СОЗДАНИЕ ОТКАЗОУСТОЙЧИВОЙ СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ ДЛЯ ВИРТУАЛЬНОЙ ИНФРАСТРУКТУРЫ НА ОСНОВЕ СВОБОДНО РАСПРОСТРАНЯЕМОГО ПРОГРАММНОГО ОБНСПЕЧЕНИЯ

А. А. Крыжановский, А. В. Шишкин, К. В. Павлов, И. Л. Бондарь, М. Ю. Осипов

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

Для обеспечения работы различных корпоративных сервисов и служб во ВНИИЭФ широко используется виртуализация с использованием систем хранения данных (СХД), реализованных на базе коммерческого программного обеспечения (ПО) NexentaStor.

В соответствии с общей тенденцией предприятие активно занимается вопросами импортозамещения, т. е. перехода на отечественное программное обеспечение, либо на свободно распространяемое ПО с открытым исходным кодом.

Учитывая тот факт, что NexentaStor в 2015 году попала в список продуктов, запрещенных к продаже предприятиям ОПК, одним из первых продуктов, которые решено заменить, выбран этот. Соответственно, началась работа по созданию СХД на базе свободно распространяемой ОС и продуктов с открытым исходным кодом. В качестве ОС выбрано семейство Линукс, для реализации функционала СХД решено применить возможности, предоставляемые файловой системой ZFS. Вторая задача, которая решалась одновременно с выбором нового решения для построения СХД, получение производительности, сопоставимой с производительностью СХД на коммерческом ПО NexentaStor.

Краткая информация по продукту NexentaStor. Это ПО для построения универсальных систем хранения данных, доступное для платформы x86. Продукт основан на файловой системе ZFS. Ключевым преимуществом NexentaStor является встроенный механизм устранения дублирования (дедупликации) данных, позволяющий значительно сократить физический объем данных, записываемых в хранилища, без приобретения дополнительных модулей.

Другими особенностями продукта являются:

– неограниченный размер файловой системы (поправка: для бесплатной версии Community Edition размер хранилища ограничен 18Тб) и неограниченное количество моментальных снимков;

– виртуализация дисков;

– блочное зеркалирование с непрерывной защитой данных (CDP);

– синхронная и асинхронная репликация данных;

– механизм Thin Provisioning для плавного выделения места под хранилище;

– поддержка томов однократной записи – WORM;

– поддержка протоколов NFS/CIFS/RSYNC/FTP;

– поддержка подключения FC/iSCSI/SAS;

– поддержка отказоустойчивого кластера;

– поддержка гибридных томов с дисками SSD.

ZFS (Zettabyte File System) – файловая система, разработанная в компании Sun Microsystems для операционной системы Solaris и, впоследствии, перенесенная на ряд других операционных систем семейства Linux.

ZFS обладает следующими основными преимуществами:

– встроенные механизмы для работы с дисками организации RAID;

– отсутствие привязки к оборудованию;

– автоматическое исключение из работы вышедших из строя дисков, с включением в работу резервных дисков, исправление ошибок и перестроение RAID;

– поддержка огромных размеров томов, файлов, пулов, а также легкая масштабируемость хранилища;

– быстрое и удобное администрирование ZFS Pool;

– увеличение скорости хранилища при увеличении кол-ва дисков;

– дедупликация и сжатие данных.

К минусам данной системы можно отнести следующее:

– высокие требования к ресурсам ЦП и ОЗУ;

– необходимость использования ОЗУ с контролем четности (ECC);

– отсутствие возможности дефрагментации.

При разработке отказоустойчивой СХД выбор ОС осуществлялся на основании соответствия следующим требованиям:

– ОС Российской разработки или свободно распространяемая ОС с открытым исходным кодом;

– поддержка файловой системы ZFS на уровне ядра;

– обеспечить совместимость с существующими во ВНИИЭФ системами хранения данных;

– поддержка широкого спектра аппаратного обеспечения (драйвера устройств), для совместимости с используемым во ВННИЭФ коммутационным оборудованием;

– регулярное обновление версии программного продукта.

