

Пантелеев Николай Николаевич, преподаватель военного учебного центра,

«Цикл связи» РТУ МИРЭА, РФ, г. Москва

Панов Сергей Сергеевич, студент военного учебного центра Российского

Технологического Университета МИРЭА, РФ, г. Москва

Матвеев Артем Владленович, студент военного учебного центра Российского

Технологического Университета МИРЭА, РФ, г. Москва

СРАВНЕНИЕ ХАРАКТЕРИСТИК И ВОЗМОЖНОСТЕЙ СОВРЕМЕННЫХ ФАЙЛОВЫХ СИСТЕМ LINUX: EXT4, XFS, BTRFS

Аннотация: В этой статье сравниваются три современные файловые системы Linux: ext4, Btrfs и XFS. Обсуждается их архитектура и функциональные особенности. Файловые системы сравнивались с точки зрения производительности. Кроме того, учтены специфика твердотельных накопителей и адаптация к работе с ними. Для тестирования функциональности и производительности этих трех файловых систем Linux используется дистрибутив Ubuntu 15.10.

Ключевые слова: файловая система; ext4; btrfs; xfs.

Annotation: This article compares three modern Linux file systems: ext4, Btrfs and XFS. Their construction and functional features were discussed. File systems were compared in terms of performance. Additionally modern SSD drives were included in discussion. Ubuntu 15.10 distribution was used to test functionality and performance of these three Linux file systems.

Keywords: file system; ext4; btrfs; xfs.

Файловая система используется для управления способом записи и чтения данных. Файловая система определяет формат содержимого и способ

физического хранения информации, которую принято группировать в виде файлов. Конкретная файловая система определяет размер имен файлов (и каталогов), максимальный возможный размер файла и раздела, набор атрибутов файла. Некоторые файловые системы предоставляют сервисные возможности, например, разграничение доступа или шифрование файлов.

Почти все операционные системы используют файловые системы. В случае самых популярных операционных систем, таких как Windows, Linux или MacOS, у нас есть выбор из множества файловых систем в зависимости от потребностей. Значительное развитие информационных технологий в последние годы создало большое количество проблем, с которыми приходится сталкиваться файловым системам. В системах Linux все создается в виде файлов. Файл представляет собой набор связанной информации. Пользователь видит только имя файла, в то время как Linux рассматривает файл как последовательность байтов. Выбор файловой системы зависит от приложения. Одна подойдет для систем с высокими требованиями к непрерывной и безотказной работе, другая — для систем, где критична производительность, и ещё одна — для систем, где важна возможность миграции или изменения емкости «на лету» (например, в случае виртуальные машины).

В связи со значительным развитием технологий возросли требования, связанные с безопасностью данных и эффективностью информационных систем. Переход от магнитных накопителей к полупроводниковым технологиям создал растущий спрос на совершенно новые функции файловых систем.

Linux предлагает широкую палитру файловых систем. В статье будут сравниваться три самых популярных: ext4, XFS и Btrfs. Ext4 это одна из самых популярных файловых систем по умолчанию во многих дистрибутивах Linux. Файловая система ext4 является продолжением файловой системы ext, которая была первой файловой системой, созданной с нуля для Linux [2]. XFS доступна в Linux уже двадцать лет, постоянно развивается крупными ИТ-компаниями, позволяет обрабатывать очень большие файлы и тома [6]. XFS можно рассматривать как современную файловую систему для эффективного способа

работы с дисками в эпоху «больших данных». Vtrfs - самая молодая файловая система, предлагаемая в Linux [4]. Она постоянно развивается, имеет много новых функций, таких как оптимизация или безопасность, но многие из них все еще находятся на стадии тестирования [4].

Файловая система. Файловая система отвечает за организацию, построение иерархии и создание структуры упорядочения файлов. Кроме того, она выполняет операции, необходимые для работы операционной системы, в том числе:

- создает и удаляет файлы;
- открывает файлы для чтения и записи;
- ищет данные в файлах;
- закрывает файлы;
- создает каталоги;
- перечисляет содержимое каталогов;
- удаляет файлы из каталогов [1].

