BUCKET`

Удалить корзину можно только в том случае, если она пуста. Прежде чем удалить корзину, необходимо пошагово, один за другим удалить из нее все объекты. Скоро в составе утилиты `s3cmd` появится опция `--recursive`, предназначенная для использования с командой `del`, но необходимо иметь в виду, что она всего лишь будет перечислять содержимое корзины и удалять объекты один за другим. Если пользователь использует API Web-сервисы, то потребуется написать собственную подпрограмму рекурсивного удаления корзины, которая содержит объекты.

## Заключение

Если вы воспринимаете Amazon S3 как удаленную файловую систему, то это заблуждение. Amazon S3 во многих отношениях гораздо примитивнее файловой системы. В действительности пользователь не хранит там нечто, называемое "файлами", пользователь хранит там объекты. Более того, объекты (objects) хранятся в "корзинах" (buckets), а не в каталогах (directories), как в файловой системе. Хотя эти различия могут казаться чисто семантическими, необходимо их учитывать, принимая во внимание следующие важные факты:

- объекты, хранимые в S3, не могут иметь размер, превышающий 5 Гбайт;

- "корзины" (Buckets) существуют в плоском пространстве имен, предоставленном в общий доступ всем пользователям Amazon S3. Пользователь не имеет возможности создавать вложенные корзины и должен быть предельно внимателен к пересечениям в пространстве имен;

- пользователь может сделать свои корзины и объекты доступными для публичного просмотра;

- без помощи инструментов от сторонних разработчиков вы не можете "монтировать" (mount) хранилище S3.

Доцент кафедры «Информационных систем»

к.т.н., доцент

В.Е. Рачков

« \_\_\_ » \_\_\_\_\_ 20 \_\_\_ г.