Рассматривались следующие ОС: Ubuntu, OpenSolaris, FreeBSD, Astra Linux, Alt Linux и другие. В результате выявлено, что некоторые ОС перестали поддерживаться, другие не имеют драйверов под применяемое коммутационное оборудование, третьи не поддерживают ZFS на уровне ядра.

В результате исследовательской работы в качестве ОС была выбрана Ubuntu Server 16.04.

Для отработки решения была создана виртуальная инфраструктура, состоящая из одного гипервизора и двух дисковых массивов. Один дисковый массив на основе NexentaStor, второй на основе Ubuntu Server 16.04. Основные технические характеристики серверов представлены в табл. 1.

Таблица 1

Технические характеристики серверов

	Гипервизор	Дисковые массивы
Процессор	1 x E5-2670 2.6 ГГц	2 x X5650 2.67ГГц
Оперативная память	32ГБ	48ГБ
Кол-во жестких дисков	2	36

В рамках исследования проводилось тестирование производительности СХД. При проведении тестирования СХД было заложено два основных условия:

– получение максимального объема полезного дискового пространства;

– получение максимальной производительности СХД.

Для получения максимального объема в файловой системе ZFS был создан пул на основе raidz1 (аналог RAID 5). Параметры пула указаны в табл. 2.

Таблица 2

Настройки параметров файловой системы ZFS

Параметр	Значение
Кол-во дисков	36
Тип RAID	raidz1 (одинарный контроль четности)
Разбиение дисков по группам	8 групп по 4 диска
Горячий резерв	4 диска
Сжатие	lz4
Основное кеширование	Только метаданные
Вторичное кеширование	Выключено

Хранение данных системы виртуализации имеет три уровня: аппаратная составляющая СХД (контроллеры жестких дисков), программный RAID (ZFS) и файловая система виртуальных машин (VMFS).

При проведении тестирования производительности СХД на уровне программного RAID использовались два различных размера блока данных: 8КБ (по умолчанию для ZFS) и 64КБ. Причина такого выбора обусловлена желанием сравнить производительность СХД при настройках по умолчанию и с одинаковыми размерами блоков данных на всех уровнях хранения данных.

Тестирование производительности СХД проводилось с помощью пакета fio со следующими параметрами:

1) Размер блока: 8, 16, 32, 64 и 128 КБ.

2) Метод тестирования: последовательное чтение, последовательная запись, произвольное чтение, произвольная запись, произвольные чтение/запись.

3) Время одного теста – 50 минут.

В результате проведения тестирования были получены данные, представленные в табл. 3. Сравнительные диаграммы представлены на рис. 1.

Таблица 3

Результаты тестирования СХД

Размер блока, КБ	Тип теста	Ubuntu (8КБ), МБ/сек	Nexenta (8КБ), МБ/сек	Ubuntu (64КБ), МБ/сек	Nexenta (64КБ), МБ/сек
8	последовательное чтение	1091,20	1069,38	786,40	770,67
	последовательное запись	431,26	439,88	348,67	355,64
	произвольное чтение	1035,80	1025,44	767,49	759,82
	произвольная запись	30,02	30,17	22,63	22,74
	произвольное чтение/запись	28,93	29,02	20,08	20,14
16	последовательное чтение	712,60	698,35	683,55	669,88
	последовательное запись	285,41	291,12	529,13	539,72
	произвольное чтение	1560,10	1544,50	1422,90	1408,67
	произвольная запись	53,09	53,36	48,68	48,92
	произвольное чтение/запись	51,58	51,73	39,38	39,49
32	последовательное чтение	876,74	859,20	985,15	965,45
	последовательное запись	290,55	296,36	638,73	651,51
	произвольное чтение	1971,70	1951,98	2421,20	2396,99
	произвольная запись	94,22	94,69	93,54	94,00
	произвольное чтение/запись	93,72	94,00	80,24	80,48