Почти с самого начала, по историческим причинам (унаследованное разнообразие версий UNIX и производителей), Linux имел в своем распоряжении более одной файловой системы. Уровень абстракции под названием VFS (Virtual File System) был создан для облегчения внедрения новых файловых систем. Уровень VFS — это интерфейс между ядром и реализацией файловой системы. Благодаря этому слою операции записи и чтения с точки зрения программиста одинаковы. Задача VFS заключается в преобразовании основных команд в команды, соответствующие спецификации файловой системы [1].

Файлы в системах UNIX относятся к так называемым инодам. Каждый файл в Linux имеет свой индексный дескриптор. В каждом *i*-узле находятся элементы структуры, описывающие файл (кроме данных и имени) и указатели на блоки данных [1]. После загрузочной области (возможно, пустой) идет суперблок, содержащий атрибуты и метаданные файловой системы [1]. Последним из основных элементов является область файловой системы, содержащая блоки данных.

EXT4. Система ext4 является продолжением файловой системы ext. Система ext — это первая версия самой популярной файловой системы, доступной в Linux. В ext имя файла составляло 255 символов, максимальный размер раздела и файла — 2 ГБ [2]. Система, однако, имела большой недостаток - использование блочного размещения приводило к невыгодному проявлению прерывистости областей данных на жестком диске, т.е. к фрагментации [1].

В ext2 проблема, связанная с фрагментацией файлов, была устранена за счет использования механизма выделения закрытых блоков и предварительного выделения [1; 2]. Расположение блоков, данных осуществляется двумя способами: напрямую — адреса первых блоков хранятся непосредственно в иноде — и косвенно — адреса блоков хранятся в блоке, адрес которого находится в иноде [1]. Также введен механизм восстановления после сбоев (fsck), который запускается при старте системы — он исправляет повреждения, восстанавливая поврежденные файлы в соответствующую директорию [5]. Сканирование было менее эффективным, чем со следующими журналируемыми файловыми системами.

Файловая система ext3 была похожа на ext2 по структуре. Наиболее важной функцией, которая была введена в ext3, был механизм учёта публикаций [3]. Учет защищает файлы от потери или повреждения и заключается в записи метаданных — информации о заданном — сначала в журнал, а затем сохраняя данные на диск.

Новая функция, представленная в ext3, — это индексация каталогов. В ext3 за это отвечает функция «hashedtrees», благодаря которой каталоги имеют вид дерева [5]. Каталогизация необходима, когда у вас есть большое количество файлов в каталогах. Древовидная структура значительно сокращает время доступа к искомому файлу. Система ext3 позволяет разделить блок на равные части, в которых можно хранить файлы, и таким образом уменьшает фрагментацию в случае большого количества мелких файлов.

В файловой системе ext4 размер одного файла увеличен до 16 ТиБ, а поддержка одиночных разделов до 1 ЭиБ, количество возможных подкаталогов увеличено до 64 000 [5]. Главной особенностью, отличающей ext4 от более

старых версий ext, является механизм экстентов [5], заменяющий блочную косвенную адресацию. Экстенты — это непрерывный набор блоков, в которых хранятся файлы, экстенты хранятся в иноде вместо указателей на блоки [5]. Экстенты увеличили размер одного файла, увеличили размер одного раздела и уменьшили размер метаданных. Уменьшение размера метаданных значительно ускоряет работу с большими файлами. Включение службы экстентов позволяет реализовать предварительное выделение блоков, которое отвечает за непрерывное выделение пространства для файла.

Новой функцией ext4 является отложенное выделение [3] — метод, который откладывает запись данных на диски, сохраняя их в кэше. Отложенное выделение происходит на виртуальном уровне файловой системы. Расширением отложенного распределения является многоблочное выделение (mballoc). Преимуществом этого решения является увеличение производительности при сохранении растущих файлов — сочетание одновременного и многоблочного выделения увеличивает шансы того, что последующие выделения будут сгруппированы в одно. Еще одним преимуществом является уменьшение фрагментации данных для растущих файлов, и высокая вероятность того, что временные файлы не будут выделены [2].

