Автор:
05.02.18 07:04 -

Проект CoreOS , на днях купленный компанией Red Hat, опубликовал релиз etcd 3.3 , высоконадёжного распределённого хранилища параметров конфигурации, задаваемых в форме ключ/значение. Основным назначением etcd является предоставление унифицированного механизма хранения конфигурации и информации о работающих сервисах для изолированных контейнеров с типовой начинкой. Например, etcd применяется в платформе оркестровки контейнеров Kubernetes для обеспечения хранения данных кластера. Код etcd написан на языке Go и распространяется под лицензией Apache 2.0.

Etcd позволяет организовать единое хранилище конфигурации для группы серверов, которое реплицируются на все узлы и поддерживается в синхронизированном состоянии с использованием протокола Raft . Наличие копии данных на всех хостах позволяет исключить потерю конфигурации при выходе из строя отдельного узла. В etcd также могут сохраняться временные данные, для которых предусмотрена возможность определения времени жизни записи. Для доступа к конфигурации предоставляется простой API , основанный на использовании gRPC .

Имеется встроенная возможность отслеживания изменения состояния ключа или директории с вызовом обработчика в случае обнаружения изменения (например, можно применить новое значение параметра конфигурации). Для защиты канала связи при обращении из внешней сети предоставляется поддержка TLS-шифрования, аутентификации клиентов по ключам и разграничения доступа через ACL. На типовом оборудовании etcd обеспечивает производительность порядка 10 тысяч операций записи в секунду. Для доступа к базе можно использовать утилиту etcdctl .

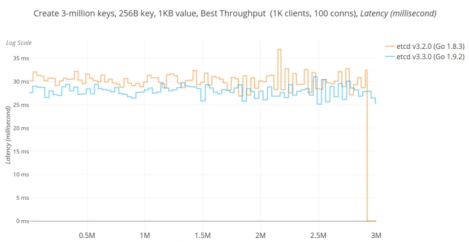
Основные новшества :

- Улучшена работа бэкенда bbolt , представляющего собой форк Bolt DB , применяемый для хранения локальных данных на каждом узле. Добавлено несколько

режимов поддержания списка свободных блоков, позволяющего оптимизировать доступ к блокам, ставшими неактуальными после выполнения транзакций и доступными для повторного использования (заполнения новыми данными). Список свободных блоков сохраняется на диске чтобы избежать ресурсоёмких операций сборки мусора во время запуска. Обратной стороной поддержания такого списка является существенное разрастание списка при выполнении большого числа транзакций. В качестве компромисса выбрана тактика перестроения списка свободных блоков во время выполнения операции восстановления ("recover"). На перестроение БД размером 10 Гб на современных системах тратится около 2 секунд. Для управления данным поведением представлено две опции "freelist sync" и "freelist no sync". Также увеличена производительность работы сборщика мусора;

- Значительно улучшен механизм восстановления соединения клиента в сетях с нестабильным каналом связи, который теперь учитывает возможное сегментирование сети и временные обрывы соединения;
- Добавлен экспериментальный режим "--experimental-corrupt-check-time" для мониторинга состояния хранилища на предмет возможных повреждений данных (например, из-за ошибок на уровне файловой системы) во время работы. Также добавлен режим "--experimental-initial-corrupt-check" который во время запуска проводит контроль целостности через запрос у пира контрольных сумм CRC32 для известной ревизии и их сверку с фактическим состоянием хранилища перед началом обработки запросов клиентов и других пиров;
- Добавлен пакет clientv3/ordering с реализацией механизма причинной консистентности (causal consistency) для сериализированных запросов на чтение, обеспечивающий логическую целостность порядка операций чтения и записи, независимо от того к какому узлу обратился клиент (т.е. клиент всегда получит доступ к самой свежей ревизии);
- Добавлен режим "--experimental-enable-v2v3", позволяющий эмулировать старый API v2 доступа к хранилищу поверх нового API v3 на базе gRPC, обеспечивая при этом более высокую эффективность и масштабироваие по сравнению с нативным API v2, который теперь объявлен устарвшим;
- Добавлена возможность поддержания отдельного списка отозванных сертификатов X.509 (Certificate Revocation List), помимо штатного списка CRL от удостоверяющего центра. Для включения следует использовать опции "--client-crl-file" и "--peer-crl-file";
- Добавлена возможность применения в TLS сертификатов, охватывающих группу поддоменов по маске (*.example.com в поле SAN (Subject Alternative Name));
- Добавлена возможность "--peer-cert-allowed-cn" для аутентификации между пирами на основе проверки совпадения имени Common Name (CN) в сертификате;
- Для работы в случае сегментации сети и невозможности прямого подключения клиента добавлена экспериментальная поддержка gRPC-прокси для трансляции запросов клиентов к кластеру etcd при помощи API v3;
- Проведена большая работа по оптимизации производительности. При проведении тестирования отмечено заметное снижение задержек и увеличение пропускной способности с 32976 запросов в секунду до 35682 запросов в секунду.

Автор:
05.02.18 07:04 -



Read more http://www.opennet.ru/opennews/art.shtml?num=48024