

2022-11-26 S3

Протокол S3 плавно завоевывает все больше информационных систем, и его поддержка становится практически повсеместной. С развитием облачной модели потребления потребность в механизме гибкого выделения дисковых ресурсов для хранения непрогнозируемых объемов данных привела к трансформации известных ранее объектных протоколов в то, что мы сейчас подразумеваем под этим термином. Объектные системы хранения данных существовали задолго до того, как компания Amazon решила перестать продавать книги и начать предоставлять облачный сервис всем желающим. Изначально уделом объектных систем хранения данных было хранение медиа файлов, медицинских записей, финансовых документов внутри крупных коммерческих организаций. В основном это были предки современных программно-определяемых систем хранения данных. Объектные СХД обеспечивали заказчикам следующие преимущества, по сравнению с традиционными решениями:

1. Защита данных на уровне объекта за счет технологии избыточного кодирования или репликации.
2. Механизм восстановления отказоустойчивости или "Self Healing" позволяющий создавать практически вечные системы хранения данных, которые следят за состоянием объектов и не требуют, по заявлению производителей, резервного копирования.
3. Использование большого количества серверов стандартной архитектуры, оснащенных дисками максимального объема с наиболее низким соотношением \$/ТБ.

Для доступа к данным, хранимым на первобытных объектных хранилищах, требовалась поддержка проприетарного API выбранной системы и полная интеграция на уровне программного обеспечения. Поэтому внедрение таких систем не могло происходить в стиле "Box Moving". В большинстве случаев двигателем таких проектов служили разработчики.

С появлением протокола S3 случилось одно важное событие. В связи с его обширной популярностью и статусом стандарта де-факто в объектном доступе проблема несовместимости протоколов была наконец решена. Единственным протоколом объектного доступа стал s3-совместимый протокол, часто, у многих со своими отдельными хранимыми процедурами, но в большинстве совместимый со всеми системами, которые его хоть как-то поддерживают.

Как я уже не раз говорил главное преимущество протокола S3, если рассматривать его с точки зрения обычного системного интегратора, это возможность гибкого выделения и оплаты ресурсов хранения. Представим себе типичную ситуацию, которая идеально ложится на модель использования S3.

У вас популярный веб ресурс, пусть это будет мессенджер, блог, видео хостинг, не важно. Все веб ресурсы современной эпохи так или иначе работают с медиа контентом. И чем ваш ресурс популярнее, чем дольше он существует – тем больше у вас становится контента, который необходимо хранить, обрабатывать и отдавать пользователям. В традиционной модели потребления (без облаков например) мы имеем несколько физических серверов, платформу виртуализации, набор виртуальных машин, централизованную СХД и возможно отдельную СХД для «медиа файлов» которая чуть дешевле чем наше основное хранилище на NVMe дисках.

Представьте себе ситуацию, когда сегодня у вас 10 ГБ медиа, завтра 12, а послезавтра 50. Рост который невозможно прогнозировать, на который работает весь ваш маркетинговый

отдел, потому что пользователи это деньги. Как вы будете планировать систему хранения медиа? На год вперед? На три?

Когда мы имеем дело с традиционным Enterprise то темпы роста обычно ясны и они не велики. Мы закладываем 10-20% роста на год и заказчик считает это «чересчур». Но в современном мире ютуба, твиттера и тиктока изменение объемов такое значительное, что закладывать емкости на год вперед мало кто себе может позволить.

S3 позволяет платить непосредственно за использованное дисковое пространство и не имеет такого ограничения как например объем файловой системы, размер LUN. При использовании S3 вам не нужно думать о том на сколько увеличить логический том сегодня, чтобы хватило емкости на завтра, потому что если мы говорим про облако - вся выделенная емкость виртуальным машинам должна быть оплачена в полном объеме, не важно используете вы ее или нет.

From:

<https://micronode.ru/> - **micronode.ru**

Permanent link:

<https://micronode.ru/blog/2022/11/26>

Last update: **2022/11/26 20:31**