В ext4 механизм учета также был улучшен благодаря добавлению контрольных сумм журнала [5]. Журнал является наиболее часто используемой областью на диске, поэтому повреждение этой области может привести к сбою всей файловой системы. Контрольная сумма — это последовательность символов, полученная в результате математических операций при передаче данных. В случае файловой системы контрольная сумма вычисляется для каждой транзакции и дескриптора группы блоков [5]. Новая функция, которая была введена, - это барьерный механизм. Он отвечает за выдачу команд драйверу диска для записи данных в определенном порядке, что повышает согласованность данных на диске [5].

XFS. XFS — это высокопроизводительная 64-битная файловая система с поддержкой журналирования. Она поддерживается в большинстве

операционных систем Linux. При разработке файловой системы XFS были приняты следующие цели [1]:

- файловая система должна быть пригодна для работы на научных файловых серверах, коммерческих серверах обработки данных и электронных медиа-серверах
- должна быстро восстанавливаться после сбоев, обеспечивать большую доступность данных, поддерживать данные на диске в согласованном состоянии
- должна поддерживать 64-битные файлы
- должна эффективно обрабатывать разреженные файлы, то есть файлы, которые содержат дыры в любом месте (области, где запись никогда не происходила, считываются как блоки нулей);
- повышенная производительность для обработки небольших файлов размером менее 1 КБ;
- должна эффективно обрабатывать большие каталоги;
- поддержка списка управления доступом и функций POSIX 1003.6 – расширения безопасности системы, соответствующие критериям, опубликованным Министерством обороны США;
- возможность динамического изменения логических размеров блоков [1; 6].

Файловая система поддерживает файлы размером до 8 EiB и максимальный размер раздела до 16 EiB. Длина имени файла представлена строкой из 255 символов в таблице ASCII [18]. Размер единицы размещения составляет от 512 КБ до 1 МБ. Как и все файловые системы, XFS также работает под управлением VFS. XFS — это журналируемая файловая система, что означает, что обновление метаданных происходит последовательно в области журнала, прежде чем произойдет фактическое обновление [1]. В случае сбоя операции записи можно повторить или отменить, используя данные для восстановления файловой системы, содержащиеся в журнале. Основными компонентами XFS являются:

- менеджер журналов — любые изменения метаданных

последовательно регистрируются в специально отведенной области. Каждая файловая система ведет отдельный журнал, чтобы можно было быстро восстановить неисправную файловую систему.

- менеджер буферного кеша — блоки диска хранятся в буферном кеше, здесь хранятся метаданные файловой системы и файловые данные
- менеджер блокировки — выполняет блокировку пользовательских файлов
- менеджер дискового пространства — управляет распределением дискового пространства в файловой системе
- менеджер атрибутов — выполняет операции над атрибутами файловой системы
- интерфейс вызова системы и VFS
- менеджер пространства имен — переводит пути доступа в ссылки на файлы [1].

Файловая система XFS разделена на группы размещения одинакового размера. Группа размещения — это автономная единица файловой системы, которая содержит отдельные структуры, необходимые для управления своим пространством [1].

XFS также обеспечивает дефрагментацию файлов во время работы, т.н. онлайн-дефрагментация и онлайн-изменение размера, так что файловая система может увеличиваться в размерах, пока на физических устройствах есть свободное место [6]. XFS также поддерживает механизм учета для обеспечения согласованности данных в случае сбоя питания или системного сбоя [1; 6]. XFS обеспечивает учет метаданных, которые записываются перед обновлением дисковых блоков. Файлы системного журнала сохраняются на отдельной части диска, которая не используется при нормальной работе системы. В XFS есть два типа журналов: логический и физический [1]. Первый логически описывает в удобочитаемой форме, какие действия были выполнены, в то время как физический журнал сохраняет копию блоков, измененных во время каждой операции. Обновления журнала выполняются асинхронно, что повышает

производительность файловой системы [1]. В случае сбоя XFS может восстановить файлы, которые не были сохранены в соответствии с информацией журнала. Восстановление выполняется автоматически при первой загрузке файловой системы после сбоя, скорость восстановления зависит не от размера файлов, а от выполняемых с ними операций.

