

PROCESS STRATEGY SERVER PLANNING DEVICE SERVICE VOLUME TYPE MEDIA MONITORING TECHNOLOGY SET ANALYSIS FULL BACKUP IMPLEMENTATION SYSTEM SCHEDULING RESTORE SOLUTIONS NORMAL IMAGE INCREMENTAL SUPPORT DATA TAPE RECOVERY SOFTWARE REPORTING DIFFERENT MANAGEMENT



Алексей Бережной

# Сохранение данных: теория и практика

Алексей Бережной

# Сохранение данных: теория и практика



Москва, 2016

**УДК 004.056**  
**ББК 32.97**  
**Б48**

Бережной А. Н.  
Б48 Сохранение данных: теория и практика. – М.: ДМК Пресс, 2016. – 317 с.: ил.

**ISBN 978-5-97060-185-3**

В книге рассказано о том, что необходимо для превращения введенной ИТ-инфраструктуры в отказоустойчивую систему. Какие существуют способы защиты информации, какие этапы необходимо пройти при проектировании, как создать Disaster Recovery Plan (план полного восстановления), как создать эффективную систему резервного копирования, как организовать защиту перемещаемых данных – обо всем об этом и о множестве других полезных вещей вы узнаете, прочтя эту книгу.

Большое внимание уделяется связи информационных систем и бизнеса, выстраиванию эффективных, экономически оправданных и легких в освоении систем.

Изложение материала построено по принципу «теория + практика», автор не только приводит информацию по основополагающим вопросам, но и щедро делится своим богатым опытом.

Издание предназначено для системных архитекторов, инженеров, администраторов, разработчиков отказоустойчивых систем и систем резервного копирования, руководителей ИТ-подразделений, ИТ-менеджеров, специалистов по продажам, а также преподавателей и студентов технических вузов.

**УДК 004.056**  
**ББК 32.97**

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

**ISBN 978-5-97060-185-3**

© Бережной А. Н., 2016  
© Издание, оформление, ДМК Пресс, 2016

*Посвящается моей дочери Марии*

# Содержание

<b>Как и почему я написал эту книгу .....</b>	<b>14</b>
<b>Как устроена эта книга .....</b>	<b>17</b>
<b>Благодарности .....</b>	<b>21</b>
 <b>Часть I. Краеугольные камни .....</b>	<b>22</b>
<b>Глава 1. О реальном месте ИТ в современном бизнесе....</b>	<b>23</b>
1.1. Зачем нужны информационные технологии.....	23
1.1.1. Почему нужно читать эту главу? .....	24
1.1.2. Пример работы торговой фирмы .....	24
1.1.3. И вот приходит эра информационных технологий.....	28
1.2. Ахиллесова пята современного бизнеса.....	30
1.3. О понимании политического момента.....	31
1.4. Заключение.....	32
 <b>Глава 2. Популярная терминология .....</b>	<b>33</b>
2.1. Начнем с изучения специфики бизнеса .....	33
2.1.1. Бизнес-процесс .....	33
2.1.2. Мера терпения.....	34
2.2. О факторах риска и мерах предосторожности.....	36
2.3. Общие термины обеспечения отказоустойчивости .....	37
2.3.1. Business Continuity Planning (BCP) .....	37
2.3.2. Disaster Recovery .....	38
2.3.3. Disaster Recovery Plan (DRP) .....	38
2.4. Что такое центр обработки данных и чем он отличается от серверной комнаты.....	38
2.4.1. Определение ЦОД .....	38
2.4.2. Компоненты ЦОД .....	39
2.5. Уровни Disaster Recovery .....	39
2.6. Специальные термины для резервного копирования .....	41
2.6.1. Backup window .....	42
2.6.2. Частота выполнения резервных копий.....	42
2.6.3. Глубина хранения .....	43
2.6.4. Off-site .....	43
2.7. Системы высокой доступности .....	44

2.7.1. High Availability cluster, HA-cluster .....	44
2.7.2. Primary-Standby .....	45
2.8. Заключение.....	45

### **Глава 3. Избыточность на службе отказоустойчивости ....46**

3.1. Об этой главе.....	46
3.2. Обозначения и расчеты .....	47
3.3. Подробнее о RAID-массивах.....	47
3.3.1. RAID-0.....	47
3.3.2. RAID-1.....	48
3.3.3. RAID-1E.....	49
3.3.4. RAID-5.....	50
3.3.5. RAID-6.....	51
3.3.6. Составные, или сложные, массивы .....	52
3.3.7. RAID-10 .....	52
3.4. Программный или аппаратный RAID? .....	53
3.4.1. Программный RAID .....	53
3.4.2. На основе аппаратного RAID-контроллера .....	54
3.4.3. Программный или аппаратный RAID – что выбрать?.....	55
3.4.4. Технология RAID+.....	55
3.5. Методы защиты информации от ошибок при записи .....	57
3.5.1. Копирование при записи (Copy-On-Write) .....	57
3.5.2. Перенаправление при записи (Redirect-On-Write) .....	59
3.6. Заключение.....	60

### **Глава 4. Архитектура систем резервного копирования .....61**

4.1. Топология резервного копирования .....	61
4.1.1. Децентрализованная топология .....	62
4.1.2. Централизованная топология .....	63
4.1.3. Псевдоцентрализованная топология.....	63
4.1.4. Смешанная топология .....	65
4.2. Уровни архитектуры систем резервного копирования .....	66
4.2.1. Архитектура для небольших инфраструктур.....	66
4.2.2. Архитектура системы резервного копирования крупного предприятия.....	67
4.2.3. Архитектура для среднего уровня.....	70
4.3. Проблема выбора при организации системы резервного копирования.....	71

4.4. Рекомендации по выбору системы резервного копирования.....	72
4.5. Заключение.....	73

## **Вопросы к первой части для закрепления материала.....74**

## **Часть II. Резервное копирование как оно есть.....75**

### **Глава 5. Виды резервного копирования .....76**

5.1. Зачем все так усложнять, если нужно просто сохранить данные?.....	76
5.2. Планирование работы системы резервного копирования.....	77
5.2.1. Глубина хранения данных.....	77
5.2.2. Ограничение времени создания резервной копии.....	77
5.2.3. Ограничение объема.....	78
5.2.4. Ограничение времени восстановления.....	78
5.3. Оптимизация резервного копирования.....	79
5.4. Полное резервное копирование (Full backup).....	79
5.5. Неполное резервное копирование и его виды.....	81
5.5.1. Инкрементальное копирование (Incremental backup).....	81
5.5.2. Инкрементальный парадокс.....	83
5.5.3. Дифференциальное резервное копирование (Differential backup).....	85
5.5.4. Обратное инкрементальное резервное копирование (Reversed incremental backup).....	86
5.6. Добавочная, или дополнительная, резервная копия.....	88
5.7. Заключение.....	89

### **Глава 6. Схемы ротации носителей .....90**

6.1. Схемы ротации.....	90
6.2. А для чего вообще необходимы схемы ротации?.....	90
6.3. Когда схема ротации не используется.....	91
6.4. Полное резервное копирование.....	92
6.5. Самая популярная схема ротации – «отец–сын».....	92
6.6. «Дед–отец–сын» как классическая схема ротации.....	96
6.7. Операция «слияние», или Как пожертвовать историей на благо экономики.....	99
6.8. Постоянное резервное копирование, или Continuous backup.....	102
6.9. Заключение.....	104

## **Глава 7. Цели и технологии резервного копирования. О различиях в технологии резервного копирования..... 105**

7.1. Дуализм систем резервного копирования .....	105
7.1.1. Подробнее о задачах резервного копирования .....	106
7.2. Технология резервного копирования .....	109
7.2.1. Технология файлового копирования .....	109
7.2.2. Технология блочного копирования .....	110
7.2.3. В каких случаях применяют ту или иную технологию?..	113
7.3. Заключение.....	115

## **Глава 8. Резервное копирование виртуальных систем..... 116**

8.1. Предисловие.....	116
8.2. Вариант 1 – виртуальная машина создает свою копию самостоятельно .....	117
8.2.1. Преимущества данного метода .....	118
8.2.2. Ограничения данного метода .....	118
8.3. Копирование виртуальной машины целиком .....	118
8.3.1. Механизмы для копирования виртуальных машин целиком.....	119
8.3.2. Особенности резервного копирования виртуальной машины с использованием снапшота.....	121
8.4. Виртуальный шторм и его последствия.....	125
8.5. Заключение.....	127

## **Вопросы ко второй части для закрепления материала ..... 128**

## **Часть III. Оборудование для систем резервного копирования ..... 129**

## **Глава 9. Системы хранения резервных копий – единство и борьба противоположностей ..... 130**

9.1. Предисловие.....	130
9.2. Что из себя представляют дисковые хранилища .....	131
9.2.1. Преимущества дисковых подсистем.....	131
9.2.2. Общие черты для всех дисковых хранилищ .....	132
9.2.3. Разнообразие дисковых подсистем .....	133
9.2.4. Сетевая система хранения данных – NAS .....	134

9.2.5. Сеть хранения данных – SAN .....	135
9.3. Ленточные системы хранения .....	136
9.3.1. Преимущества ленточных систем хранения.....	136
9.3.2. Ограничения использования ленточных накопителей....	137
9.4. Диски и ленты работают вместе – технология D2D2T, или «Disk-to-Disk-to-Tape» .....	140
9.5. Виртуальные ленточные хранилища – Virtual Tape Library .....	141
9.6. Заключение.....	142

## **Глава 10. Магнитные ленты и вездесущий LTO ..... 143**

10.1. Предисловие .....	143
10.2. Различные устройства для записи на ленту .....	144
10.2.1. Одиночный ленточный накопитель .....	144
10.2.2. Автозагрузчик.....	145
10.2.3. Ленточная библиотека .....	146
10.2.4. Рекомендации по применению тех или иных устройств.....	146
10.3. Стандарты хранения на ленточных накопителях. Вездесущий LTO .....	147
10.3.1. Характеристики различных поколений LTO .....	148
10.3.2. Сравнительное описание поколений LTO .....	150
10.4. Поддержка на уровне файловых систем.....	152
10.5. Загадочный чип LTO-CM.....	152
10.6. Дополнительно о шифровании данных на ленте.....	152
10.7. Заключение .....	154

## **Глава 11. Дисковая система хранения на базе NAS ..... 155**

11.1. Описание принципов работы NAS.....	155
11.2. Работа с NAS на примере NETGEAR ReadyNAS Pro.....	156
11.2.1. О выборе RAID .....	156
11.2.2. Первоначальная настройка.....	157
11.2.3. Учетные записи пользователей .....	162
11.2.4. Настройка доступа по протоколу CIFS.....	164
11.3. Continuous Backup с рабочих станций на сетевое хранилище .....	166
11.4. Настройка резервного копирования содержимого хранилища .....	170
11.5. Заключение .....	173

## **Глава 12. Сеть обмена данными и резервное копирование ..... 174**

12.1. Страшная история со счастливым концом .....	174
12.2. Как изначально предотвратить потерю управляемости.....	176
12.3. Топология LAN-free и Server-free .....	177
12.3.1. Краткое описание топологии LAN-free .....	177
12.3.2. Краткое описание топологии Server-free .....	178
12.4. Использование дополнительной сети передачи данных.....	179
12.5. Организация дополнительной сети для управления серверами .....	181
12.6. Заключение .....	183

## **Вопросы к третьей части для закрепления материала... 184**

## **Часть IV. «Что нам стоит дом построить, нарисуем – будем жить». Проектирование отказоустойчивых систем..... 185**

## **Глава 13, которой быть не должно. Что мешает создать отказоустойчивую инфраструктуру? ..... 186**

13.1. Зачем об этом писать? .....	186
13.2. Дуализм подходов к работе IT.....	186
13.2.1. «Архитекторы».....	187
13.2.1. «Шаманы».....	188
13.2.3. Рекомендации .....	190
13.3. Проблемы перехода.....	190
13.4. «Надкусывание».....	192
13.5. «Дожигание».....	194
13.6. Заключение .....	197

## **Глава 14. Десять шагов к вершине. Процесс проектирования отказоустойчивых систем ..... 198**

14.1. Аудит системы и бизнес-процессов .....	199
14.2. Составление технического задания .....	202
14.3. Технический проект .....	206
14.4. Пилотный проект .....	210
14.5. Подготовка рабочей документации .....	212
14.6. Внедрение системы .....	214
14.7. Подготовка исполнительной документации .....	216

14.8. Приемо-сдаточные испытания .....	217
14.9. Передача объекта заказчику .....	218
14.10. Опытная эксплуатация и техническое обслуживание объекта .....	218
14.11. Заключение .....	219

## **Глава 15. «Возьми с собой ключи от моих дверей». О безопасности резервного копирования ..... 221**

15.1. Почему вопросы защиты информации будут всегда актуальны .....	221
15.2. Избыток так же плох, как и недостаток .....	222
15.3. Разделение ролей при работе с резервными копиями .....	224
15.4. Классификация и структура методов обеспечения безопасности при резервном копировании .....	225
15.5. Ограничение возможности копирования данных .....	225
15.5.1. Ограничение доступа к системе резервного копирования .....	226
15.5.2. Ограничение доступа к ресурсам, которые могут быть скопированы .....	227
15.5.3. Ограничение доступа через учетную запись .....	227
15.5.4. Ограничение доступа общего характера .....	228
15.6. Ограничения возможности использования резервных копий .....	229
15.6.1. Ограничение физического доступа к резервным копиям .....	230
15.6.2. Ограничение возможности применения резервных копий .....	231
15.7. Заключение .....	232

## **Глава 16. Жизнь после смерти, или Как восстановить систему после мировой катастрофы ..... 233**

16.1. Предисловие .....	233
16.2. Несколько слов о резервном копировании .....	234
16.3. Создание Disaster recovery plan .....	235
16.3.1. Для чего нужно писать Disaster Recovery Plan? .....	235
16.3.2. Что такое Disaster Recovery Plan .....	235
16.3.3. Описание бизнес-процессов компании и их связь с ИТ .....	236
16.4. Стратегия восстановления .....	237
16.4.1. С использованием виртуализации .....	237
16.4.2. С использованием физического резервирования .....	237

16.5. Тактика восстановления .....	238
16.6. Рабочие документы DRP.....	240
16.7. Подготовка персонала .....	241
16.7.1. Подготовка специальной группы .....	241
16.7.2. Формирование кадрового резерва.....	242
16.8. Тестирование восстановления.....	243
16.8.1. Условное, или «бумажное», тестирование.....	243
16.8.2. Проверка на тестовом стенде .....	243
16.6.3. Учения на production-системе .....	244
16.7. Заключение .....	245

