

2. Мартынова, Р. Ю., Вереитина, И. А., Богиш, А. В. Your computerland: [электрон. учеб. пособие] / Р. Ю. Мартынова, И. А. Вереитина, А. В. Богиш. – Одесса: ОНАХТ, 2010. – 1 CD диск

3. Вереитина, И. А., Богиш, А. В. Your Amazing Ecoland: [электрон. учеб. пособие] / И. А. Вереитина, А. В. Богиш – Одесса: ОНАХТ, 2013. – 1 CD диск

4. Вереитина, И. А., Пустовой, Б. В. Your Professional Portfolio: [электрон. учеб. пособие] / И. А. Вереитина, Б. В. Пустовой – Одесса: ОНАХТ, 2014. – 1 CD диск

УДК 004.45

ИСПОЛЬЗОВАНИЕ ФАЙЛОВОЙ СИСТЕМЫ ZFS В LINUX

А.В. Бережняк¹, Н.В. Карпенко²

¹студент 4 курса, кафедра электронных вычислительных машин, Днепропетровский национальный университет им. Олеся Гончара, г. Днепропетровск, Украина, e-mail: core@irc.lg.ua

²кандидат физико-математических наук, доцент кафедры электронных вычислительных машин, Днепропетровский национальный университет им. Олеся Гончара, г. Днепропетровск, Украина, e-mail: karpenko_nadija@mail.ru

Аннотация. В данной работе рассмотрены основные характеристики и особенности современной файловой системы ZFS, приведен пример ее установки в операционной системе CentOS Linux и осуществлено сравнение основных характеристик ZFS и других часто используемых в Linux файловых систем.

Ключевые слова: zfs, linux, zol, centos, файловая система.

USING ZFS FILE SYSTEM IN LINUX-BASED OPERATING SYSTEMS

Oleksii Berezhniak¹, Nadiia Karpenko²

¹Student 4th year, Department of Computer, Dnipropetrovs'k National University named by Oles Honchar, Dnipropetrovs'k, Ukraine, e-mail: core@irc.lg.ua

²Ph.D. in Physics and Mathematical Sciences, Associate Professor, Department of Computer, Dnipropetrovs'k National University named by Oles Honchar, Dnipropetrovs'k, Ukraine, e-mail: karpenko_nadija@mail.ru

Abstract. The paper covers the analysis of characteristics and features of the ZFS file system, current status of "ZFS on Linux" implementation. Comparison of ZFS and other popular Linux file systems has been made.

Keywords: zfs, linux, zol, centos, file system.

Введение. Файловая система ZFS создана компанией Sun Microsystems для операционной системы Solaris. Разработка была начата в 2001 году, а первая версия представлена в июне 2006 года в обновлении «6/06» для

Solaris 10 [1]. Вместо оригинального названия «Zettabyte File System» в настоящее время используется простая аббревиатура ZFS.

Файловая система предоставляет богатые возможности по нахождению и исправлению ошибок, встроенный менеджер томов, многоуровневый кэш чтения, кэш записи, практически недостижимые ограничения на максимальный размер файла (2^{64} байт) и файловой системы (2^{78} байт).

Цель работы. Составляющей частью курса «Операционные системы» являются лекции, посвященные файловым системам. Цель данной работы – анализ особенностей ZFS, возможностей ее использования в Linux и сравнение с другими часто используемыми в Linux файловыми системами.

Основная часть. Проанализированы ключевые особенности, принципы организации и предоставляемые возможности файловой системы ZFS в целом и ее реализации для Linux в частности. Произведена установка ZFS в CentOS Linux и ее сравнение с другими файловыми системами.

Ключевые особенности. В ZFS предусмотрены встроенные механизмы обнаружения и исправления ошибок – каждая операция чтения/записи данных и метаданных сопровождается вычислением их контрольной суммы и автоматической коррекцией ошибок. Избыточность может быть обеспечена на уровне файлов (путем записи нескольких копий каждого файла) и на уровне дисков (путем использования различных уровней RAID). Для контроля доступа к файлам и каталогам поддерживаются два вида ACL – POSIX и NFSv4. Встроенный в ZFS программный RAID имеет преимущества перед классическими программными и аппаратными его реализациями и позволяет избежать некоторых характерных для них проблем. Менеджер томов ZFS позволяет управлять томами и файловыми системами. Среди прочих возможностей следует отметить поддержку сжатия, дедупликации и шифрования.