Btrfs. Другой рассматриваемый формат файловой системы — 64-битный Btrfs. В системе Btrfs максимальный размер файла составляет 16 EiB, пространство имен — 255 символов из таблицы ASCII, а количество файлов, которые могут поместиться в каталоге, — 264 [4]. Основные характеристики файловой системы Btrfs:

- использование механизма копирования при записи для создания снимков системы;
- возможность восстановления состояния системы из точек (снапшотов) в прошлом;
- использование контрольных сумм для обеспечения целостности данных;
- более эффективное использование дискового пространства за счет удобного сжатия данных;
- возможность работы на нескольких томах, полная совместимость с RAID-массивами;
- возможность динамического изменения емкости тома путем удаления и добавления физических устройств [4].

Система Btrfs полностью основана на лесе B-дерева в измененном варианте. В системе Btrfs используется модифицированная версия, называемая деревьями B+ [4]. Были удалены связи между соседними листьями, вместо этого создан односторонний список, что значительно ускоряет обход содержимого структуры. В архитектуре системы Btrfs структура дерева была обогащена несколькими решениями: обновление дерева сверху до листьев, удалены связи между листьями, введен подсчет ссылок. Благодаря этому типу модификации добавление и удаление элементов не требует копирования всего дерева, а только

пути к изменяемому элементу. В файловой системе Vtrfs дерево используется как первичное дерево. Этот тип структуры позволяет создать динамическую систему хранения данных. Вся информация и структуры, входящие в архитектуру системы (за исключением суперблока), хранятся внутри леса деревьев [3].

Как уже упоминалось, файловая система Vtrfs основана на механизме копирования при записи [4]. Механизм копирования при записи — это прием оптимизации, когда вам нужно работать с большими объемами данных. Вместо фактического копирования памяти, которое отнимает много времени и памяти, возвращается только указатель на исходные данные. Копирование происходит тогда, когда возникает необходимость их модификации. Важнейшим преимуществом является отсутствие возможности создания копии не модифицированного файла [3]. В случае Vtrfs функция COW позволяет копировать сектор диска, при этом диск перезаписывается новым фрагментом информации [4]. Это упрощает создание резервных копий. До сих пор для этого приходилось физически копировать данные в другое место на диске.

Из-за структуры Vtrfs физическая копия не требуется. Достаточно сделать дополнительную ссылку в файловой системе, благодаря которой два отдельных файла будут указывать на один и тот же блок данных, физически хранящийся на носителе. Решение не только быстрое, но и экономичное, поскольку копируется только та информация, которая отличается в исходном файле и его копии [4].

В одном томе Vtrfs может быть несколько подтомов [4]. Подтом может действовать как отдельная файловая система. Разделение не требуется при создании подтомов. Подтома создаются из доступного дискового пространства. Созданный том рассматривается как корневой каталог [4]. В сочетании с механизмом копирования при записи вы можете создавать резервные копии всей файловой системы, не теряя много места на диске. При этом изменяются только связи в логических структурах, а копирование происходит только после модификации исходного файла. В случае удаленных и облачных решений нет необходимости копировать все данные при обновлении данных. Копируются только измененные сектора [4]. Решения, внедренные в Vtrfs, значительно

ускоряют работу с большими файлами [4]. Введение контрольных сумм значительно повышает защищенность файлов от микроповреждений. В Vtrfs для каждого сектора диска, содержащего информацию, создается контрольная сумма. При каждом чтении контрольная сумма пересчитывается, и при изменении любого из битов контрольная сумма не будет совпадать с записанной ранее. В случае ошибки система Vtrfs может автоматически восстанавливать информацию, тем самым предотвращая необратимое повреждение файла [4].

Система Vtrfs вводит ряд функций для поддержки управления разделами без необходимости размонтировать файловую систему [4]. Решение используется во время динамического изменения размера тома. Благодаря функции динамического изменения размера тома можно изменять размер разделов во время работы системы.

Новой функцией, представленной в Vtrfs, является функция моментального снимка [4]. Традиционные менеджеры томов требуют, чтобы в моментальный снимок был включен весь логический диск. Vtrfs, с другой стороны, позволяет создавать снимки как целых папок, так и отдельных файлов, расположенных в любом месте на диске [4]. Моментальный снимок — более полезное решение, чем копирование всего тома. В производственных средах можно избежать простоя машины и сохранить делимость данных при архивировании целых файлов или папок. Снапшоты можно планировать по расписанию, что значительно облегчает администраторам возврат к заархивированной версии файла (в случае модификации или удаления) без остановки производственной среды [4].