## **Вопросы к четвертой части для закрепления материала ..... 246**

## **Часть V. Суровый практический подход ..... 247**

### **Глава 17. «То, что нас не убивает, делает нас сильнее». Ошибки при создании отказоустойчивых систем ..... 248**

17.1. Предисловие .....	248
17.2. Наиболее часто встречающиеся ошибки .....	249
17.2.1. Игнорирование проблемы роста объемов данных при закупке оборудования .....	249
17.2.2. Отсутствие регулярной проверки резервных копий .....	250
17.2.3. Отсутствие контроля за копируемым контентом .....	250
17.2.4. Слишком долгий период между полным резервным копированием .....	251
17.2.5. Слишком много полных копий .....	252
17.2.6. Много копий на одном носителе.....	252
17.3. Реальные примеры того, как нельзя делать резервное копирование.....	253
17.3.1. «Надежное железо», или Как сохранить резервную копию на том же диске.....	253
17.3.2. «Спешка важна при ловле блох». Как не сделать бэкап, когда все копируется на один сервер.....	256
17.3.3. Чем заканчиваются попытки заменить резервное копирование синхронизацией.....	258
17.3.4. Праздничная гирлянда снапшотов .....	259
17.3.5. Ночной кошмар без резервной копии .....	260
17.3.6. Зато у вас антивирус надежный!.....	261
17.4. Заключение .....	262

**Глава 18. Инструменты снятия образов системы для критичных случаев ..... 263**

18.1. Предисловие .....	263
18.2. Краткое описание дистрибутива Redo Backup and Recovery .....	264
18.3. Принципы работы .....	265
18.3.1. Процесс снятия образа .....	265
18.3.2. Восстановление системы .....	270
18.4. Меню Settings .....	271
18.5. Дополнительные возможности .....	272
18.5.1. Программа Gparted .....	272
18.5.2. Другие средства .....	273
18.6. Некоторые ограничения .....	274
18.7. Заключение .....	274

**Глава 19. Система резервного копирования для малого предприятия ..... 275**

19.1. Предисловие .....	275
19.2. Выбор оборудования .....	276
19.2.1. Использование медиа-сервера в качестве хранилища для резервных копий .....	276
19.2.2. Запись данных на ленточный картридж .....	276
19.2.3. Сочетание обоих методов .....	277
19.3. Создание системы резервного копирования .....	277
19.3.1. Планирование процесса восстановления .....	278
19.3.2. Выбор программного обеспечения .....	279
19.3.4. Выбор оборудования .....	280
19.3.5. Подготовка медиа-сервера .....	280
19.3.5. Планирование заданий .....	290
19.4. Тестирование восстановления .....	294
19.5. Завершающая настройка системы резервного копирования .....	295
19.6. Заключение .....	295

**Глава 20. Система резервного копирования крупной компании с многофилиальной структурой ..... 296**

20.1. Предисловие .....	296
20.2. Рассмотренные варианты .....	298
20.2.1. Вариант от EMC .....	298

20.2.2. Вариант автономной системы резервного копирования .....	300
20.3. Решение .....	301
20.4. Процесс внедрения .....	302
20.4.1. Процесс планирования .....	303
20.4.2. Закупка оборудования .....	303
20.4.3. Установка ПО и тестирование .....	304
20.4.4. Передача в эксплуатацию и последующее сопровождение .....	304
20.5. О печальном опыте использования DFS .....	305
20.6. Заключение .....	306

<b>Вопросы к пятой части для закрепления материала .....</b>	<b>307</b>
--	------------

<b>Вместо послесловия .....</b>	<b>308</b>
---------------------------------	------------

<b>Литература и источники .....</b>	<b>309</b>
-------------------------------------	------------

<b>Предметный указатель .....</b>	<b>314</b>
-----------------------------------	------------

# Как и почему я написал эту книгу

*Когда мне хочется прочесть книгу, я ее пишу.*

(Бенджамин Дизраэли –  
английский политический  
и государственный деятель,  
премьер-министр, писатель)

*Теория – это когда вы знаете все,  
но ничего не работает.*

*Практика – это когда все работает,  
но никто не знает, почему.*

*Нужно правильно совмещать  
теорию и практику.*

В свое время я столкнулся с необходимостью сделать вверенную мне инфраструктуру более защищенной от внезапных сбоев и других неприятностей, чтобы вся информация сохранялась в определенной последовательности и при необходимости можно было без труда восстановить данные.

Природная хитрость и осторожность диктуют нам, что, прежде чем кидаться что-то создавать, необходимо для начала провести информационную подготовку. Сначала следует выстроить будущую систему в голове, затем перенести ее на бумагу, а уж только потом воплощать на практике.

И первое, с чем я столкнулся, – по вопросам отказоустойчивости и сохранения данных очень мало какого-либо теоретического материала. То есть сами такие системы есть, а учебников, как правильно их построить, почти нет. На компьютерных полках можно найти массу пособий по программированию, готовым программам, операционным системам и аппаратному обеспечению. Немало трудов издано по вопросам компьютерной безопасности в плане защиты от вторжений.

Но очень мало информации о том, как сохранить свои данные.

Какие-то фрагменты публикуются в инструкциях к программам резервного копирования, какие-то – в книгах по операционным системам и СУБД. Какой-то материал из этого я узнал на курсах по системам хранения. Вот так с миру по нитке собирал материал.

С 2006 года я сотрудничаю с журналом «Системный администратор». Позже появилось приложение «Бизнес и информационные технологии». «СА» – это самый лучший журнал для «айтишников», в нем, в принципе, есть все, что может потребоваться. Если вы не знаете, какой журнал выписать, то «Системный администратор» – это ваш выбор.

Во время своей работы с «Системным администратором» я много написал о спасении данных. И при этом все чаще и чаще я понимал, что нельзя уложить в формат одной журнальной статьи все то, что хотелось бы выплеснуть на страницы публикации.

Порой заканчиваешь статью, уже исчерпаны все разумные границы по ее размеру, и понимаешь, что не успел описать нечто, о чем хочется не просто говорить, а кричать: «Не делайте этого, делайте вот так!»

Через какое-то время я понял, что говорить только о системе резервных ЦОД, только о Disaster Recovery или только об архивном копировании – это еще не все. Нужно комплексное решение проблемы в рамках общей отказоустойчивости. Действительно, какой смысл – лишний раз все восстанавливать из резервной копии, если можно просто не допустить катастрофы.

Так появилась идея написать книгу, в которой бы описывались главные принципы создания таких вот «систем выживания». Дело в том, что если вы будете постоянно что-то чинить, то не хватит времени на самого себя. Представьте, вам нужно почитать книгу для саморазвития, а вместо этого приходится лечить полумертвую файловую систему. Вам хочется пригласить на ужин любимого человека или посидеть с друзьями – а вместо этого всю ночь восстанавливаете «упавший» сервер. И так день за днем. Постепенно это надоест всем, в первую очередь самому «виновнику торжества».

Были случаи, когда меня чествовали как великого героя, например весь персонал бухгалтерии, когда я возвращал из резервной копии убитую базу данных. Программист 1С допустил ошибку и уничтожил данные, все уже пали духом и приготовились работать по ночам и выходным, а тут я как маг и волшебник достаю из бэкапа целехонькую базу данных. Бухгалтеры – это самые прагматичные люди, они ценят вас не за «позитивное мышление», а за решение их проблем. Все слухи про вражду сисадминов с бухгалтерами придумали люди, у которых просто в нужный момент не оказалось нужной копии.

У меня было еще несколько забавных ситуаций. Несколько раз я приходил на разные собеседования, и местные «айтишники» начинали меня забрасывать идиотскими вопросами, а потом внезап-

но самый большой босс спрашивал: «А как бы Вы сохранили наши данные для самой критичной ситуации?» И я объяснял, как создавал такую систему в других местах. Мне часто не верили: «Вы это сами делали, без посторонней помощи?» Ну конечно я ее делал сам, таких как я, любителей сбережения данных, вообще мало на этой планете. «А когда Вы готовы выйти к нам на работу?» Дело в том, что большие боссы – чаще всего неглупые люди, привыкшие взвешивать риски.

В общем, зря я все это рассказал. Теперь вы прочтете книгу и будете на равных конкурировать с «серьезными ребятами» вроде меня. Мне будет сложнее искать высокооплачиваемую работу. Ну и пусть. Когда я доживу до 100 лет, состарюсь и умру, должен же кто-то прийти на мои похороны. Вы придете и скажете: «Да, это тот самый человек, книгу которого я прочел и сделал свою систему восстановления. И это не раз меня выручило». А я тихо улыбнусь про себя.

# Как устроена эта книга

Для простоты восприятия весь основной материал условно разбит на 5 частей. Каждая часть объединяет 4 главы, близкие по тематике. Разумеется, это совсем не означает, что между этими разделами нет взаимосвязи. В конце каждой части приводятся контрольные вопросы для закрепления материала. Ниже приведена более подробная информация о том, что рассказывается в каждой главе.

**Часть I. Краеугольные камни.** Как следует из названия, изложенный в ней материал служит фундаментом для понимания остальных частей книги.

**Глава 1. О реальном месте ИТ в современном бизнесе.** Наверное, это странно осознавать, но очень многие люди мало отдают себе отчет в том, как бы изменилась жизнь, не будь информационных технологий. И уж точно мало задумываются, что делают ИТ для их бизнеса. В итоге роль таких понятий, как отказоустойчивость или сохранение данных, становится для них чем-то призрачным, необязательным. Чтобы приобрести яркий блеск в глазах и твердость духа в осуществлении своих замыслов по созданию отказоустойчивой системы с надежным резервным копированием, нужно понимать, зачем все это нужно. И первая глава дает это понимание.

**Глава 2. Популярная терминология.** Чтобы говорить с кем-то, нужно знать его язык. Чтобы что-то обсуждать, нужно иметь соответствующий словарный запас. И вторая глава сообщает читателям необходимый набор терминов и определений. Но не нужно готовить себя к чтению скучной академической статьи наподобие энциклопедического материала. Наоборот! Читателя ждут не только новые слова, но и интересные факты, поучительные примеры с «разбором полетов».

**Глава 3. Избыточность на службе отказоустойчивости.** Вначале эта глава задумывалась как расширенное продолжение предыдущей. Но нельзя говорить о таких важных вещах в рамках «термин – пояснение». В итоге все вылилось в рассказ о том, как сделать надежной саму основу информационной инфраструктуры – массив хранения данных. Как использовать методы избыточности, какие бывают алгоритмы защиты при записи. Эти понятия еще будут встречаться на протяжении всей книги.

**Глава 4. Архитектура систем резервного копирования.** Трудно построить дом, если не знаешь, на сколько человек он рассчитан, какая у него планировка. Точно так же обстоит дело с резервным копи-

рованием. Эта глава расскажет читателю о базовых понятиях, таких как топология и различные уровни позиционирования систем.

**Часть II. Резервное копирование как оно есть.** В этой части собраны главы о внутреннем мире систем резервного копирования-восстановления.

**Глава 5. Виды резервного копирования.** В этой главе пойдет речь о том, как можно сберечь ресурсы и научить работать систему резервного копирования в определенных условиях. Какие для этого есть способы.

**Глава 6. Схемы ротации носителей.** Продолжаем рассказ о методах адаптации системы резервного копирования. Читатель узнает, как рассчитать необходимое количество носителей, когда нужно менять кассету и по какой системе лучше организовать изъятие носителей для перемещения резервных копий за пределы периметра предприятия.

**Глава 7. Цели и технологии резервного копирования. О различиях в технологии резервного копирования.** Как можно куда-то идти, не видя перед собой ясной цели? Седьмая глава расскажет читателю о двойном подходе при резервном копировании и двух схожих, но и в то же время различных целях, которые оно может преследовать. И каждой цели лучше соответствует определенный метод.

**Глава 8. Резервное копирование виртуальных систем.** Внутреннее устройство систем виртуализации серверов приоткроет свою завесу, и читатель узнает, как лучше, быстрее, надежнее организовать резервное копирование в этих непростых условиях, какие внутренние механизмы для этого используются и какие нюансы следует учесть.

**Часть III. Оборудование для систем резервного копирования.** Эта часть посвящена в основном аппаратным системам хранения и доставки резервных копий. Ведь как оптимально не была бы организована система резервного копирования, получившиеся результаты необходимо где-то размещать. Какое оборудование предпочесть, в пользу чего нужно сделать свой выбор, и стоит ли вообще от чего-то отказываться?

**Глава 9. Системы хранения резервных копий – единство и борьба противоположностей.** В этой главе идет речь о вечном соперничестве двух направлений: дисковых хранилищ и ленточных библиотек. Что предпочесть в том или ином случае? А может быть, стоит использовать и то, и другое для разных целей в одном проекте?

**Глава 10. Магнитные ленты и вездесущий LTO.** Здесь читателя ждет подробный, но совсем нескучный рассказ о системах записи на ленту, вездесущем стандарте LTO и различных интересных нюансах.

**Глава 11. Дисковая система хранения на базе NAS.** Знакомимся с работой сетевого хранилища на примере реального устройства. И, как всегда, читателя ждут интересные особенности.

**Глава 12. Сеть обмена данными и резервное копирование.** Выбор средства хранения резервных копий – это очень важно. Но не менее важно то, как они туда попадают. Как организовать процесс резервного копирования так, чтобы не мешать бизнесу? Можно ли заранее спроектировать сетевую инфраструктуру так, чтобы не пришлось чем-то жертвовать? И что делать, если в бюджете недостаточно денег для создания серьезной сети хранения данных? Обо всем об этом рассказывается в 12-й главе.

**Часть IV. «Что нам стоит дом построить, нарисуем – будем жить».**  
**Проектирование отказоустойчивых систем.** По названию нетрудно догадаться, что в этой части собрано все, что касается непосредственно вопросов проектирования, начиная от построения команды и заканчивая составлением специального плана восстановления системы.

**Глава 13, которой быть не должно. Что мешает создать отказоустойчивую инфраструктуру?** Тринадцатый номер считается несчастливым у суеверных людей. И глава под этим номером посвящена достаточно сложным вопросам. Что может помешать успешному внедрению самого замечательного проекта? И какие подводные камни скрывает дефицит времени? Как личность влияет на качество внедряемых систем? И как не погрязнуть в вечных недоделках? Несмотря на серьезность вопросов, чувство юмора и взгляд под неожиданным углом позволяют найти оригинальные ответы.