Принципы организации. Файловые системы ZFS размещаются в пуле zpool, который состоит из виртуальных устройств vdev, каждое из которых представляет собой диск или группу дисков. Группа дисков может быть сконфигурирована в различных режимах RAID: stripe (аналог RAID0), mirror (аналог RAID1), raidz (аналог RAID5), raidz2 (аналог RAID6), raidz3 (аналог RAID5/6, допускается отказ трех дисков). В качестве виртуального устройства можно использовать блочное устройство или файл. Использование аппаратных RAID-массивов в качестве виртуальных устройств не рекомендуется. Вместо этого рекомендуется переключить контроллер в режим HBA и использовать каждый подключенный диск по отдельности. Пул может быть в любой момент расширен путем добавления в него новых виртуальных устройств. Вложенные (nested) виртуальные устройства не поддерживаются.

У каждого пула есть номер версии, который определяет поддерживаемые функциональные возможности. При установке новой версии ZFS, существующий пул может быть обновлен администратором для активации новых возможностей.

В одном пуле может быть размещено несколько файловых систем и томов zvol (виртуальных блочных устройств). Во время создания пула автоматически создается основная файловая система, которой доступно все свободное пространство. При необходимости, можно создать дополнительные файловые системы и ограничить их размер. Все файловые системы и тома в одном пуле разделяют общее свободное пространство.

В отличие от обычного программного или аппаратного RAID, файловой системе ZFS известно в каких блоках содержится полезная информация, поэтому при замене диска на него копируются только необходимые блоки, что уменьшает общее время восстановления.

Модель записи. При записи информации используются транзакции и принцип «copy-on-write» – новые данные никогда не записываются поверх старых, вместо этого выделяется новый блок, данные записываются в него, после чего обновляются ссылки в метаданных. Любая операция записи выполняется полностью или не выполняется совсем, что делает ненужной проверку целостности файловой системы после некорректного завершения работы и решает проблему «write hole¹». Для уменьшения накладных расходов отдельные транзакции объединяются в группы.

Сжатие данных. В настройках каждой файловой системы можно включить или выключить сжатие данных. Поддержка современного алгоритма LZ4 позволяет достигнуть хороших результатов при незначительном падении производительности. Сжатие может быть включено или выключено в любой момент – изменение этого параметра не влияет на уже сохраненные данные.

Дедупликация данных. При записи каждого блока вычисляется его контрольная сумма, поэтому ZFS может определить, когда одинаковый блок записывается несколько раз и записать его только однажды, а во всех остальных случаях использовать ссылки. Это позволяет экономить дисковое пространство, но требует накладных расходов и большего объема оперативной памяти.

Кэш чтения. В ZFS используется два уровня кэша чтения – ARC и L2ARC. ARC размещается в оперативной памяти и содержит наиболее часто используемые блоки. L2ARC – такой же кэш, который может быть расположен на

¹ Эта проблема характерна для уровней RAID, которые основаны на чередовании и четности – RAID5, RAID6 и т.д. [2].

отдельном накопителе (например, на SSD-диске).

Кэш записи. Для выполнения требований стандартов POSIX относительно синхронной записи ZFS использует специальный журнал записи ZIL (ZFS Intent Log). Когда приложение запрашивает синхронную запись, данные вначале помещаются в ZIL, затем асинхронно перемещаются на выделенное для них пространство в пуле. По умолчанию ZIL размещается в пуле с остальными данными, но при необходимости можно использовать отдельное высокопроизводительное устройство (например, SSD-диск). Выделенные для ZIL устройства называются SLOG (Separate Logging Device). Устройство SLOG может состоять из отдельного диска или группы дисков в режиме зеркалирования. В случае отказа устройства SLOG, ZFS размещает ZIL в пуле.

Advanced Format. В современных жестких дисках используются физические секторы размером 4 килобайта (вместо 512 байт). ZFS позволяет указать размер сектора на дисках при создании пула и добавлении новых виртуальных устройств в существующий пул. Данные размещаются на дисках таким образом, чтобы смещение блоков было согласовано с размером сектора.

Средства диагностики. Для проверки пулов системные администраторы могут использовать специальный механизм scrub. При его запуске все блоки с данными будут прочитаны и проверены, а обнаруженные ошибки исправлены. Эта проверка выполняется в режиме «он-лайн» и не требует приостановки работы.

Уведомления об ошибках (TLER/ERC). Одно из отличий промышленных жестких дисков от дисков пользовательского класса – возможность настройки уведомлений об ошибках. Когда у «обычного» жесткого диска не получается прочитать сектор, он будет производить повторные попытки неограниченно долгое время, при этом ничего не сообщая операционной системе. Это допустимо при использовании в домашних компьютерах, но недопустимо при подключении дисков к RAID-контроллерам, т.к. через несколько секунд RAID-контроллер посчитает диск неисправным и отключит его. В случае непосредственного подключения к серверу, операционная система будет бесконечно долго ждать завершения операции ввода/вывода. Жесткие диски промышленного класса позволяют установить тайм-аут, по истечении которого диск сообщает об ошибке ввода/вывода, что дает возможность операционной системе или RAID-контроллеру корректно ее обработать [3]. Для того чтобы ZFS имела возможность обрабатывать ошибки ввода/вывода необходимо установить тайм-аут TLER/ERC в конечное не большое значение.