Размер блока, КБ	Тип теста	Ubuntu (8КБ), МБ/сек	Nexenta (8КБ), МБ/сек	Ubuntu (64КБ), МБ/сек	Nexenta (64КБ), МБ/сек
64	последовательное чтение	1016,40	996,07	1625,70	1593,19
	последовательное запись	641,42	654,25	665,85	679,16
	произвольное чтение	2263,90	2241,26	3904,20	3865,16
	произвольная запись	153,56	154,33	183,46	184,38
	произвольное чтение/запись	153,95	154,41	159,68	160,16
128	последовательное чтение	1111,70	1089,47	1427,90	1399,34
	последовательное запись	342,14	348,98	846,02	862,94
	произвольное чтение	2351,60	2328,08	4331,60	4288,28
	произвольная запись	236,01	237,19	356,51	358,30
	произвольное чтение/запись	237,14	237,85	278,29	279,12

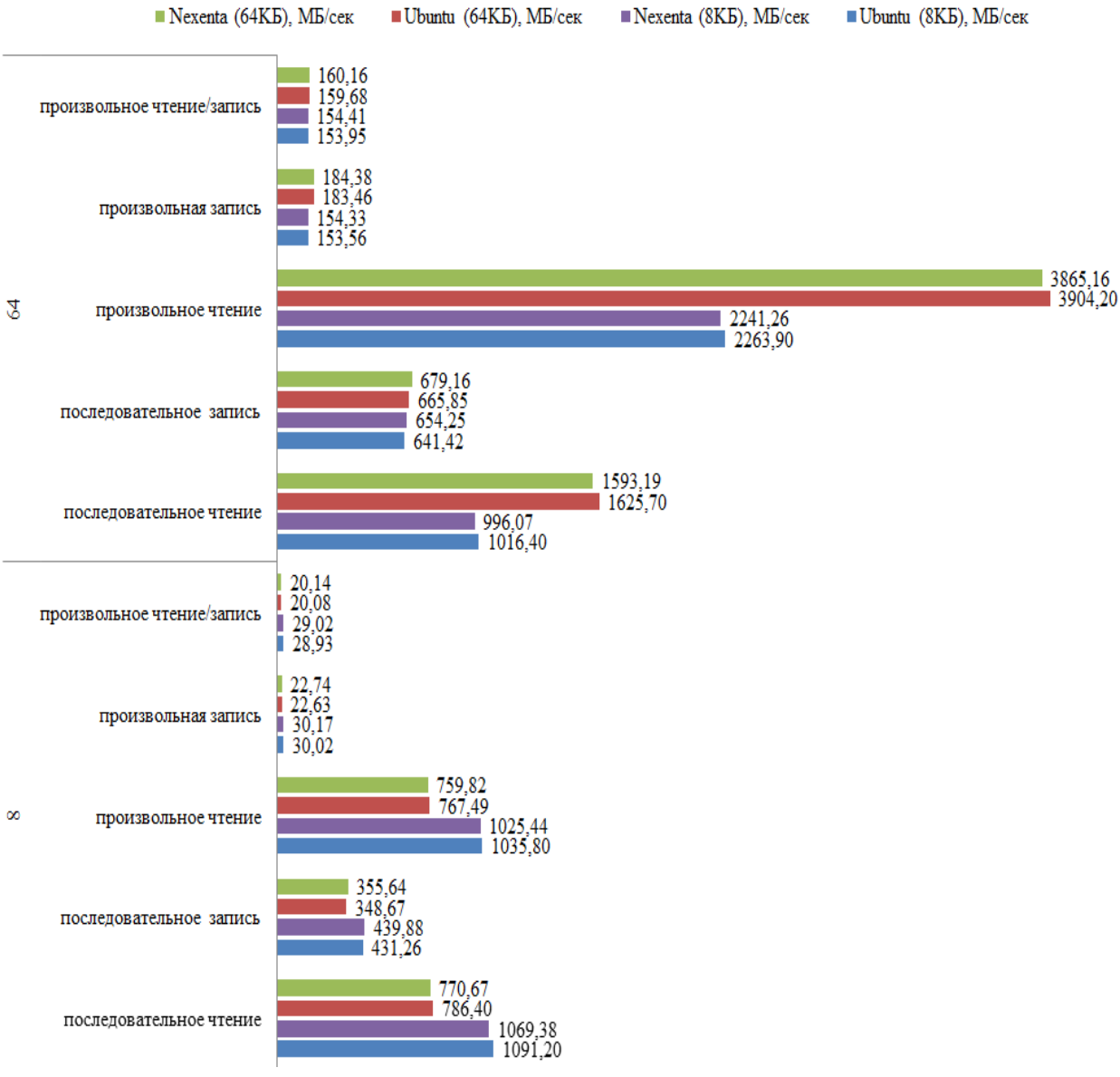

Рис. 1. Результаты тестирования на блоках 8, 64 КБ

Анализ полученных результатов тестов показал, что производительности СХД на основе Ubuntu и NexentaStor практически идентичны, разница составляет не более 1 %.