**Глава 14. Десять шагов к вершине. Процесс проектирования отказоустойчивых систем.** Здесь описан универсальный порядок проектирования любых систем. Автор специально не делает различий между внутренними проектами и внешними. Постигнув основные принципы, читателю уже не составит труда организовать процесс создания и внедрения той или иной системы. Разумеется, она должна быть отказоустойчивая и иметь надежное резервное копирование.

**Глава 15. «Возьми с собой ключи от моих дверей». О безопасности резервного копирования** Как избежать кражи важных данных? Как свести на нет риск при утере носителя с резервной копией? Почему не нужно подписывать ленточные картриджи «понятными названиями»? Обо всем об этом читатель узнает, прочитав эту главу.

**Глава 16. Жизнь после смерти, или Как восстановить систему после мировой катастрофы.** Это фактически кульминация книги. Автор в простой и понятной форме расскажет о множестве интересных

вещей, например зачем досконально знать бизнес-процессы компании, когда можно попытаться «восстановить все из бэкапа» или почему нужно позаботиться о своих сотрудниках сразу после катастрофы.

**Часть V. Суровый практический подход.** В последней части книги автор делится своим опытом, как надо и как не надо строить свои отказоустойчивые системы.

**Глава 17. «То, что нас не убивает, делает нас сильнее». Ошибки при создании отказоустойчивых систем.** В этой главе в легкой манере повествования читатель узнает о чужих ошибках и о том, к каким последствиям это могло бы привести.

**Глава 18. Инструменты снятия образов системы для критичных случаев.** Хорошо, когда есть система резервного копирования, работающая как часы и создающая копии в одно и то же время. Но что делать, если нужно сделать образ системы важного сервера непосредственно перед обновлением? Или сделать полную копию с директорского ноутбука, в котором забарахлил жесткий диск? На помощь приходят системы снятия образов, о которых автор рассказывает в этой главе.

**Глава 19. Система резервного копирования для малого предприятия.** Малый бизнес всегда является болевой точкой для ИТ-технологий. Малое финансирование в сочетании с высокой нестабильностью создает дополнительные трудности. Но специалисту, владеющему принципами резервного копирования, любая задача по плечу. В этой главе читатель узнает на реальном примере, как помочь малому предприятию в этом непростом вопросе.

**Глава 20. Система резервного копирования крупной компании с многофилиальной структурой.** Что делать, если компания быстро разрослась и процесс резервного копирования вышел из-под контроля? Как эффективно организовать перемещение копий данных из филиалов в центральный офис? Какие опасности таит в себе репликация? На все эти вопросы читатель получит ответы на примере конкретного проекта.

# Благодарности

В этой части я хочу выразить свою благодарность тем людям, без которых эта книга просто никогда не была бы написана.

Моим родителям, Николаю Алексеевичу и Ольге Ивановне, которые привили мне вкус к чтению и научили не бояться выражать свою мысль.

Моим Учителям и друзьям, профессиональным историкам и прекрасным преподавателям: Игорю Ивановичу Кустову и Олегу Ивановичу Кустову – за поддержку и наставничество в студенческие годы. Они помогли мне обрести смысл в жизни. Я навсегда запомнил слова Игоря Ивановича: «У человека должно быть дело, которому он посвящает свою жизнь».

Моей любимой жене Мадине, которая поддерживала меня и мои начинания все эти годы.

Главному редактору журнала «Системный администратор» Галине Владимировне Положевец и всему коллективу и руководству этого прекрасного издания.

Благодарю всех, кто оказывал мне помощь и поддержку все эти годы.

Вы дали мне все: опыт, знания, возможность донести свои мысли до других людей.

За это вам огромное спасибо.

Часть I



# КРАЕУГОЛЬНЫЕ КАМНИ

# Глава 1

## О реальном месте ИТ в современном бизнесе

– Папа, а что значит «скупой платит дважды»?

– Пример: наши менеджеры по продажам скинулись по 50 рублей и купили китайский чайник на рынке. Сейчас скидываются по 800 долларов на ремонт сервера с клиентской базой и по 200 рублей на нормальный чайник.

(Из разговора системного администратора со своим сыном)

### 1.1. Зачем нужны информационные технологии

И правда, зачем? Жили сто лет без компьютеров, войну выиграли, страну из руин подняли и дальше проживем! Разумеется, это сарказм. Тем не менее такие рассуждения можно услышать даже из уст тех, кто, по идее, должен бороться за выживание информационных технологий.

Увы, ИТ-специалисты сами порой не осознают, для чего нужны плоды их труда и какую пользу они приносят. На вопрос: «А для чего вообще нужны информационные технологии?» – в ответ обычно получаешь что-то вроде: «Чтобы бухгалтерия работала... чтобы почта ходила, Интернет опять же...» и т. п. Но бухгалтерия и почтовая переписка существовали задолго до изобретения компьютеров. Так зачем они все-таки понадобились бизнесу, эти самые компьютеры и все, что с ними связано?..

### 1.1.1. Почему нужно читать эту главу?

«Что, я не знаю, зачем компьютеры нужны?» – воскликнет нетерпеливый читатель. И будет прав. Мы прекрасно знаем, для чего используются компьютеры. Но мало кто отдает себе отчет об их реальной роли в нашей жизни.

Любой продавец с результатами «чуть выше среднего» скажет вам, что эффективные продажи невозможны без веры в высокие потребительские качества товара. Но нет веры без понимания, без попытки осознать, а зачем такое вообще нужно?

ИТ-специалист – тоже своего рода продавец. Он «продает» не только свои услуги как специалиста, но и результаты работы вверенной инфраструктуры. И если он не может сделать это грамотно, ни о каком нормальном финансировании, тем более высокой оплате труда, и речи быть не может.

Можно сколько угодно жаловаться на то, что власть и бизнес не хотят тратить денег на техническое развитие, что денег на новый жесткий диск не выпросишь, не говоря уже об отказоустойчивых кластерах и промышленных системах резервного копирования. Но как можно кого-то убедить, если сами «айтишники» не до конца понимают значения своего труда? И пока они не овладеют идеологией и не приведут в порядок свои внутренние установки с реалиями современной жизни, ни о какой поддержке со стороны и речи быть не может. Именно поэтому я начал эту книгу с проработки идеологических вопросов.

Вы уже обратили внимание, что когда обычно говорят об ИТ, начинают перечислять отрывочные сервисы и программы, и мало кто вспоминает о глобальном влиянии информационных технологий? Но дело в том, что компьютеры так прочно вплелись в нашу с вами жизнь, что уже невозможно представить себе тот или иной аспект без их использования.

### 1.1.2. Пример работы торговой фирмы

Возьмем самый простой вариант – обычную торговую фирму. Тот самый случай, когда не нужно заниматься строительством производственных помещений, организацией технологического процесса и т. д. Простой вариант с перепродажей уже имеющегося товара.

Вот три человека объединились и открыли торговое предприятие. Один из них закупает товар, второй этот товар реализует, а третий осуществляет учет и общее руководство (см. рис. 1.1). Сейчас у них нет острой необходимости вводить какую-либо автоматизацию. Все,

что касается движения товара, а также бухгалтерский и управленческий учет можно отразить по старинке – «на бумажках».

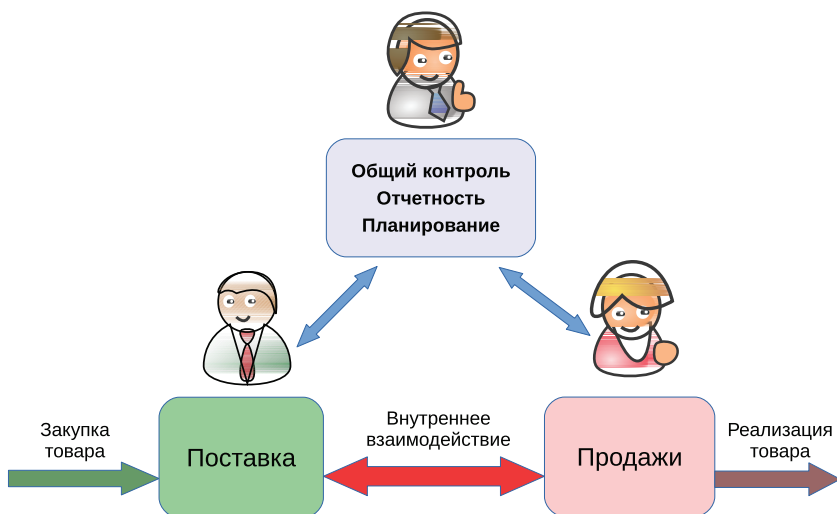


Рис. 1.1. Стартовая позиция компании

Бизнес постепенно растет, растет и наша бизнес-тройка: «закупка – учет – продажи» уже не справляется с возросшими объемами. Они приглашают на работу своих друзей. Теперь в бизнес-процессе присутствуют два слоя: учредители и наемные работники (см. рис. 1.2). Так как все сотрудники знают друг друга лично и состоят в хороших отношениях, и оборот компании не такой уж большой, то в такой ситуации особых проблем в работе, как правило, не возникает.

Но вот наша компания выросла еще больше, и учредителям придется принимать на работу уже новых людей, по рекомендации своих подчиненных. В бизнесе участвуют уже три уровня (см. рис. 1.3). Что касается новых сотрудников из «нижнего слоя», они знакомы с учредителями только понаслышке и не особенно посвящены в дела компании. Их участие в компании уже сводится к выполнению предписанных обязанностей.

До этого момента людей на работу принимали исключительно по рекомендации, когда уровень навыков и личные качества хорошо известны хотя бы одному сотруднику. Любой бизнес рано или поздно достигнет такого момента, когда придется набирать в штат малозна-



Рис. 1.2. Подчиненные лично знакомы с учредителями

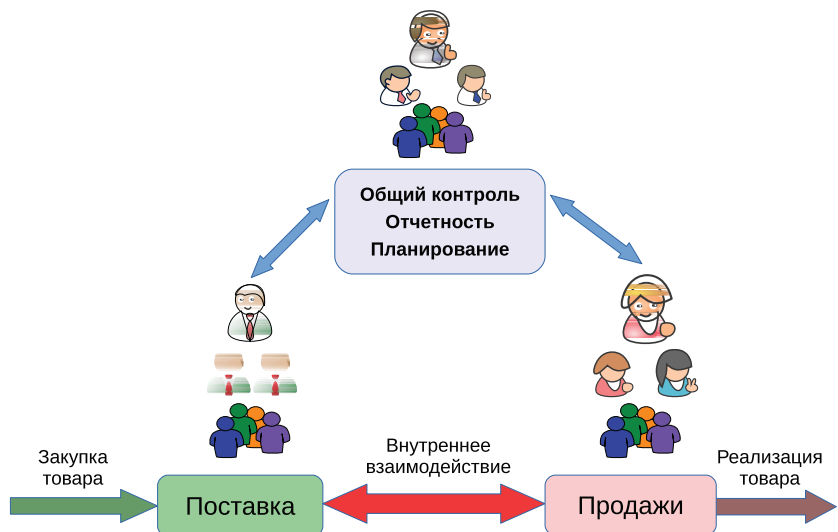


Рис. 1.3. Постепенный рост компании

комых или совсем незнакомых людей. Таким образом появляются четвертый, пятый и другие уровни, которые уже почти целиком состоят из наемных сотрудников (см. рис. 1.4).

Посмотрим внимательнее на эту новую структуру компании. Основное место в ее рядах составляют наемные работники, которым уже все равно, как будут обстоять дела у бизнеса. Они не являются соучредителями, их доход почти не зависит от успехов компании в целом. Этот бизнес не является их детищем, а всего лишь очередным трудоустройством. Судьба компании их не волнует. Если им перестанут выплачивать зарплату и/или бонусы, они просто уволятся и пойдут искать другую работу. Люди заняты огромным количеством мелких операций: посчитать на деревянных счетах, напечатать отчет, вручную заполнить множество бланков, доставить почту и т. д. При этом основные мысли этих людей сосредоточены на том, чтобы поскорее закончился их рабочий день.

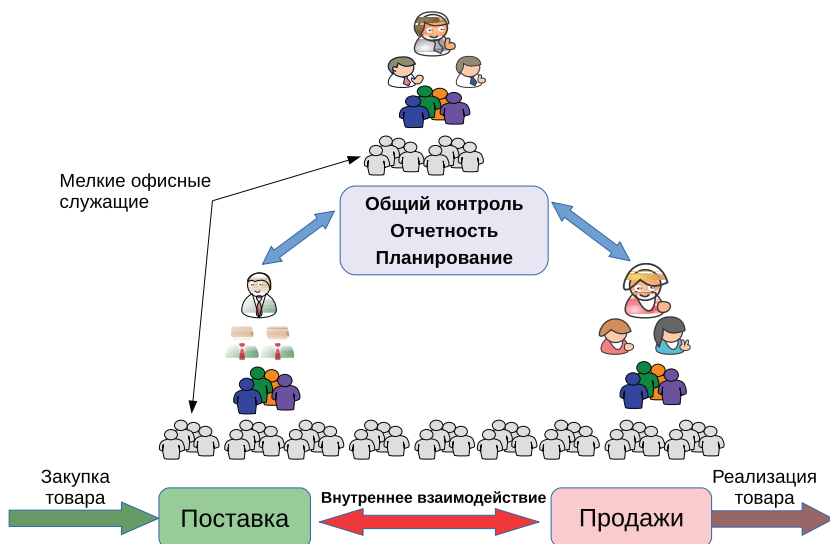


Рис. 1.4. Увеличение численности наемного персонала

Так было на заре индустриальной эпохи – любой бизнес сопровождала многочисленная армия клерков. И мельчайшая ошибка такого «маленького человека» могла вызвать сбой и даже разрушить огромную корпорацию. Если же говорить о вредительстве (это явление

присутствует везде, где есть конкуренция), то всем известно, что основной урон бизнесу приносят не злые гении, а обиженные сотрудники на младших должностях. Именно они способны изрешетить надежную систему любого процесса, подобно термитам, превращающим в труху новенькое здание.

### 1.1.3. И вот приходит эра информационных технологий

На протяжении всей истории бизнес ищет возможность освободиться от человеческого фактора или хотя бы снизить его влияние.

Если мы посмотрим на рис. 1.5, то увидим совершенно иную картину. Несколько нижних слоев нашей модели предприятия заменили на автоматизированные процессы посредством внедрения информационных технологий. Проще говоря, мелких клерков заменила компьютерная техника. Больше нет того ненадежного слоя, переполненного бесчисленными «трагедиями маленького человека». Вместо них образовалась мощная прослойка ИТ, которая служит продвижению бизнес-процессов, подобно тому, как железная дорога перевозит грузы и пассажиров. И вместо целой орды наемных работников рутинные операции выполняет компьютерная система.