Системные требования. Для корректной работы и оптимальной производительности рекомендуется использование 64-битной операционной системы и не менее 2 гигабайт оперативной памяти с возможностью коррекции ошибок (ECC). В случае использования функций сжатия и дедупликации требования к памяти значительно возрастают.

Ограничения и недостатки.

- Невозможность уменьшения размера пула;
- Невозможность расширения виртуального устройства raidz за счет добавления новых дисков в группу;
- Невозможность использования вложенных виртуальных устройств;
- Отсутствие механизма дефрагментации;
- Лицензионные ограничения препятствуют включению исходного кода ZFS в исходный код ядра Linux.

Реализация в Linux. Существует несколько различных проектов по портированию ZFS для работы в Linux. Наиболее перспективным считается «ZFS on Linux». В рамках этого проекта к настоящему времени реализованы все основные возможности ZFS (кроме шифрования) и поддержка пулов версии до 5000. Утверждается, что текущая версия достаточно стабильна для повседневного использования [4].

ZFS распространяется под лицензией CDDL, которая несовместима с GPL, поэтому исходный код ZFS невозможно включить в исходный код ядра Linux. Для обхода этого ограничения исходные коды поставляются в виде отдельных пакетов, а их автоматическая сборка и установка осуществляется с помощью DKMS (Dynamic Kernel Module Support).

Программный интерфейс (API) ядра Linux отличается от того что используется в Solaris, поэтому для работы ZFS в Linux требуется установка модуля ядра SPL (Solaris Porting Layer), который транслирует системные вызовы специфичные для Solaris в соответствующие системные вызовы Linux, что позволило избежать значительных изменений кода ZFS при портировании.

Установка ZFS в CentOS Linux. В операционной системе CentOS Linux используется менеджер пакетов yum, который осуществляет автоматическую установку rpm-пакетов. Разработчики ZFS on Linux предоставляют rpm пакеты ZFS и SPL, для установки которых необходимо добавить в систему репозитории ZOL (ZFS on Linux) и EPEL (Extra Packages for Enterprise Linux) [5]. Добавление репозитория и установка ZFS осуществляется командами:

```
$ sudo yum localinstall --nogpgcheck https://download.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
```

```
$ sudo yum localinstall --nogpgcheck http://archive.zfsonlinux.org/epel/zfs-release.el6.noarch.rpm
```



```
$ sudo yum install kernel-devel zfs
```

Последняя команда установит пакеты необходимые для сборки ZFS и SPL из исходных кодов: dkms – система автоматической сборки модулей ядра, spl-dkms – исходные коды SPL, zfs-dkms – исходные коды zfs, spl и zfs – пакеты с утилитами командной строки, kernel-devel – заголовки ядра, необходимые для сборки модулей. В процессе установки пакетов автоматически запустится процесс DKMS, который произведет сборку модулей ядра и генерацию начального корневого диска системы (initrd).

Основные утилиты командной строки – zpool и zfs. Первая позволяет выполнять все операции по работе с пулами, вторая управляет томами и файловыми системами. Например:

```
# zpool create ... // создание пула
# zpool list // отображение списка пулов
# zpool status // отображение состояния каждого пула
```

Рассмотрим пример вывода информации о состоянии пула:

```
[root@video ~]# zpool status
pool: lms2
state: ONLINE
scan: scrub repaired 0 in 1h27m with 0 errors on Sun
May 3 12:11:32 2015
config:
```

NAME	STATE	READ	WRITE	CKSUM
lms2	ONLINE	0	0	0
raidz2-0	ONLINE	0	0	0
lms2_dev1	ONLINE	0	0	0
lms2_dev2	ONLINE	0	0	0
lms2_dev3	ONLINE	0	0	0
lms2_dev4	ONLINE	0	0	0
lms2_dev5	ONLINE	0	0	0
lms2_dev6	ONLINE	0	0	0
lms2_dev7	ONLINE	0	0	0
lms2_dev8	ONLINE	0	0	0

```
errors: No known data errors
```

В этом примере выведена информация о пуле, который называется «lms2», состоит из одного виртуального устройства типа raidz2 (RAID6), которое в свою очередь представляет собой группу из 8 дисков. При последней проверке, которая была выполнена 3 мая 2015 г. в 12:11:32, ошибок не обнаружено. В случае наличия ошибок, в информации о состоянии пула выведется подробная информация о них, в том числе способы их исправления и список затронутых файлов.

Подробную информацию о параметрах утилит `zpool` и `zfs` можно узнать в системной справке (`man`).