Для получения максимальной производительности файловой системы ZFS был создан пул на основе mirror (аналог RAID10). Параметры пула указаны в табл. 4.

Таблица 4

Настройки параметров файловой системы ZFS

Параметр	Значение
Кол-во дисков	36
Тип RAID	mirror
Разбиение дисков по группам	17 групп по 2 диска
Горячий резерв	2 диска
Сжатие	lz4
Основное кеширование	Только метаданные
Вторичное кеширование	Выключено

Тестирование проводилось аналогичным способом. Результаты теста приведены в табл. 5. Сравнительные диаграммы представлены на рис. 2.

Таблица 5

Результаты тестирования raidzи mirror

Размер блока, КБ	**Тип теста**	**Ubuntu (64КБ) raidz, МБ/сек**	**Ubuntu (64КБ) mirror, МБ/сек**	**Nexenta (64КБ) mirror, МБ/сек**
8КБ	последовательное чтение	786,40	838,99	822,21
	последовательное запись	348,67	537,60	548,35
	произвольное чтение	767,49	779,83	772,03
	произвольная запись	22,63	113,49	114,06
	произвольное чтение/запись	20,08	91,05	91,32
16КБ	последовательное чтение	683,55	893,06	875,20
	последовательное запись	529,13	818,55	834,92
	произвольное чтение	1422,90	1425,50	1411,25
	произвольная запись	48,68	222,19	223,30
	произвольное чтение/запись	39,38	176,80	177,33
32КБ	последовательное чтение	985,15	1435,40	1406,69
	последовательное запись	638,73	1068,40	1089,77
	произвольное чтение	2421,20	2531,20	2505,89
	произвольная запись	93,54	423,41	425,52
	произвольное чтение/запись	80,24	334,33	335,33
64КБ	последовательное чтение	1625,70	2164,90	2121,60
	последовательное запись	665,85	1383,70	1411,37
	произвольное чтение	3904,20	3907,50	3868,43
	произвольная запись	183,46	933,77	938,44
	произвольное чтение/запись	159,68	717,75	719,90
128КБ	последовательное чтение	1427,90	2470,10	2420,70
	последовательное запись	846,02	1494,90	1524,80
	произвольное чтение	4331,60	4303,90	4260,86
	произвольная запись	356,51	1182,50	1188,41
	произвольное чтение/запись	278,29	901,11	903,81