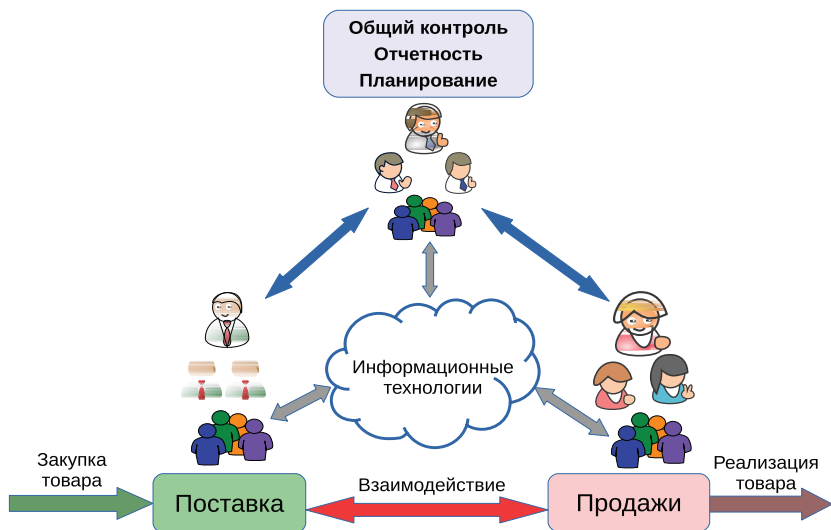


Рис. 1.5. Информационные технологии на службе бизнеса

Разумеется, новая инфраструктура, построенная на информационных технологиях, нуждается в заботливом уходе и планомерном развитии. Именно поэтому образовалось новое подразделение бизнеса – ИТ-департамент. Но заинтересовать нескольких профессионалов в результатах работы компании гораздо проще, чем множество мелких клерков.

При использовании информационных технологий резко возрастает требование к уровню профессионализма людей, работающих с новой системой. Но в то же время, если сопоставить величину фонда оплаты труда, который тратился на многочисленных мелких клерков, и сравнить его пусть с неплохой, но отнюдь не самой высокой зарплатой системных инженеров и администраторов, выгода от использования компьютерной техники становится очевидной. И это не говоря о том, что каждого рядового служащего нужно обеспечить рабочим местом, средствами для работы (мебель, канцтовары, пишущая машинка и много чего другого), а также минимальными удобствами (питьевая вода, освещение, отопление, туалет и т. д.). Учитывая цены на аренду офисных помещений, бизнес получает большую экономию от внедрения информационных технологий.

Разумеется, применение ИТ не ограничивается функциями учета и документооборота между контрагентами. Ниже идет описание далеко не полного списка функций, которые взяли на себя информационные технологии.

Функции, которые перешли к ИТ или появились с их развитием:

- почтовая переписка;
- управленческий учет, включая анализ прошлых периодов и составление прогнозов;
- бухгалтерский учет и формирование отчетности;
- системы автоматического проектирования;
- системы управления технологическим процессом;
- средства голосовой и видеосвязи;
- реклама нового типа;
- возможности удаленной работы;
- хранение и каталогизация информации в больших объемах;
- распределенный доступ к информации;
- замена эфирного аналогового вещания;
- обеспечение аудио- и видеосвязи;
- контроль доступа в помещение;
- компьютерный дизайн и создание произведений искусства.

Поэтому было бы неправильно говорить о роли информационных технологий только в рамках экономии денежных средств. Сегодня информационные технологии – собственно, это и есть бизнес в его современной интерпретации. Его внутренняя жизнь, его связь с внешним миром, его «светлое будущее и темное прошлое». Даже если глобальный вывод из строя ИТ-ресурсов не приведет к полному краху предприятия, в любом случае произойдет остановка бизнес-процессов на неопределенное время.



В данном случае мы рассмотрели самый простой, примитивный пример с торговой фирмой среднего уровня. Если речь идет о бизнесе, связанном с созданием интеллектуальной собственности, например архитектура, научные разработки, дизайн, управление инвестициями, АСУ ТП и т. д., то зависимость от ИТ возрастает в десятки раз. Содержать армию чертежников вместо нескольких компьютеров с САД или провести компьютерное моделирование без компьютеров пока что не представляется возможным.

## 1.2. Ахиллесова пята современного бизнеса

Но что же произойдет, если эта прекрасная, стройная система вдруг выйдет из строя? Бизнес понесет убытки, не сопоставимые с проблемами от некорректной работы одного работника. Следовательно, информационные технологии, с одной стороны, уменьшают риски, связанные с человеческим фактором, с другой – значительно появляются новые угрозы, в первую очередь в области информационной безопасности и отказоустойчивости.

Выше я уже писал о том, что при использовании информационных технологий бизнес попадает в зависимость от людей, которые обслуживают новую ИТ-инфраструктуру. Но это еще не все.

Насколько бы хорошо не работала группа системных инженеров и администраторов, все равно остается риск стать жертвой форс-мажорных обстоятельств: пожара, наводнения и других напастей. Нет защиты и от ошибок, случайно вносимых компьютерными пользователями при обработке данных.

И всегда остается риск стать жертвой мошенников, на этот раз вооруженных новыми технологиями. Поэтому возрастают требования к моральным качествам персонала, который работает в информационном окружении.

Современным управленцам нужно очень хорошо запомнить и никогда не забывать, что ИТ-специалист – это не просто наемный со-

трудник или «слесарь по компьютерам», а добрый ангел-хранитель, оберегающий бизнес от разрушения. Что такое сегодня кража информации? Например, передача конкурирующей фирме базы данных контрагентов или перспективных идей и проектов? По сути, это и есть отъем бизнеса как таковой. И нужно обязательно учитывать репутационные риски, когда остальные участники рынка просто не захотят иметь дело с бизнесом неудачников, которые так и не смогли организовать нормальное функционирование информационной составляющей.

Теперь становится понятно, почему с увеличением доли информационных технологий необходимо заботиться о сохранении ИТ-инфраструктуры в рабочем состоянии. В первую очередь это резервное копирование, обеспечение отказоустойчивости и безопасности. Как раз та самая «троица», которой в России, по традиции, уделяют не слишком много внимания и заботы.

### 1.3. О понимании политического момента

С первого взгляда автор пишет очевидные вещи, но так ли уж они очевидны для большинства жителей бывшего СССР?

Исторически сложилась уникальная ситуация. В XX веке две ведущие сверхдержавы устроили между собой странное состязание в демонстрации превосходства, назвав это «холодной войной».

Пока советские идеологи клеймили «лженауку» кибернетику как «продажную девку империализма», на «загнивающем Западе» уже строили мощные вычислительные комплексы.

В СССР содержали миллионы канцелярских служащих, в то время как «проклятые буржуи» проводили массовые увольнения бесполезных офисных работников. Основной процент внедрения решений на базе информационных технологий присутствовал в областях, связанных с оборонной промышленностью. При этом своих наработок в этой сфере было совсем немного, большая часть технологий была правдами-неправдами заимствована у «потенциального противника».

Потом пришла эпоха перестройки, а с ней эпоха прозрения и понимания масштабов отставания в высокотехнологичных областях. Началась эра насильственной «компьютеризации», когда предприятия в приказном порядке обязывали приобретать и внедрять вычислительную технику. Что делать с этим свалившимся на голову «богатством», в то время как на большинстве советских предприятий даже в области управления присутствовала колоссальная доля

ручного труда, – этого почти никто не понимал и не хотел понимать. И самое главное – никто не знал, что делать с высвобождающимся персоналом, ведь безработицы в СССР официально не существовало и, по заверениям КПСС, не было, да и быть не могло. В результате сформировалось устойчивое отношение к ИТ как к чему-то необязательному, вроде подспорья для бухгалтерии. В первые годы после развала СССР это мнение только укрепились на фоне всеобщей нищеты и дезориентации. Увы, подобное отношение во многих местах сохраняется до сих пор.

Из-за такого «культурного наследия» ИТ-специалисты в большинстве своем являются такими же ретрансляторами наплевательского отношения к ИТ, как и все остальные сотрудники.

Вот почему во многих российских предприятиях гораздо проще «выбить деньги» у руководства на новую офисную мебель, чем на модернизацию морально устаревшей системы резервного копирования, не говоря уже о чем-то другом.

## 1.4. Заключение

Нужно помнить, что если мы уже пришли к тому, что информационные технологии – это и сам бизнес в его нынешнем воплощении, по сути, это и есть наша с вами жизнь. Умрет ИТ – значит умрет бизнес, умрет государство, умрет все, где применяются «эти проклятые компьютеры». И задача людей, работающих в этой отрасли, – продлить жизнь там, где от них это зависит.

# Глава 2

## Популярная терминология

*Работа программиста и работа шамана имеют много общего: например, оба произносят непонятные слова, совершают непонятные действия и не могут объяснить, как это все работает.*

(Неизвестный программист-шаман)

### 2.1. Начнем с изучения специфики бизнеса

В самом начале нелишним будет уделить внимание терминам, обозначающим те или иные, на первый взгляд, такие знакомые вещи.

#### 2.1.1. Бизнес-процесс

Это последовательность процедур по преобразованию ресурсов, полученных на входе, в готовый продукт, необходимый для заказчика. В роли заказчика бизнес-процесса могут выступать не только внешние лица или организации, но и внутренние службы. Например, в случае с поддержанием работы системы складского учета в этом качестве выступает бухгалтерия и т. д.

«ИТ-бизнес-процесс» или «бизнес-процесс ИТ». Под этим названием чаще всего подразумевают процессы, происходящие внутри ИТ-инфраструктуры или где степень участия ИТ-службы достаточно велика. Например, обработка заявок пользователей. Такое ограничение позволяет отсеять факторы, напрямую не относящиеся к данной сфере, и сконцентрироваться на решении конкретных задач, связанных непосредственно с ИТ-инфраструктурой.

### *Business Environment Analysis (BEA)*

В переводе – анализ бизнес-процессов. Фактически означает выстраивание иерархии действующих в компании бизнес-процессов по степени важности и непрерывности.



Метод управления, основанный на бизнес-процессах, – отнюдь не единственный вариант. Есть еще **функциональный подход** к управлению, когда организация представляется как набор подразделений (департаментов), каждое из которых исполняет определенные функции. Такой вариант до сих пор используется во многих российских организациях. Особенно это характерно для различных государственных структур и ведомств. Однако для эффективного управления мало знать, «кто за что отвечает». В реалиях XXI века уже недостаточно время от времени вызывать начальников подразделений «на ковер» и «устраивать головомойку». Необходимо вдумчиво и тщательно выстраивать взаимодействие между всеми участниками процесса.

Определившись с общими понятиями ведения бизнеса, самое время углубиться в описание специальных терминов, описывающих взаимодействие деловой сферы с «виртуальным отражением» нашей жизни – информационными технологиями.

#### 2.1.2. Мера терпения

##### *Maximum tolerable period of disruption (MTPOD)*

Дословно можно перевести как «максимально терпимый период разрушения». В российской литературе более известен как «максимальное время простоя». Обозначает предельное время, когда ключевые продукты или услуги предприятия могут быть недоступны, прежде чем партнеры начинают считать данные условия неприемлемыми. Проще говоря, некий срок, по истечении которого каждая минута простоя ускоряет приближение к окончательному краху.

Возьмем, к примеру, сферу строительства. Завод, производящий стройматериалы, по причине сбоя сервера базы данных не может выполнять отгрузки товара. Соответственно, его партнеры – строительные компании могут потерпеть пару дней, а потом объявят о срыве поставок и будут искать другого поставщика. Так вот два дня – это и есть то самое «максимальное время простоя».

Надо ли говорить, что MTPOD, по сути, и есть ключевая величина, определяющая основные правила игры. Своего рода мера терпения бизнеса по отношению к работе ИТ-подразделения.

Все же становится немного проще от того, что в данном случае мера терпения имеет явное числовое выражение. В отличие от субъективной оценки работы системного администратора капризными пользователями.

Чтобы не упустить самого важного, а самое главное – не потерять контроль над ситуацией, этот параметр совместно с другими характеристиками закрепляется в специальных документах, о которых пойдет речь ниже.

### ***Service Level Agreement (SLA)***

Дословно – «соглашение об уровне предоставления услуги» из библиотеки ITIL, формальный договор между заказчиком услуги и ее поставщиком. Договор включает описание услуги, права и обязанности сторон и, самое главное, согласованный уровень качества предоставления данной услуги. Следует также заметить, что в некоторых случаях в роли заказчика и потребителя могут выступать разные лица.

### ***Recovery Point Objective (RPO)***

В переводе означает «допустимая точка восстановления». Это период допустимого уровня потери данных в случае прерывания операций. Проще говоря, это временная точка, начиная с которой можно восстановить данные. Например, если процесс может предоставить данные за последние 24 часа до аварии, то последняя резервная копия должна быть завершена не позднее 24 часов назад.

### ***Recovery Time Objective (RTO)***

Расшифровывается как «время восстановления». Промежуток времени, необходимый для реанимации функционирования ИТ-сервисов, необходимых для работы. На такую «реанимацию» из резервной копии может потребоваться от нескольких минут до нескольких дней. Все зависит от используемой системы резервного копирования, доступности резервных копий и сложности организации работы самого сервиса и т. д. (см. рис. 2.1).

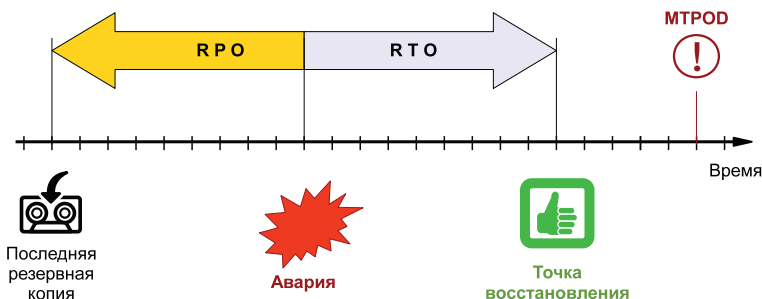


Рис. 2.1. Точки RPO, RTO и МТПОД на графике

При расчете времени восстановления и соотнесении ее с МТРОД часто не учитывают одну простую вещь. Мало восстановить систему в первоначальном виде, который был на момент RPO. Необходимо занести в систему данные, которые накопились со времени последней точки отката. Например, во вторник к концу рабочего дня рухнул сервер баз данных бухгалтерии. Имеющаяся резервная копия была сделана в ночь с понедельника на вторник. Таким образом, были потеряны данные практически за весь рабочий день. Еще сутки понадобились на восстановление системы (включая закупку комплектующих), и в ночь со среды на четверг сервер был запущен. При этом допустимое время простоя – два дня. На первый взгляд, затраченное время соответствует предъявленным требованиям. Но сама база данных содержит информацию только на момент последней резервной копии, то есть в конце понедельника. И бухгалтерам потребуется еще два дня на занесение в базу первичных документов и выполнение проводок. Итого на полное восстановление системы потребовалось 3 дня, то есть к полноценной работе предприятие вернулось только в пятницу вечером. Учитывая предстоящие выходные: субботу, воскресенье, – нормальное функционирование бизнеса стало возможным только со следующей недели (см. рис. 2.2). В данном случае можно говорить, что критичные для бизнеса показатели были рассчитаны неверно и, соответственно, выбрана неправильная стратегия восстановления.