Особенности Linux-версии ZFS:

- Использование несогласованных версий пакетов `spl` и `zfs` может привести к ошибкам, номера версий должны совпадать. DKMS и `yum` позволяют сделать процесс их обновления одновременным;
- Размер блока тома `zvol` должен совпадать с размером страницы памяти при использовании томов `zvol` в качестве SWAP;
- Нестабильная работа ZFS в 32-разрядных системах из-за особенностей работы с памятью в Solaris;
- Не все загрузчики Linux поддерживают ZFS в качестве корневой файловой системы;
- Монтирование томов ZFS осуществляется автоматически, без использования `fstab`, но при желании можно включить режим совместимости.

Сравнение основных характеристик ZFS и других часто используемых в Linux файловых систем показано в табл. 1.

Таблица 1 – Сравнение ZFS с другими файловыми системами

Характеристики	ext4 [6]	xfс [7]	reiserfs [8]	zfs
Макс. размер ФС	1 ЭБ	8 ЭБ	16 ТБ	256 ЗБ
Макс. размер файла	16 ТБ	8 ЭБ	8 ТБ	16 ЭБ
Прозрачное сжатие	–	–	–	+
Прозрачное шифрование	+	–	–	+
Прозрачная дедупликация	–	–	–	+
POSIX ACL	+	+	–	+
Встроенный менеджер томов	–	–	–	+
Встроенная избыточность	–	–	–	+
Увеличение размера ФС	+	+	+	+
Уменьшение размера ФС	+	–	+	–
Код ФС включен в код ядра Linux	+	+	+	–
Код ФС доступен	+	+	+	+
Работа на 32-битных ОС	+	+	+	–
Низкие требования к ОЗУ	+	+	+	–
Горячая замена дисков	–	–	–	+
«On-Line»-проверка ФС	–	–	–	+
Дефрагментация	+	+	–	–
Снимки файловой системы	–	–	–	+
Поддержка 4К секторов	+	+	+	+

Не все функции оригинальной ZFS могут быть реализованы в Linux из-за ее тесной интеграции с Solaris. Такие возможности как RAID, менеджер томов и снимки реализованы в Linux в виде отдельных программных пакетов (LVM, MD), но из-за отсутствия их интеграции с файловыми системами они не могут обеспечить такой же уровень обнаружения и исправления ошибок, что и ZFS.

Вывод. Файловая система ZFS предъявляет повышенные требования к ресурсам и характеристикам используемого оборудования, предоставляя взамен богатые возможности. Ее использование оправдано в случае наличия повышенных требований к сохранности данных и возможности приобретения дорогостоящего оборудования. В случае если повышенная надежность не требуется или при ограниченных системных ресурсах, более оправдано использование классических файловых систем.

ЛИТЕРАТУРА

1. ZFS - Wikipedia, the free encyclopedia [Электронный ресурс]. – Режим доступа: URL: <https://en.wikipedia.org/wiki/ZFS> – название с экрана.
2. Closing the RAID5 write hole [Электронный ресурс]. – Режим доступа: URL: <http://neil.brown.name/blog/20110614101708> – название с экрана.
3. Error recovery control - Wikipedia, the free encyclopedia [Электронный ресурс]. – Режим доступа: URL: https://en.wikipedia.org/wiki/Error_recovery_control – название с экрана.
4. The State of ZFS on Linux [Электронный ресурс]. – Режим доступа: URL: <https://clusterhq.com/blog/state-zfs-on-linux/> – название с экрана.
5. ZFS on Linux [Электронный ресурс]. – Режим доступа: URL: <http://zfsonlinux.org/> – название с экрана.
6. ext4 - Wikipedia, the free encyclopedia [Электронный ресурс]. – Режим доступа: URL: <https://en.wikipedia.org/wiki/Ext4> – название с экрана.
7. XFS - Wikipedia, the free encyclopedia [Электронный ресурс]. – Режим доступа: URL: <https://en.wikipedia.org/wiki/XFS> – название с экрана.
8. ReiserFS - Wikipedia, the free encyclopedia [Электронный ресурс]. – Режим доступа: URL: <https://en.wikipedia.org/wiki/ReiserFS> – название с экрана.

УДК 378.147.39.004

ОБ ИСПОЛЬЗОВАНИИ ПАКЕТА «STATISTIKA» В ПРЕПОДАВАНИИ КУРСА «МАТЕМАТИЧЕСКИЕ МЕТОДЫ ОБРАБОТКИ РЕЗУЛЬТАТОВ ЭКСПЕРИМЕНТА» ДЛЯ СТУДЕНТОВ ТЕХНИЧЕСКОГО ВУЗА

К.Ф. Габдрахманова¹

¹кандидат педагогических наук, доцент кафедры информационных технологий, математики и естественных наук, филиал ФГБОУ ВПО «Уфимский государственный технический университет» в г. Октябрьском, г. Октябрьский, Россия, e-mail: klara47@mail.ru