Работа над Vtrfs продолжается. Некоторые из функций, которые будут добавлены: быстрая проверка файловой системы, динамическая проверка файловой системы, балансировка нагрузки данных при поддержке многопоточных машин, добавление новых алгоритмов сжатия и функции производительности передачи.

Тесты. В тестах проверялась скорость чтения и записи большого количества маленьких файлов и больших файлов. Тесты проводились на

ноутбуке Acer Swift 3 с твердотельным накопителем на 500 ГБ и установленной системой Ubuntu 15.10. Для запуска тестов использовался проприетарный простой скрипт bash.

Перед выполнением тестов необходимо было выполнить правильную настройку системы. Для того, чтобы полученные тесты были достоверными, система Ubuntu трижды устанавливалась с тестируемой файловой системой. Тесты производительности будут включать количество времени, которое файловая система занимает операцию копирования. Тесты будут включать копирование большого количества небольших файлов размером 1 КБ, 1 МБ, 1 ГБ, 5 ГБ, 10 ГБ, 75 ГБ. Для тестов использовался скрипт bash, который предназначен для генерации файлов, выполнения операции копирования заданного раздела из папки в папку, измерения времени операции, удаления файлов. Сценарий bash, используемый для тестирования производительности файловой системы, показан на рисунке 1.

```
1. #!/bin/bash
2. NUM_FILES=10 #ilość plików
3. SIZE_FILE=500M #rozmiar pliku
4. FROM_COPY=from_copy
5. TO_COPY=to_copy
6. mkdir $FROM_COPY #folder tworzenia plików
7. mkdir $TO_COPY #folder do którego kopiowane są pliki
8. cd $FROM_COPY #przejdźcie do katalogu, w którym
   generowane będą pliki
9. echo "Tworzę pliki"
10. for i in $(seq $NUM_FILES); do
11. dd if=/dev/zero of=file_${i}bs=$SIZE_FILE count=1 &>
   /dev/null #funkcja generująca pliki
12. done
13. echo "Utworzyłem pliki"
14. cd ..
15. echo "Rozpaczynam kopiowanie"
16. echo "Kopiowanie $NUM_FILES plików o rozmiarze
   $SIZE_FILE zajęło: "
17. time(cp $FROM_COPY $TO_COPY -rf; sync)#pomiar
   czasu kopiowania
18. rm $FROM_COPY $TO_COPY -rf#kasowanie plików
19. sync
```

Рисунок 1. Скрипт Bash

Результаты исследования. Тесты показали, что файловая система Btrfs

является наиболее эффективной из обсуждаемых файловых систем. На рисунках 2 – 8 показаны результаты испытаний. Она показала лучшее время в шести из семи проведенных тестов. Тестирование показало, что Vtrfs хуже справлялась с большим количеством маленьких файлов, но очень хорошо справлялась с большими файлами. XFS оказалась на второй позиции, получив лучший результат в одном из шести проведенных тестов. XFS потребовалось примерно на 10% меньше времени для выполнения операций с большими файлами. XFS хуже всего справлялась с большим количеством маленьких файлов, показатель XFS был на 50% хуже, чем у BTRFS. Во втором тесте система XFS получила результат, аналогичный Vtrfs, в остальных четырех тестах результат был от 3% до 6% хуже результатов, полученных Vtrfs.

Система Vtrfs — это файловая система, которая используется для обеспечения максимальной производительности. Высокая эффективность работы с файлами в сочетании с адаптацией для поддержки SSD-накопителей — очень хорошее сочетание для высокопроизводительных решений. Система XFS, хотя и существует на рынке более двадцати лет, по производительности не уступает последней системе Vtrfs. XFS находится в постоянном развитии с 2013 года и вскоре может заменить одну из самых популярных систем ext4. Ext4 постепенно перестает быть файловой системой по умолчанию во многих дистрибутивах Linux. Он менее эффективен на других проверенных файловых системах и занимает больше всего места при монтировании на раздел.