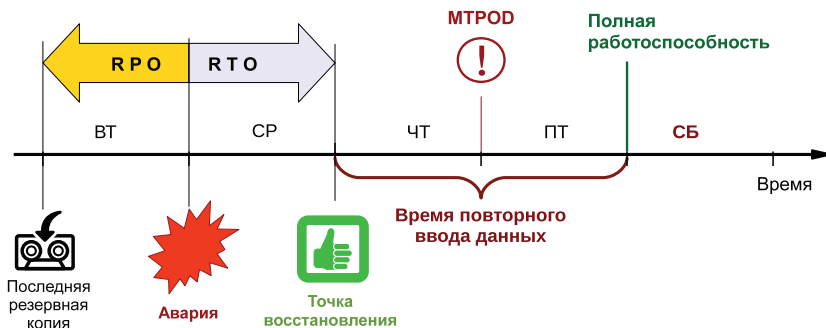


Рис. 2.2. Расчетное и полное восстановление работы

## 2.2. О факторах риска и мерах предосторожности

Неприятности, преследующие ИТ-инфраструктуры, условно могут быть разделены на несколько категорий.

**Природные**, такие как наводнения, ураганы, торнадо и землетрясения. Предотвратить стихийные бедствия невозможно, но мы можем спланировать наши действия, чтобы избежать опасных ситуаций и хорошо подготовиться на случай, если они все-таки произойдут.

**Техногенные**, такие как производственные аварии и другие проблемы, разрушающие инфраструктуру. Сюда же можно отнести серьезные сбои оборудования, повлекшие за собой остановку бизнес-процессов. Например, пожар или прорвавшаяся батарея в серверной. Или случайные короткие замыкания, выводящие из строя аппаратное обеспечение.

**Антропогенные.** Теоретически их также можно отнести к техногенным факторам, если бы не одно серьезное отличие – непосредственное участие человека. Сюда стоит отнести угрозы безопасности: заражения вирусами, всевозможные внутренние и внешние атаки и взломы, серьезные ошибки ИТ-персонала, а также неудачные реализации изменений, например установка обновлений. Также стоит упомянуть о риске использования нелегального программного обеспечения и его последствиях: изъятии компьютерной техники, судебных разбирательствах и т. д.

Для предотвращения аварий второго и третьего типов особую роль играют наблюдение за системой, тестирование компонентов и тщательное планирование своих действий. Но в любом случае, для того чтобы избежать тяжелых последствий всех катастроф, необходимы превентивные меры, о которых пойдет речь ниже.

## 2.3. Общие термины обеспечения отказоустойчивости

Определившись с терминами описания бизнеса, хотелось бы, чтобы он был достаточно устойчив к нашим факторам риска. Для этого также необходимо определиться с соответствующей системой обозначений.

### 2.3.1. Business Continuity Planning (BCP)

Обозначает созданный заранее, документально утвержденный и принятый к исполнению комплекс мер и процедур для обеспечения непрерывной работы предприятия путем предотвращения сбоев в бизнес-процессах, согласно принятому SLA. Проще говоря, набор документов на тему, что нужно делать и чего делать нельзя, чтобы

не было мучительно больно за уничтоженную ИТ-инфраструктуру и весь бизнес в целом.

### **2.3.2. Disaster Recovery**

Проще говоря, полное восстановление после сбоев – комплекс мер для обеспечения непрерывности действия ИТ-системы. Если в результате аварии предпринимательская деятельность остановится, наиболее важной функцией Disaster Recovery является восстановление работы.

### **2.3.3. Disaster Recovery Plan (DRP)**

Это подготовленный заранее, утвержденный и отработанный план восстановления бизнес-процессов при их нарушении вследствие нештатных ситуаций, инструкция на случай, если проблемы избежать не удалось. Проще говоря: «Что делать, если случилось горе».

По сути, DRP – важная часть BCP программы. Помните, что это разные документы, хотя многие ИТ-специалисты подчас не видят различий. Иногда руководство компании отказывается рассматривать DRP, ссылаясь на то, что накануне был подписан BCP (или наоборот). Основное отличие заключается в том, что BCP – это документ более общего плана, и вопросы, связанные с восстановлением работоспособности ИТ-инфраструктуры, описаны в крупномасштабных чертах.

Отсутствие здравого смысла непременно приводит к катастрофе и нищете. По бедности и недомыслию во многих российских компаниях до сих пор принято считать, что вышеописанные документы нужны только высокотехнологичным корпорациям, у которых финансовые потери от кратковременного сбоя измеряются миллионами, миллиардами, тысячами миллиардов долларов и евро. Но разве для владельцев небольшого бизнеса пусть и маленький, но ощутимый убыток менее значим, нежели миллион для миллиардера?

## **2.4. Что такое центр обработки данных и чем он отличается от серверной комнаты**

### **2.4.1. Определение ЦОД**

Центр обработки данных (ЦОД) – это комплексная централизованная система, спроектированная с учетом современных требований отказоустойчивости и безопасности.

Основное отличие ЦОД от обычной серверной комнаты заключается в комплексном подходе при проектировании и реализации не только информационных, но и инженерных систем, таких как электропитание, кондиционирование, видеонаблюдение, шумоподавление и т. д.

Основное назначение ЦОД – обеспечение гарантированной безотказной работы ИТ-инфраструктуры организации с заданными характеристиками высокой доступности, надежности, безопасности. Важным условием при этом является наличие системы наблюдения и управления каждой единицей оборудования и ПО. Такой подход позволяет создавать резервные площадки для дублирования ИТ-структуры предприятий с сохранением максимально возможной функциональности в случае необходимости.

### 2.4.2. Компоненты ЦОД

Ниже перечислено несколько стандартных составляющих:

- высоконадежное серверное оборудование;
- системы хранения и передачи данных (чаще всего построение по технологии SAN);
- сетевая инфраструктура (LAN и WAN);
- программное обеспечение;
- архитектурно-технические решения;
- инженерная инфраструктура;
- система контроля доступа и физическая защита помещений;
- противопожарная защита, например автоматическая система пожаротушения;
- система мониторинга и управления.

## 2.5. Уровни Disaster Recovery

Для более тщательного анализа возможностей той или иной системы восстановления, а также для наведения порядка в проектной и рабочей документации принято следующее разделение по уровням для систем Disaster Recovery.

- «0» – Локальный. Данные хранятся на диске или магнитной ленте только на одном объекте. Время RPO и RTO зависит от используемой политики резервного копирования. Недостатками такого решения являются непредсказуемость времени восстановления и высокая вероятность потери данных без восстановления.

- «1» – Резервная копия в удаленное хранилище. Данные хранятся на магнитных лентах, которые время от времени переносятся в удаленное хранилище (off-site). Это обеспечивает резервирование данных в случае выхода из строя ИТ-инфраструктуры на единственном объекте и дает возможность провести восстановление. RPO зависит от графика перемещения лент, например для ежедневного резервного копирования и транспортировки ленты в другое место необходимо 24 часа. Из опыта: при таком подходе на восстановление ИТ-инфраструктуры понадобится около недели.
- «2» – Резервная копия на резервный центр. Резервное копирование выполняется в основном центре. Резервные копии раз в день транспортируются в резервный центр. Время RPO и RTO зависит от используемой политики резервного копирования. Время, необходимое для восстановления работы, – один день.
- «3» – Резервное копирование при помощи сети. Резервные копии выполняются на диски или магнитную ленту в основном центре. Через определенный промежуток времени происходит перенос копии в резервный ЦОД. RPO и RTO зависит от используемой политики резервного копирования. При правильной политике резервного копирования и соответствующей частоте обновления в резервном ЦОД на необходимое для операции восстановление (RTO) уходит несколько часов.
- «4» – Асинхронная репликация. Резервное копирование данных выполняется внутри основной инфраструктуры и автоматически реплицируется на резервную площадку с ранее заданным периодом задержки. Между центрами существует связь в двух направлениях, позволяющая взаимное восстановление потерянных данных и восстановление работы операционного центра в период менее одного часа.
- «5» – Синхронная репликация. Выполняются требования 4-го уровня, но запись исходных данных в основном центре подтверждается только после записи в резервном центре. Благодаря этому количество потерянных данных (ADL) может быть равно нулю.
- «6» – Географический кластер. Процесс записи данных происходит одновременно в основном и резервном центрах. Это решение требует использования в обоих центрах такой инфраструктуры, которая позволила бы одновременное действие тех же приложений в двух удаленных местах. В случае сбоя основ-

ного центра все функции центра автоматически переключаются на резервный центр, где можно продолжить работу.



Начиная со второго уровня Disaster Recovery, для обеспечения работы системы восстановления требуется второй, или резервный, ЦОД, на который можно перевести обеспечение критически важных сервисов, когда основной ЦОД уже недоступен. В то же время наличие подобной схемы восстановления не отменяет необходимости использовать традиционное резервное копирование.

## ***Replication***

Переводится созвучно: репликация. Данный термин заслуживает более подробного описания. Процесс приведения данных на нескольких узлах к одному значению, например копирования с замещением с одного узла на другой. Различают синхронную и асинхронную репликации:

- **синхронная репликация** считается завершенной, когда изменения успешно распространились на локальной и всех удаленных системах;
- **асинхронная репликация.** В этом случае ответ о завершении записи приходит уже тогда, когда операция записи передается локальному устройству хранения. При этом система не ждет подтверждения от удаленного устройства хранения.

Проще говоря: при асинхронной репликации обновление распространяется на другие узлы спустя некоторое время. Таким образом, при асинхронной репликации вводится задержка, или время ожидания, в течение которого данные на отдельных узлах могут быть неидентичными. Такой режим может использоваться, например, для систем с медленными или нестабильными каналами передачи данных.



Чем выше уровень Disaster Recovery, тем больших капиталовложений он потребует. При этом необходимо учесть не только закупочную стоимость оборудования и услуг по предоставлению ЦОД, но и стоимость обслуживания, например оплату электроэнергии, арендную плату, сервисное обслуживание работающего оборудования и т. д. Именно поэтому так важно найти точку равновесия для бизнеса между затратами на отказоустойчивость и размером финансовых потерь при аварии.

## **2.6. Специальные термины для резервного копирования**

Внимательный читатель заметит, что тема резервного копирования проходит красной нитью через многие разделы. Действительно, без надежной резервной копии невозможно создать отказоустойчивую систему.

На 1–2-м уровнях Disaster Recovery восстановление из резервной копии является единственным способом вернуть к жизни данные.

Но даже при наличии более мощных мер защиты, таких как организация резервного ЦОД в случае антропогенной аварии, то есть вызванной в результате целенаправленных действий человека, восстановить данные можно только из резервной копии. Гарантированным средством от всех взломов, компрометирующих данные, катастрофических ошибок системных администраторов, некорректных изменений в данных, сделанных пользователями, может быть только резервная копия.

Резервное копирование – это своего рода иммунитет ИТ-инфраструктуры к возможным «болезням» и повреждениям. Если его нет, то любая, не слишком серьезная опасность может привести к колоссальным последствиям.

Но за все приходится платить, в том числе и за возможность иметь резервную копию. Цена вопроса заключается не только в необходимости покупать оборудование и носители информации. Дело в том, что сам процесс резервного копирования достаточно ресурсоемкий и наносит ощутимый удар по быстродействию системы. Для решения этой проблемы можно пойти двумя путями: использовать оборудование с расчетом возрастающей нагрузки или попытаться равномерно перераспределить нагрузку в течение нужного временного интервала.

Таким образом, появляется несколько важных терминов, о которых пойдет речь ниже.

### **2.6.1. Backup window**

Дословно переводится как «окно бэкапа». Время минимальной нагрузки на систему или сервис. Например, если организация работает по «пятитдневке», то «окном бэкапа» будут выходные дни, а также ночное время по будням (см. рис. 2.3). Для предприятий, работающих по другому графику, график распределения таких «окон» может быть совершенно иной.

### **2.6.2. Частота выполнения резервных копий**

Этот показатель напрямую влияет на RPO, то есть, по сути, устанавливает временную метку, на которой могут быть восстановлены данные. То есть если посуточно выполняется инкрементная копия, то максимальный срок потери данных – 1 сутки (если сбой произойдет прямо перед следующим выполнением задачи копирования).

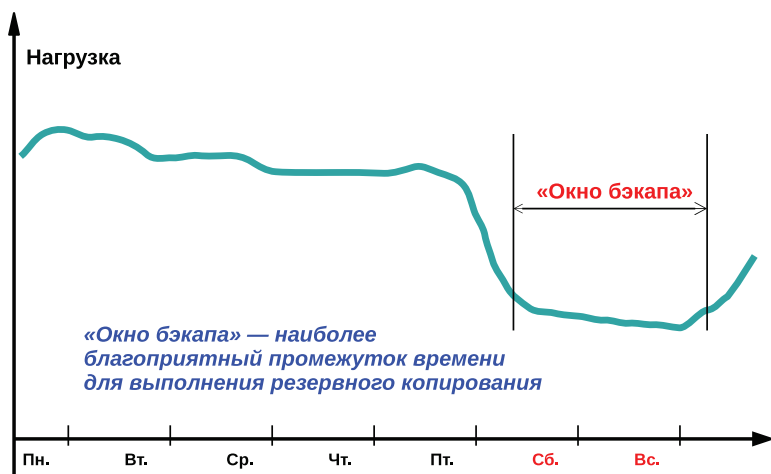


Рис. 2.3. Backup window – «окно бэкапа»

### 2.6.3. Глубина хранения

Время, за которое необходимо хранить резервные копии. Соответственно, чем длиннее срок, тем больший объем нужно зарезервировать: закупить дополнительные съемные носители, приобрести более объемное дисковое хранилище и т. д. Этот показатель влияет на общую сумму инвестиций, которые необходимо вложить в создание системы резервного копирования.