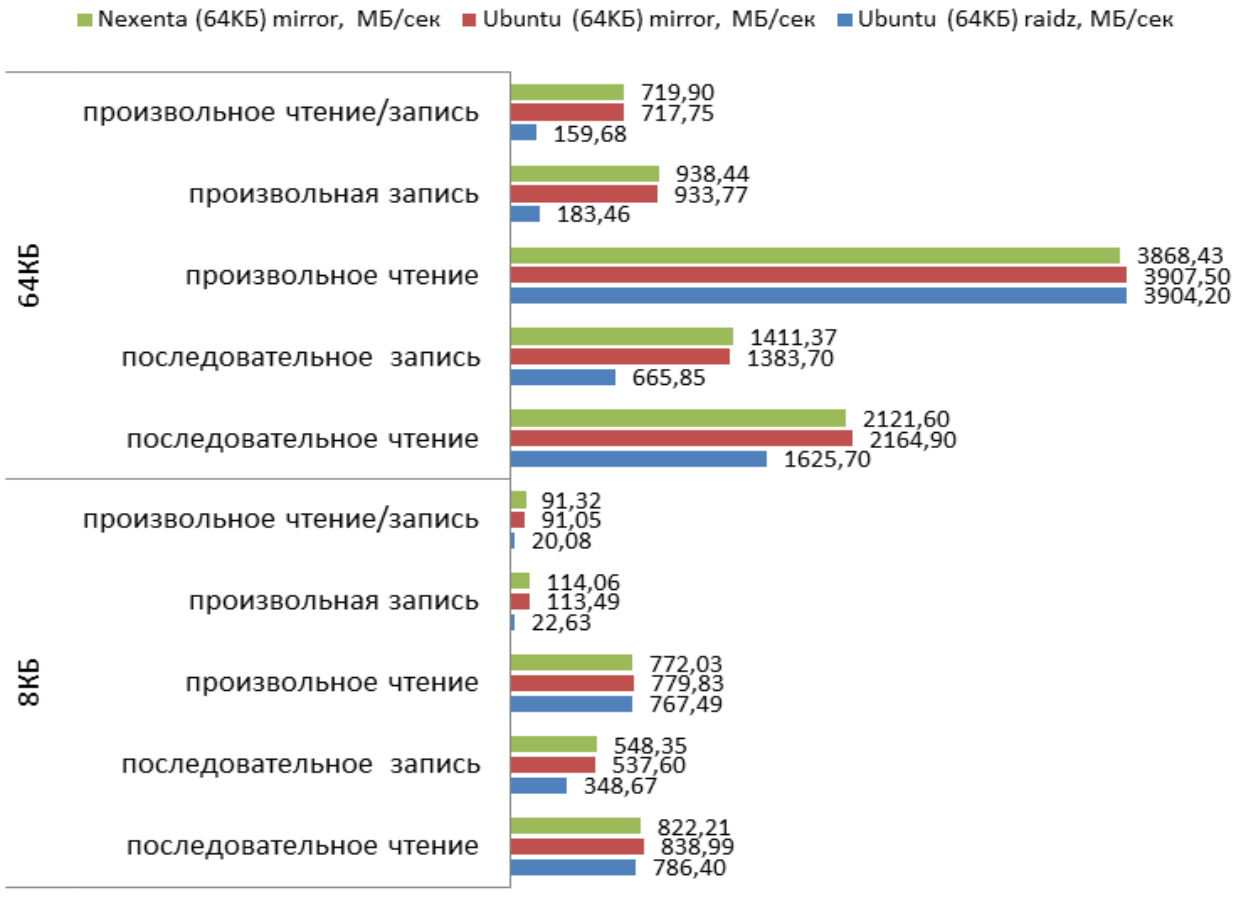

Рис. 2. Результаты тестирования raidzи mirror

Результаты теста показали, что производительность СХД на основе Ubuntu и NexentaStor при использовании emirror практически одинакова, при этом скорость записи данных в несколько раз выше, чем при использовании raidz. Это может быть полезным для систем, которым требуется высокая скорость записи данных.

Для организации доступа гипервизоров к виртуальным разделам СХД (LUN) на основе Ubuntu был использован SCST (generic SCSI target subsystem for Linux). SCST позволяет работать с любыми ссылками, которые поддерживают обмен данными по стандарту SCSI:

– iSCSI;
– Fibre Channel;
– FCoE;
– SAS;
– InfiniBand (SRP);
– Wide (parallel) SCSI;
– и т. д.

В результате проделанной работы была создана отказоустойчивая СХД на программных продуктах с открытым исходным кодом, которая по производительности сравнима с коммерческим решением на основе NexentaStor. Данное решение совместимо с системами, которые используют ПО NexentaStor. Оно является универсальным и позволяет осуществлять переход на него при модернизации существующих и создании новых систем. На основе тестовой СХД был создан шаблон, со встроенным механизмом SCST, включающий в себя сценарии автоматизации процесса настройки СХД. Шаблон позволяет в короткие сроки создавать новые СХД на имеющемся оборудовании.

На данный момент на основе полученного решения организован полигон виртуализации с использованием технологии VDI, в состав которого входит 11 гипервизоров и две СХД на основе Ubuntu, обеспечивающий работу более 50 виртуальных машин. Это же решение будет использоваться в подсистеме виртуализации корпоративных сетевых служб и сервисов автоматизированных систем ВНИИЭФ.