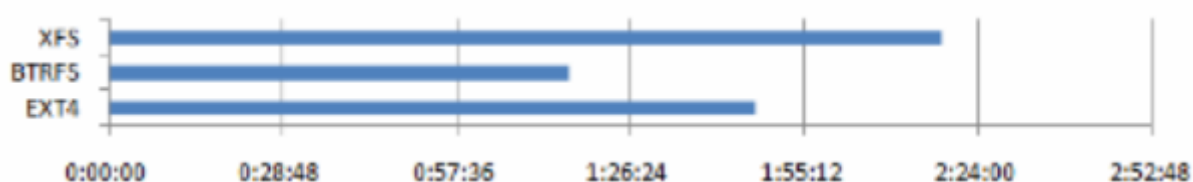


Рисунок 2. Результаты тестов производительности для отдельных файловых систем для файла размером 1КВ

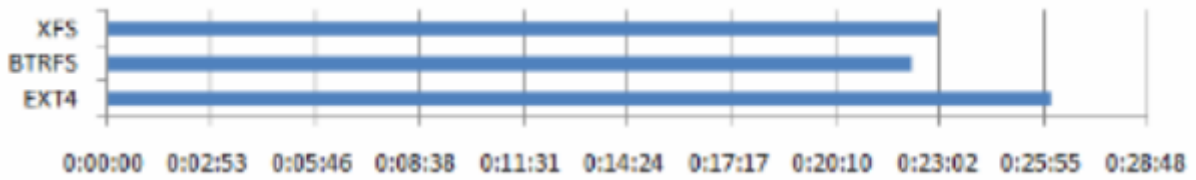


Рисунок 3. Результаты тестов производительности для отдельных файловых систем для файла размером 1МБ

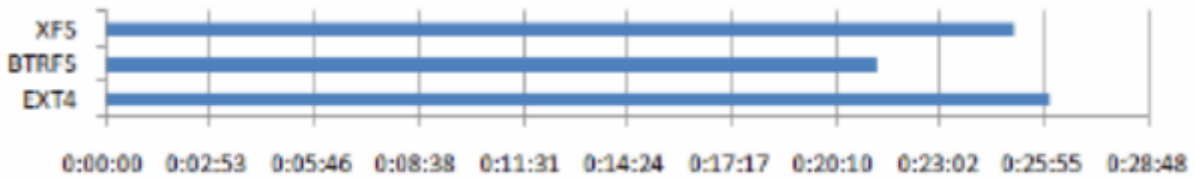


Рисунок 4. Результаты тестов производительности для отдельных файловых систем для файла размером 1ГБ

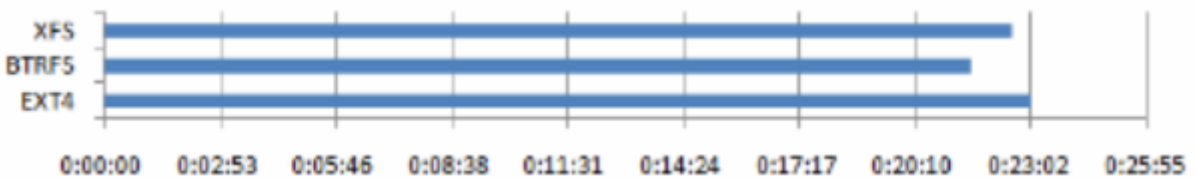


Рисунок 5. Результаты тестов производительности для отдельных файловых систем для файла размером 5ГБ

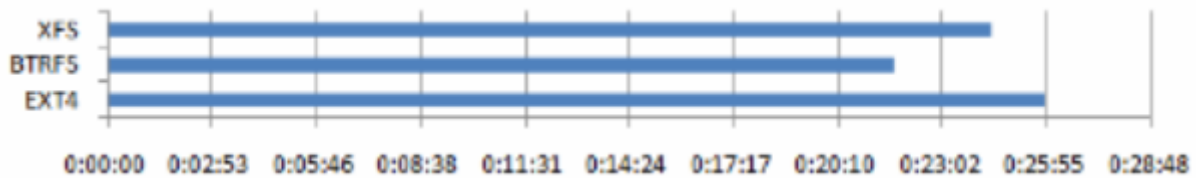


Рисунок 6. Результаты тестов производительности для отдельных файловых систем для файла размером 10ГБ