### 2.6.4. Off-site

Переводится как «за периметр». Это метод перемещения резервных копий из основного местоположения ИТ-инфраструктуры в безопасное место. Для этой процедуры обычно используется перемещение съемного носителя с резервной копией в безопасное хранилище или копирование данных по сети. Каждый из этих методов имеет свои преимущества и недостатки. В случае, когда требуется высокий уровень безопасности отказоустойчивости, целесообразно использовать оба метода. Например, данные для восстановления критичных сервисов копируются по сети на удаленный ЦОД, а менее значимые данные и архивные копии переносятся на носителях.

## 2.7. Системы высокой доступности

Для поддержания критичных высоконагруженных сервисов в рабочем состоянии при круглосуточном режиме работы недостаточно иметь свежую резервную копию. Необходимо применить принцип избыточности, когда нагрузка с неработающего компонента системы может быть легко переведена на резервный без остановки работы сервиса. Для описания таких систем используется следующая терминология.

### 2.7.1. High Availability cluster, HA-cluster

Этот термин обозначает кластер высокой доступности. Группа серверов (два и более), связанных специальным механизмом контроля, позволяющим с минимальной задержкой перевести обслуживание клиентов на резервный узел.

HA-кластеры и любые другие системы высокой доступности могут иметь следующую архитектуру:

- Active-Passive – активный/пассивный. Активный узел несет основную нагрузку, а пассивный находится в ожидании. При аварийной ситуации происходит смена ролей, что позволяет выполнить ремонт активного узла. Для обеспечения автоматического переключения используется специальная система внутреннего оповещения, например через дополнительные сетевые соединения;
- Active-Active – активный/активный. В этом режиме обслуживанием запросов занимаются все доступные узлы, в случае отказа одного нагрузка перераспределяется между оставшимися;
- с модульной избыточностью. Применяется только в случае, когда предыдущие два варианта не обеспечивают требуемого уровня отказоустойчивости. Все узлы одновременно выполняют одинаковые операции (либо фрагменты общей операции, но с условием, что результат гарантирован и при отказе любого узла), из результатов берется любой. Необходимо гарантировать, что итоги работы разных узлов всегда будут идентичны (либо имеющиеся различия не являются существенными и не влияют на дальнейшие результаты).

Стоит отметить, что только кластер с модульной избыточностью способен гарантировать нулевой простой оборудования. Для всех остальных существует так называемая «точка недоступности» – время, когда осуществляется перевод с вышедшего из строя узла на работающий.

### 2.7.2. Primary-Standby

Можно перевести как «основной–резервный». Данный метод организации отказоустойчивости в общих чертах напоминает Active-Passive-кластер, за исключением следующих отличий: в качестве узлов схемы используются полноценные серверы со свежей версией всех необходимых данных, актуальность и целостность которых поддерживаются периодической репликацией. В случае выхода основного узла весь функционал переходит к резервному. По этому принципу работают системы Disaster Recovery от 3-го уровня и выше.

## 2.8. Заключение

В данной главе приведен хотя и подробный, но далеко не полный список всех терминов, встречающихся при проектировании отказоустойчивых ИТ-систем. Но, несмотря на кажущуюся сложность этого вопроса, понимание логики работы и наличие здравого смысла – вот практически и все, что требуется для создания непотопляемых ИТ-инфраструктур.

# Глава 3

## Избыточность на службе отказоустойчивости

*– Помогите, у меня все пропало!*

*– Что пропало?*

*– Ну все, в-а-а-ще: погодите, во,  
уже появилось!*

*– Что появилось?*

*– Все появилось!*

(Из архива записей  
«Горячей линии техподдержки»  
крупной компании)

### 3.1. Об этой главе

Данная глава замышлялась как продолжение предыдущей. Первоначально я думал, что будет достаточно просто перечислить указанные ниже системы и этого хватит, чтобы ввести читателя в курс дела. Но по мере того, как я стал собирать и обрабатывать материал, на меня нахлынули воспоминания о том, как я сам создавал дисковые системы и сколько крови у меня попили те или иные «небольшие особенности».

В итоге возникло стойкое желание отойти от знакомой структуры «термин – пояснение» и попытаться рассказать читателю о методах отказоустойчивости в дисковых подсистемах так, как это есть на самом деле.

«Невозможно объять необъятное», – писал Козьма Прутков. В случае с данной главой это не совсем так, но подробное описание всего, что применяется на этом поприще, займет не одну главу и, дай бог, поместится в один книжный том размером с энциклопедию.

В то же время огромное количество алгоритмов, программ и даже устройств, которые призваны трудиться на благо в отказоустойчивости, читатель, скорее всего, так никогда и не встретит. Какие-то разработки так и не вышли из разряда теоретических, какие-то стандарты так и остались на бумаге, какие-то устройства уже морально устарели и «вышли из моды», какие-то методы оказались менее эффективными, чем другие...

Поэтому я принял простое решение – не забивать голову читателя вещами, которые, скорее всего, ему не понадобятся, и сосредоточиться на том, что он может повстречать каждую минуту.

## 3.2. Обозначения и расчеты

Прежде чем мы двинемся дальше по пути более подробного описания дисковых томов на основе RAID, необходимо определиться с некоторыми обозначениями.

Для расчета полезного объема  $V$  будем применять формулы, например:

$$V = N * C / 2,$$

где  $N$  – количество дисков в массиве, а  $C$  – их емкость.

## 3.3. Подробнее о RAID-массивах

Как я уже писал выше, существует масса решений, которые на практике редко используются. В связи с этим остановимся только на самых популярных решениях в этой области. На том, что любой системный администратор, инженер и простой пользователь могут встретить внутри сервера, дискового хранилища и другого устройства.



Стандарты современных дисковых массивов описаны в Common RAID Disk Drive Format (DDF), выпущенном Storage Networking Industry Association (SNIA). Для получения подробной информации я очень рекомендую ознакомиться с этим документом.

### 3.3.1. RAID-0

В случае RAID-0 два или несколько жестких диска объединяются в один массив, равный суммарному объему дисков. За счет того, что разные фрагменты данных в режиме чередования записываются на разные дисковые накопители, повышается скорость чтения-записи. При этом выход из строя одного диска влечет за собой неработоспо-

способность всего массива. И чем больше дисков собрано в RAID, тем ниже его надежность. По сравнению с этим, использование отдельных дисков дает большую отказоустойчивость за счет диверсификации.

Данный массив – всего лишь средство для объединения дисков для повышения скорости за счет чередования, а также увеличения объема дискового пространства.

В «чистом виде» он обычно используется для экспериментальных систем, когда можно пожертвовать надежностью во имя производительности.

Расчет полезного объема производится по формуле:

$$V = N * C.$$

Минимальное количество дисков – 2.



В некоторых системах можно создать массив RAID-0 из одного диска. Это делается, чтобы «обмануть» RAID-контроллер, в котором отсутствует возможность работать с дисками напрямую, без образования массива (JBOD).

### 3.3.2. RAID-1

Данный тип дискового массива является наиболее распространенным за счет своей простоты и хорошей отказоустойчивости. Он поддерживается практически всеми RAID-контроллерами. Это два или более дисков, информация на которых полностью дублируется в момент записи. Именно поэтому такие массивы также называют «зеркальными». При применении RAID-1 скорость выполнения операций чтения удваивается за счет чередования дисков (то есть блоки информации считываются поочередно то с одного, то с другого диска). В то же время возможно уменьшение скорости операций записи из-за рассинхронизации. В самом худшем варианте операции чтения-записи на каждый диск выполняются не параллельно, а подряд, одна за другой, тем самым удваивая общее время.

При использовании в одном массиве различных дисков скорость записи определяется самым медленным из них. Зато данный тип массива легко подлежит модернизации за счет установки более производительных дисков. RAID-1 хорошо подходит для размещения небольших файлов с последовательной записью, например журнала транзакций СУБД.

Расчет полезного объема производится по формуле:

$$V = 1 * C.$$

Минимальное количество дисков – 2.

### **RAID-1 хорошо подходит для ситуаций:**

- когда используется недорогой RAID-контроллер с небольшим количеством функций и поддержка RAID-1 – одно из немногих его достоинств;
- когда необходимо получить высокую надежность при небольших затратах;
- когда имеются конструктивные ограничения на количество устанавливаемых дисков.

Ограничения использования массива RAID-1: размер массива ограничен размером меньшего из дисков.

Другое ограничение заключается в скорости обмена данными. Если нужно получить скорость чтения записи больше, чем скорость одного винчестера, или скорость чтения больше двух скоростей винчестера, необходимо использовать другие типы RAID (например, RAID-10 для ускорения операций чтения-записи или RAID-5 для ускорения операций чтения).

### **3.3.3. RAID-1E**

Это модификация стандартного «зеркала» (RAID-1). Буква «Е» означает «Enhanced» («улучшенный»). Система работает по принципу чередования блоков по всему массиву, впоследствии сдвигая «чередование» на один диск. Создается минимум на трех дисках.

Основным отличием является то, что для обеспечения отказоустойчивости производится дублирование отдельных блоков, а не жестких дисков целиком. Это позволяет распределить функцию зеркалирования на нечетное количество дисков.

Основное преимущество перед RAID-1 – возможность использования нечетного числа дисков. При использовании четного числа дисков лучше, конечно, применять RAID-10.

Результирующая емкость массива с использованием RAID-1E вычисляется по формуле  $N * C/2$ .

Минимальное количество дисков для данного типа массива – 3 (разумеется, можно использовать 2 диска, но тогда массив RAID-1E превратится в обычный RAID-1).

Положительными сторонами являются:

- хорошая скорость передачи данных и обработки запросов (но не забываем про задержки при рассинхронизации записи, свойственные схемам, реализующим зеркалирование на недостаточно совершенном оборудовании или ПО);
- хорошая устойчивость, стойкость при отказе.

Ограничением для использования является все тот же нерациональный расход объема жестких дисков на обеспечение отказоустойчивости (полезный объем равен половине от общего объема).

### 3.3.4. RAID-5

В отличие от простого зеркалирования дисков и отдельных блоков, как, например, в RAID-1, RAID-10 и RAID-1E в RAID-5, применяется более сложная технология, основанная на подсчете и записи контрольных сумм (см. рис. 3.1). Это позволяет восстановить информацию в случае выхода из строя одного из дисков. В то же время на создание контрольных сумм требуются дополнительные ресурсы, возрастает нагрузка на аппаратное обеспечение. В связи с этим дисковые массивы на основе RAID-5 с операциями записи могут работать медленнее, чем их собратья на основе зеркалирования RAID-1, RAID-10 и RAID-1E или одиночные диски.

Расчет полезного объема также производится по формуле

$$V = (N - 1) * C.$$

Минимальное количество дисков – 3.

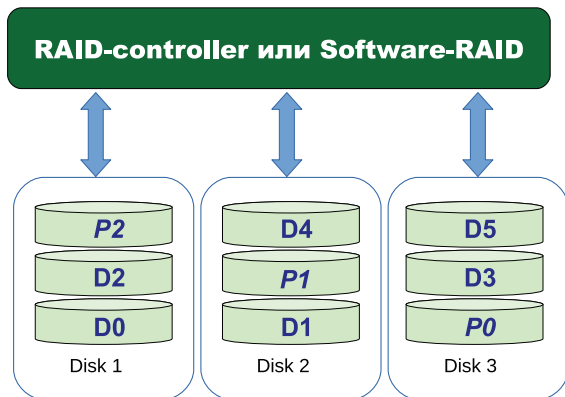


Рис. 3.1. Принцип организации RAID5

В то же время при операции чтения с RAID-5 будет наблюдаться большая производительность, по сравнению с одиночным диском, за счет того, что одновременно работают все те же  $N - 1$  дисков (диск, на котором в данный момент располагается контрольная сумма, не задействован).

Но основное преимущество RAID-5 – самые низкие потери от общего объема дисков. Например, при создании массива из 4 одинаковых жестких дисков реальный объем будет равен объему 3 накопителей. Для сравнения: RAID-10 при тех же условиях позволит создать массив, равный 2 жестким дискам.

Современные RAID-контроллеры позволяют минимизировать накладные расходы при расчете и записи контрольных сумм за счет кэширования и дополнительных вычислительных средств, встроенных в аппаратный контроллер. В то же время при реализации более простых контроллеров и тем более программных решений применение RAID-5 все также может нанести ощутимый урон быстродействию при выполнении большого числа операций записи.

#### **RAID-5 хорошо подходит для ситуаций:**

- когда требуется высокая производительность при операциях чтения и операции записи составляют не более 10% от всего объема;
- когда требуется надежное и очень экономичное решение для создания дискового массива большого объема, при этом требования к производительности находятся на втором плане.

#### **Ограничение использования массива RAID-5:**

- когда требуется высокая производительность при операциях записи данных;
- когда конструктивные особенности не позволяют использовать большое число дисков.

### **3.3.5. RAID-6**

Технология RAID-6 является дальнейшим развитием технологии RAID-5.

Основой повышения отказоустойчивости является то, что контрольная сумма рассчитывается и хранится дважды, при этом расчет производится с применением разных алгоритмов и для разных наборов блоков. Один и тот же блок с данными фигурирует в расчете двух различных контрольных сумм. Таким образом, если выйдут из строя два жестких диска, можно восстановить часть данных с применением первой контрольной суммы, а потом, используя только что восстановленные блоки и вторую контрольную сумму, восстановить остальную информацию.

Пережить потерю сразу двух жестких дисков является основным преимуществом RAID-6.

Разумеется, за все приходится платить, в данном случае уменьшением объема и потерей быстродействия.

Расчет полезного объема также производится по формуле

$$V = (N - 2) * C.$$

Минимальное количество дисков – 4.

### 3.3.6. Составные, или сложные, массивы

Несмотря на низкую надежность, у RAID-0 есть и «вторая профессия». Он используется для создания «сложных» массивов. Для этого два и более дисковых массива, предварительно созданных по более простым технологиям, дополнительно объединяются еще в один RAID. Это дает дополнительный выигрыш в быстродействии.

Самым простым вариантом составного массива является RAID-10.

Еще одним перспективным решением выглядит RAID-60, который, соответственно, состоит из двух массивов RAID-6, объединенных дополнительно в RAID-0. Это позволяет минимизировать снижение быстродействия из-за высоких накладных расходов RAID-6.

### 3.3.7. RAID-10

RAID-10, по сути, является синтезом RAID-1 и RAID0. То есть сначала диски зеркалируются в массивы RAID-1, после чего объединяются в массив RAID-0.

Расчет полезного объема также производится по формуле

$$V = N * C / 2.$$

Минимальное количество дисков – 4.

Несмотря на то что RAID-10 является прямым наследником RAID-1 и RAID-0, он обеспечивает отказоустойчивость выше, чем RAID-1. Массив из 4 дисков способен пережить выход из строя двух HDD при условии, что они не находятся в одном «зеркале» RAID-1 (см. рис. 3.2).

При эффективной работе RAID-10 его производительность операций записи выше в 2 раза, чем у одиночного диска, за счет чередования, а операции записи – выше в 4 раза.

Преимущества RAID-10:

- высокая производительность;
- высокая надежность.

Ограничение применения RAID-10: так как RAID-10 состоит из RAID-1, его недостатком является все тот же нерациональный расход HDD.



Рис. 3.2. В случае выхода из строя по одному HDD из различных «зеркал» (mirror) работоспособность RAID-10 сохраняется

## 3.4. Программный или аппаратный RAID?

Традиционно различают два метода организации дисковых томов на основе RAID.

### 3.4.1. Программный RAID

Наиболее простое и недорогое решение. За все операции отвечает специальный драйвер операционной системы. На диске создается контрольная область, в которую записывается информация о принадлежности винчестера к тому или иному массиву, а также информация о типе массива, роли данного HDD в нем (например, в случае RAID-3 это может быть диск с данными или диск для контрольных сумм) и т. д.

Несомненным преимуществом данного метода является возможность обойтись подручными средствами, ведь для организации дискового массива нужен только компьютер с операционной системой.

Еще одним преимуществом является легкость увеличения объема массива путем поочередной замены существующих более вмести-тельных винчестеров вместо уже установленных. Например, имея RAID-10, состоящий из дисков по 1 Тб, можно поочередно заменить их на диски 2 Тб, тем самым удвоив доступный объем.

В то же время у программных RAID отсутствует возможность наращивать объем простым добавлением нового диска в уже созданный массив.



Разумеется, заменять диски нужно строго по одному, и каждый раз обязательно нужно дождаться полной перестройки массива. В противном случае вместо усовершенствования можно легко разрушить дисковый массив.

Основное ограничение software-RAID заключается в том, что вся нагрузка по обслуживанию RAID: вычислению контрольных сумм, организации процесса зеркалирования и т. д. – ложится на центральный процессор. В зависимости от объема массива и типа RAID это может нанести ощутимый удар по производительности системы. Именно поэтому software-RAID обычно применяют для простых решений на основе зеркалирования или чередования: RAID-0, RAID-1, RAID-10.

Иногда возникают трудности при использовании программно-го RAID в качестве системного диска. Например, когда операционная система изначально «не понимает», что в сервере присутствует software-RAID.

### 3.4.2. На основе аппаратного RAID-контроллера

В данном случае основная роль принадлежит специальному устройству – RAID-контроллеру, к которому и подключаются жесткие диски. Он имеет собственный вычислительный модуль и специальное программное обеспечение, что в совокупности позволяет работать с различными уровнями RAID без вмешательства процессора.

Всю работу по разметке дисков, вычислению контрольных сумм берет на себя именно контроллер.

Для операционной системы контроллер с подключенными винчестерами выглядит как один жесткий диск, с которым она может работать обычными средствами.

Разумеется, для того чтобы система «увидела» данную конструкцию, необходимо использовать специальный драйвер. Операционную систему можно сразу установить на аппаратный RAID, просто подгрузив драйвер при установке. (Драйверы на наиболее популяр-

ные устройства, как правило, уже содержатся в ядре операционной системы.)

Для повышения производительности многие RAID-контроллеры имеют собственную кэш-память для ускорения процессов чтения/записи.

В то же время аппаратный RAID имеет серьезное ограничение – нельзя увеличить объем поочередной установкой более вместительных дисков. В то же время в массив можно добавлять дополнительные диски.

Цена такого устройства довольно высока. Многие RAID-контроллеры уровня Enterprise продаются по цене целого сервера.



Существует семейство самых дешевых RAID-контроллеров, которые не содержат собственных вычислительных и программных средств, а все операции по обслуживанию выполняет драйвер, который задействует для этого операционную систему, центральный процессор, оперативную память и т. д. Обычно этим «грешат» контроллеры, встраиваемые в материнскую плату, так называемые Embedded RAID. Из-за того, что драйверы для дешевых устройств пишутся по технологии «тяп-ляп, лишь бы заработало», очень часто такой «бутерброд» работает нестабильно. Например, из массива пропадают жесткие диски, или сам массив время от времени становится недоступным.

### 3.4.3. Программный или аппаратный RAID – что выбрать?

Однозначного ответа здесь быть не может, в каждом случае необходим индивидуальный подход. Есть несколько популярных рекомендаций на этот счет:

- если необходимы надежность и скорость работы – стоит отдать предпочтение аппаратному RAID-контроллеру;
- если требуется экономичное масштабируемое решение – имеет смысл подумать о применении программного метода;
- лучше отказаться от использования Embedded RAID на критичных участках, установив в систему «настоящий» RAID-контроллер или просто отдав предпочтение более предсказуемому software-RAID.

### 3.4.4. Технология RAID+

У этой технологии много названий. Помимо «RAID+», в реинкарнации от некоторых вендоров она же называется «RAID 2.0+», «виртуальный RAID» «RAID Next generation» («RAID следующего поколения») и т. д. Основной смысл – подчеркнуть новизну подхода и некоторый абстрактный характер работы данного решения.

В отличие от стандартного RAID-массива, в котором жесткие диски сразу объединяются в логический массив, в «RAID+» это выглядит несколько сложнее.

Первоначально все пространство дисков объединяется в единое целое. Полученный объем делится на небольшие сегменты – *chunks*. (Следует заметить, что такие отдельные сегменты выделяются и при организации традиционных RAID.) В свою очередь, *chunks* дробятся на еще более мелкие единицы – экстенты (*extents*), из которых впоследствии и собираются логические тома (*logic volumes*) (см. рис. 3.3).

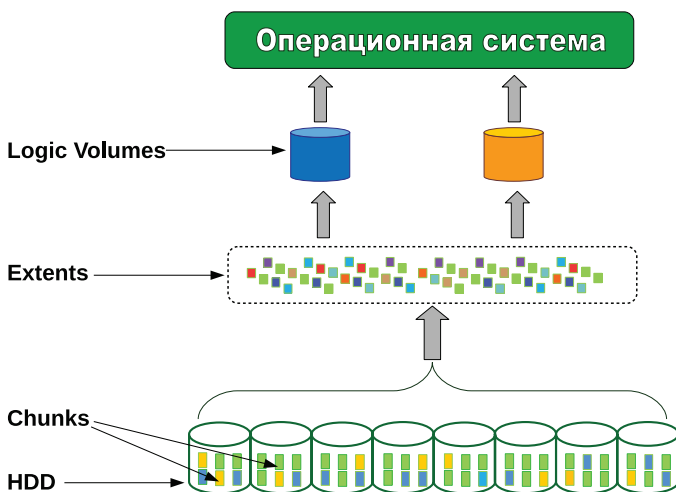


Рис. 3.3. Принцип организации RAID+

Какие это дает преимущества?

Во-первых, все диски работают одновременно. Практически исключается ситуация, когда, например, сначала одни данные записываются на один диск, потом другие на второй диск, а далее вычисляется и записывается контрольная сумма на третий винчестер. Или когда в RAID-5 в один квант времени данные могут одновременно считываться со всех HDD, кроме одного, на котором расположена контрольная сумма.

При внутренней организации устройства, как в RAID+, гораздо проще организовать режим параллельного обращения к жестким дискам. Значительно облегчаются функции кэширования, предвари-

тельной обработки и т. д. В момент перестройки диска, например при расширении массива, одновременно обрабатываются все диски, что значительно ускоряет завершение процесса, создающего для жизни массива дополнительную опасность.

Данную технологию поддерживают многие современные системы хранения от таких производителей, как EMC, Huawei. Некоторые современные файловые системы позволяют строить RAID-массивы с применением схемы работы, схожей с «RAID+». Примером может послужить RAID-Z на основе ZFS.

## 3.5. Методы защиты информации от ошибок при записи

Дисковые массивы с применением избыточности – это поистине замечательная вещь. Однако они не в состоянии защитить информацию от самой главной опасности: некорректной работы программного обеспечения или ошибок пользователя. Такие разрушительные действия с точки зрения системы являются абсолютно легальными. Например, полное «обнуление» базы данных с точки зрения сервера СУБД – вполне урядная операция, хотя последствия с точки зрения сохранения информации могут быть просто катастрофические. Для того чтобы этого избежать, применяют специальные алгоритмы защиты при копировании.

Подобные методы используются для обеспечения отказоустойчивости многих процессов, например работа с памятью, файлами на диске, таблицами в базах данных и т. д. Но наиболее часто они применяются при создании и работе с «мгновенными снимками» (snapshots). Примечание. *Snapshot* (переводится как «мгновенный снимок») – это фрагмент файловой системы на определенный момент времени.

### 3.5.1. Копирование при записи (Copy-On-Write)

Главная идея copy-on-write – создавать копию изменяемой информации, но только тогда, когда система обращается к этим данным с целью записи.

Этот метод получил большое распространение при создании новейших файловых систем, таких как ZFS и Btrfs. Благодаря ему создание снимков в данных системах происходит практически мгновенно, не занимая при этом больших ресурсов носителя информации.

Рассмотрим основные принципы работы данного алгоритма.

Когда copy-on-write используется впервые:

- исходная рабочая область функционирует «как есть» (то есть работа приложений с диском не приостанавливается);
- на логическом уровне выделяется резервная область для размещения данных, которая изначально не содержит никакой информации и состоит из ссылок на оригинальную область хранения, например на исходную файловую систему.

При операции записи в исходную файловую систему выполняются следующие шаги:

- содержимое блока «A1» исходной файловой системы копируется в блок «B1» резервной области (см. рисунок);
- производится запись новой информации в блок «A1» оригинальной файловой системы.

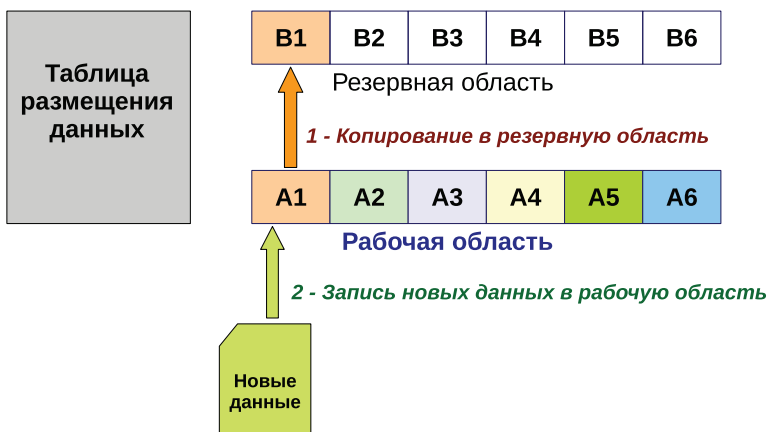


Рис. 3.4. Схема записи copy-on-write

При этом в оригинальной области всегда хранится наиболее актуальная информация, и процедура чтения производится напрямую с оригинальной файловой системы, без каких-либо накладных расходов по времени.

Дополнительно следует отметить ряд нюансов:

- процесс чтения данных из резервной области (например, во время выполнения резервного копирования) обычно приводит к некоторому снижению производительности основной файловой системы;
- производительность такой системы ограничена скоростью записи в резервную область. Фактически производятся две

операции записи вместо одной, что может значительно снизить скорость работы. Например, тот самый случай, когда для высоконагруженных серверов были приобретены высокопроизводительные SSD-диски, а для области резервного копирования установлены винчестеры подешевле, например HDD SATA. Представьте себе картину: сначала старые фрагменты перезаписываются с быстрых дисков SSD на медленные SATA, а уже потом записываются свежие данные на SSD. В такой конфигурации скорость работы заметно снизится;

- необходимо понимать, что за надежность приходится платить не только объемом, но и быстродействием, и учитывать подобные нюансы.

### 3.5.2. Перенаправление при записи (Redirect-On-Write)

Данный алгоритм похож на copy-on-write, за исключением самой процедуры записи. При работе redirect-on-write новая информация не записывается в оригинальное местоположение соответствующего блока данных, а заносится в резервную область, после чего просто изменяется ссылка. При этом исходная область остается без изменений.

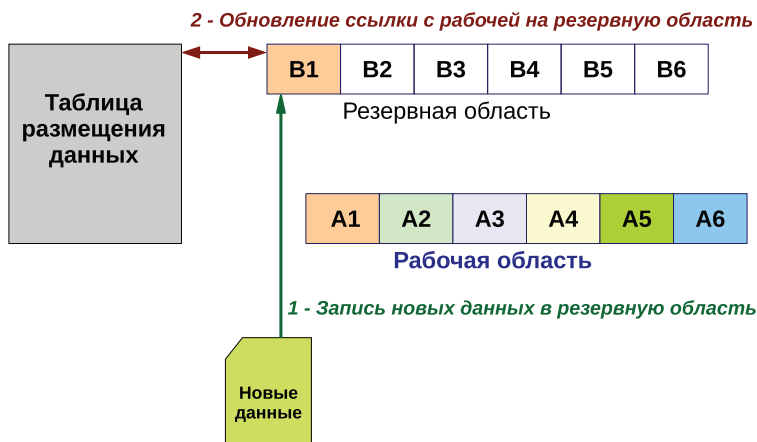


Рис. 3.5. Работа redirect-on-write

Преимуществом этого метода является более высокая общая производительность при записи, так как не требуется дополнительно выполнять копирование старых данных, как в Copy-On-Write. Однако некоторые накладные расходы, связанные с формированием ссылок

на новые фрагменты и внесением операции в таблицу размещения данных, все-таки присутствуют.

Ограничением `redirect-on-write` является необходимость копирования больших объемов данных в момент удаления снимка и объединения его блоков с оригинальной файловой системой. Иногда быстродействие системы падает настолько, что она вообще перестает отвечать на запросы. Кроме того, при работе `redirect-on-write` снижается скорость операций чтения, так как запрашиваемую информацию нужно проверить в обновляемой области и только потом, при ее отсутствии, обратиться к оригинальной области хранения данных.

## 3.6. Заключение

Создание отказоустойчивых решений даже на таком начальном уровне, как дисковый массив, — задача совсем несложная. Однако за все приходится платить. В данном случае ценой вопроса становятся затраты на дополнительное оборудование и снижение быстродействия. Но если принимать во внимание ценность данных, размещенных на системе хранения, то введение некоторых элементов избыточности может показаться вполне разумной мерой.