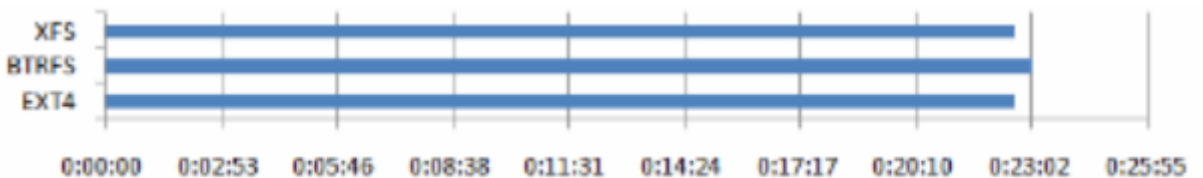


Рисунок 7. Результаты тестов производительности для отдельных файловых систем для файла размером 50ГБ

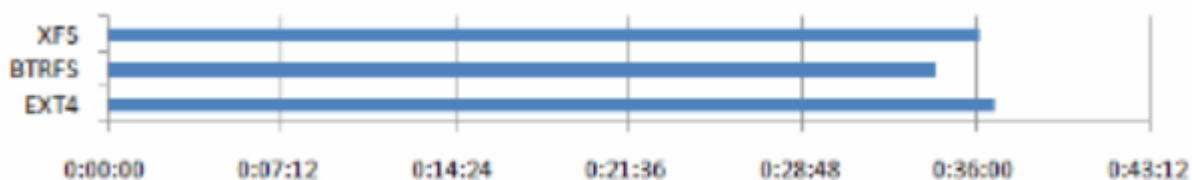


Рисунок 8. Результаты тестов производительности для отдельных файловых систем для файла размером 70ГБ

Список функциональных возможностей. В таблице 1 перечислены функциональные особенности сравниваемых файловых систем.

Таблица 1. Функциональные особенности сравниваемых файловых систем

Функциональная особенность	EXT4	XFS	BTRFS
Программы тестирования	+	-	+
Механизм контрольных сумм	+	-	+
Механизм отсроченного распределения	+	+	+
Динамическое увеличение размера тома	С помощью LVM	+	+
Индексация каталога	+	-	-
Обработка разреженных файлов	-	+	-
Поддержка 64-битных файлов	-	+	+
Снимки системы	-	+	+
Дефрагментация во время работы файловой системы	-	+	+
Механизм учета	+	-	+
Преобразование в другую файловую систему	+ на ext 3	-	+ на ext 4
Гарантированная скорость передачи данных	-	+	-
Поддержка RAID	-	-	-

Заключение. На данный момент система Btrfs оказалась лучшей — она самая эффективная, имеет ряд интересных особенностей и постоянно

развивается. Vtrfs идеально подходит для домашних и некоммерческих решений. XFS был вторым по производительности, но отличается от Vtrfs, очень стабилен и эффективно обрабатывает системы с большим количеством мелких файлов. В сочетании со сжатием «на лету» это дает неограниченные возможности для расширения файловой системы. Система ext4 оказалась худшей. Хотя легко восстанавливать файлы и выполнять сжатие «на лету» — только при использовании с разделом GTP — это все же скорее набор улучшений, чем новая в том смысле, переработанная файловая система, которая может удовлетворить потребности любого пользователя и спрос на ИТ-рынке

Библиографический список:

1. Бар М. Linux файловые системы, Wyd. I Warszawa: RM 2002. ISBN 83-87-02948-1.
2. Зайдель У., Файловая система ext// Журнал “Linux Magazine.” – 2013 – № 12 – С. 63-66.
3. Брендель Дж.К., Исследование Ext3, Ext4, XFS и Vtrfs// Журнал “Linux Magazine.” – 2014, № 9 – С.18- 21.
4. Система Vtrfs [Электронный ресурс]. Режим доступа: https://Vtrfs.wiki.kernel.org/index.php/Main_Page (Дата обращения: 17.10.2022).
5. Система ext4 [Электронный ресурс]. Режим доступа: https://ext4.wiki.kernel.org/index.php/Main_Page (Дата обращения: 17.10.2022).
6. Система XFS [Электронный ресурс]. Режим доступа: http://xfs.org/index.php/Main_Page (дата обращения: 17.10.2022).