# Глава 4

## Архитектура систем резервного копирования

*Библейский Ной был первым администратором резервного копирования. Он же был и остается самым выдающимся специалистом по этому вопросу. Сделать резервную копию генофонда всех полезных живых существ – это вам не базу 1С на флэшку передать.*

(Из юмора администраторов  
резервного копирования)

Очень часто возникает ситуация, когда существующая система резервного копирования по тем или иным характеристикам не отвечает нуждам предприятия. И проблема чаще всего кроется в «первородном грехе», потому что с самого начала были допущены промахи в разработке архитектуры. Бывает и другая ситуация, когда разработанная система первоначально полностью удовлетворяет все нужды ИТ-инфраструктуры предприятия. Но бизнес, как живой организм, растет, и ИТ-инфраструктура меняется вместе с ним. Рано или поздно все равно потребуется что-то модернизировать, чтобы привести в соответствие быстрорастущую инфраструктуру и консервативную по своей природе систему резервного копирования.

### 4.1. Топология резервного копирования

Прежде чем начать рассказ о топологии, давайте определимся с самим термином. Наверняка уважаемый читатель, помимо этой книги, уже читал какие-то материалы в Интернете и встречал похожий термин. Самое удивительное, что под словами «топология резервного копи-

рования» готовы понимать все, что угодно. Например, топологию локальной сети. Или в каком помещении расположен тот или иной сервер. В общем, порой под этим термином понимается все, что угодно, только не то, что имеет прямое отношение к резервному копированию.

На самом деле топология резервного копирования в ее первоначальном понимании является основным фактором, который влияет на архитектуру и метод работы.

Существуют два основных типа топологии систем резервного копирования: децентрализованная и централизованная.

#### 4.1.1. Децентрализованная топология

Представляет собой архитектуру, при которой на каждый объект резервного копирования (например, сервер, содержащий ценные данные) устанавливается полная самостоятельная копия программы для сохранения данных. Прекрасным примером такой топологии является использование встроенных средств в операционную систему MS Windows: `ntbackup` для поколения Windows Server 2003 и `MS Windows BackupService` для поколения Windows Server 2008. Эти решения предполагают, что предустановленное программное обеспечение собирает с сервера данные, указанные в настройках задания, и создает их копию на удаленном хранилище (см. рис. 4.1).



Рис. 4.1. Децентрализованная топология резервного копирования

### 4.1.2. Централизованная топология

Данный вариант представляет собой более сложное решение. В этом случае на специально выделенный сервер устанавливается центральный программный модуль, через который осуществляется управление сбором данных, а на объекты, подлежащие резервному копированию, устанавливаются специальные программы-агенты (см. рис. 4.2). Для хранения может использоваться как файловый сервер, так и сетевое хранилище. Если ИТ-инфраструктура строится на базе решений от Microsoft, то очень хорошим выбором может оказаться для такого файл-сервера одна из бесплатных UNIX-like операционных систем, например, Linux или BSD, с установленным Samba или FTP-сервером. В этом случае различие в платформах может послужить дополнительной гарантией безопасности в случае вирусного заражения или инсайдерской атаки.



Рис. 4.2. Централизованная топология резервного копирования

### 4.1.3. Псевдоцентрализованная топология

Иногда системные администраторы небольших компаний пытаются организовать централизованную схему резервного копирования наоборот. Когда не программы-агенты передают данные сохранения, а бэкап-программа на центральном сервере забирает данные серверов, используя

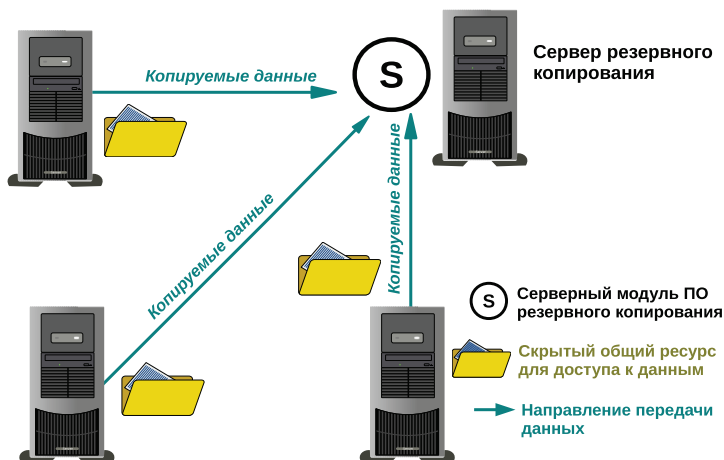


Рис. 4.3. Псевдоцентрализованная топология

уже существующие службы и сервисы. Например, «выгребает» данные из скрытых общих папок на Windows-серверах. Или соединяется по протоколам SFTP/SCP с серверами под управлением Unix-систем.

В качестве примера backup-программы, использующей такую коварную топологию, можно привести opensource-проект Backup PC.

Казалось бы, все прекрасно, сисадмин выкрутился, сэкономив кучу средств. Но в действительности он, сам не зная того, подложил в систему мину замедленного действия. Дело в том, что такая, с позволения сказать, псевдоцентрализованная топология имеет весьма существенные ограничения. В первую очередь это проблема сохранения файлов, открытых для записи. И в конце концов наступит тот прекрасный момент, когда открытые файлы будут либо пропущены и не попадут в резервную копию, либо будут скопированы с ошибками. Причем когда именно это произойдет, предсказать бывает довольно трудно. Например, пользователь всегда перед уходом домой закрывал файлы, а в один прекрасный день оставил рабочую программу открытой. Существуют различные методы обхода данной проблемы, например повторный запуск задания с целью скопировать только пропущенные файлы, но нет ни одного надежного. Поэтому такая схема подходит для применения исключительно в небольших организациях, работающих в режиме 8×5, с дисциплинированными сотрудниками, которые сохраняют изменения и закрывают файлы перед уходом домой. Вы часто встречали такие организации в реальной жизни? Вот и я не встречал. Поэтому не применяйте такую топологию!

#### 4.1.4. Смешанная топология

Помимо вышеописанных, существует смешанная топология, когда используются элементы централизованной и децентрализованной составляющей. Обычно это применяется, когда специализированные агенты не в состоянии обеспечить резервное копирование данных всех сервисов и приходится использовать такого рода ухищрения. Классическим примером являются предварительная выгрузка данных базы SQL-сервера в промежуточный файл и его последующее копирование обычными средствами резервирования файл-сервера.



Рис. 4.4. Смешанная топология

На самом деле, несмотря на неказистый вид, данная схема гораздо предпочтительнее, чем децентрализованная, и уж точно лучше, чем псевдоцентрализованная. Смешанная топология позволяет уменьшить трудозатраты на большинство рутинных операций. Остается необходимость проверки отдельных участков, где применяются децентрализованные фрагменты, но все равно это уже гораздо проще. При этом нет риска остаться без резервной копии, как в случае с псевдоцентрализацией. Разумеется, самой лучшей рекомендацией при использовании смешанной топологии будет скорейший переход на централизованную. Поэтому данный вариант лучше рассматривать как переходный, например при поэтапной модернизации существующей

системы резервного копирования.

## 4.2. Уровни архитектуры систем резервного копирования

### 4.2.1. Архитектура для небольших инфраструктур

Не нужно быть особенно проницательным, чтобы предположить, что для небольших ИТ-инфраструктур малого и среднего бизнеса как раз и применяется децентрализованная схема. Чаще всего применяют схему с единой точкой сбора данных, например некое сетевое хранилище, куда программы резервного копирования, установленные на серверах, складывают резервные копии (см. рис. 4.5).

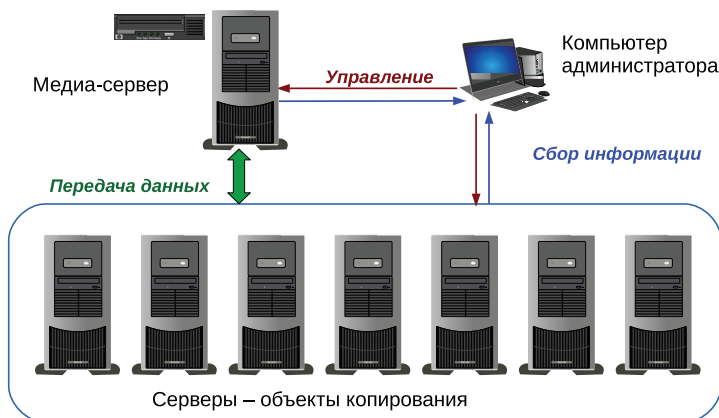


Рис. 4.5. Схема резервного копирования небольшого предприятия

На начальном этапе роль управляющего сервера, как правило, отсутствует, используется децентрализованная топология.

Функции администратора резервного копирования, как правило, ограничиваются установкой и настройкой специализированного программного обеспечения на серверах, подготовкой сетевого хранилища, внимательным составлением расписания, чтобы задания по возможности не пересекались между собой по времени.

У этой схемы есть единственное преимущество: дешевизна и возможность создания подобной системы резервного копирования из подручных средств. При этом существует масса минусов, таких как

трудности организации транспортировки резервной копии для хранения за территорию предприятия (off-site), а также сложности управления и масштабирования. Когда объектов станет слишком много, администратор резервного копирования уже не сможет управлять данной системой, просто подключаясь к каждому серверу напрямую. Кроме того, необходимость избегать одновременной работы большого числа заданий приводит к тому, что расчет времени и выявление таких «наложений» в расписания заданий вручную при большом количестве объектов практически невозможно.

В этом случае потребуется использование некоей единой консоли управления, что означает переход на централизованную топологию.

#### **4.2.2. Архитектура системы резервного копирования крупного предприятия**

Сейчас мы совершим небольшой скачок в будущее и, вместо того чтобы рассматривать вторую ступень после небольших предприятий – так называемый средний, или «middle», уровень, сразу перейдем к рассмотрению системы резервного копирования уровня большого предприятия (Enterprise). Такой лихой поворот мы сделали по одной причине: будет легче зайти с другого полюса, чтобы показать принципиальное различие между начальной и конечной ступенями.

Естественно, решения масштаба крупного предприятия принципиально отличаются от начального уровня. Поэтому если на малых инфраструктурах часто применяется децентрализованная топология, то на системах резервного копирования уровня Enterprise – обязательно централизованная. Нет смысла экономить на лицензиях программ-агентов резервного копирования, потому что потеря данных обойдется просто колоссальными убытками.

Изменяется и сама идеология построения системы. Обычно применяется трехуровневая схема. На самом верхнем уровне находятся объекты управления системой резервного копирования: выделенный сервер со специализированным ПО и консолью управления. Копия консоли может быть установлена на компьютер администратора резервного копирования для удобства работы. Именно с этого уровня обычно начинается установка системы резервного копирования по принципу сверху вниз (см. рис. 4.6).

Следующий «слой» – это уровень хранения резервных копий. На нем находятся так называемые медиа-серверы, или серверы хранения данных. Они используются для трех целей: подключение внешних накопителей и управление ими, непосредственное хранение резерв-



Рис. 4.6. Система резервного копирования крупного предприятия

ных копий в режиме файл-сервера, а также в качестве промежуточно-го, или, как его еще называют, кэш-сервера для временного хранения копии перед его постоянной записью на ленту. Использование такого сервера ускоряет процесс записи и восстановления данных и позволяет более экономно использовать «окно бэкапа» (предпочтительное время для запуска заданий резервного копирования). В рамках Enterprise решения управление резервными копиями, временными носителями производится через консоль управляющего сервера. Подчеркну: все операции, включая такие задания, как форматирование носителей, извлечение ленточных картриджей из считывающих механизмов и т. д. При этом даже если используемое оборудование позволяет самостоятельно управлять носителями, делать это крайне не рекомендуется, за исключением аварийных ситуаций. Проблема заключается в том, что программа управления резервным копированием может «не знать» о производимых действиях непосредственно на оборудовании и выдавать ошибку или выполнит нежелательные действия, например сотрет ценную резервную копию.

На самом нижнем – третьем – уровне находятся программы-агенты резервного копирования. Именно они собирают данные, которые мы хо-

тим сберечь, и заботливо складывают их на системах хранения. А само управление агентами производится, опять же из консоли предприятия.

Для чего нужен весь этот «огород»? Дело в том, что при разрастании инфраструктуры администратор резервного копирования не в состоянии вручную отследить все задачи, политики, номера носителей и другую информацию касательно всей системы резервного копирования. Вся эта рутина перекладывается на управляющую часть backup-системы. В ведении администратора остаются только функции настройки заданий и мониторинг событий. Все остальные задачи, включая управление носителями, оповещение о результатах выполнения заданий или восстановления, проверки целостности копии (и даже антивирусной проверки для некоторых систем) и т. д., возлагаются на автоматизированный центр управления системой резервного копирования уровня предприятия.

При такой организации можно легко делегировать полномочия, например назначив роль оператора резервного копирования, который может поменять записанные носители на подлежащие перезаписи, запустить-остановить задание, восстановить данные из копии, но не имеет прав на полное управление системой. Таким образом можно решить массу проблем, начиная от ухода в отпуск или на больничный администратора резервного копирования до освобождения высококвалифицированных специалистов для более эффективного использования человеческих ресурсов.

Иногда используются некоторые дополнительные элементы, например специализированные серверы для развертывания программ-агентов под ту или иную платформу, сетевые хранилища, серверы для мониторинга и построения отчетов, но их присутствие просто расширяет данную архитектуру в плане удобства использования и повышения эффективности. В основе так или иначе все равно остаются трехуровневая схема и централизованная топология.

На очень крупных предприятиях встречается распределенная система резервного копирования, при которой отдельные трехуровневые структуры объединены между собой в единую сеть взаимодействия. При этом один из управляющих серверов может быть центральным узлом, через который администратор резервного копирования контролирует всю распределенную инфраструктуру. Такая схема применяется в случае наличия у предприятия филиалов, а также при использовании резервного центра обработки данных (ЦОД). В этом случае можно даже говорить о некой четырехуровневой модели управления. Но такие «монстры» встречаются не так часто, в первую очередь из-за высоких затрат на подобную архитектуру.

### 4.2.3. Архитектура для среднего уровня

Как я уже писал выше, системы среднего уровня представляют собой некий промежуточный уровень между начальным уровнем и Enterprise-решениями.

Архитектура среднего уровня предполагает двухуровневую архитектуру. В отличие от уровня крупных предприятий, здесь уровень управления совмещен с медиа-сервером, а программы-агенты так и остаются на своем нижнем уровне (см. рис. 4.7).

На среднем уровне совмещают роли управляющего и медиа-сервера.

На среднем уровне предполагается использовать централизованную топологию, но часто из-за неполноты или недоработки программных продуктов приходится выкручиваться, организовав смешанный вариант.

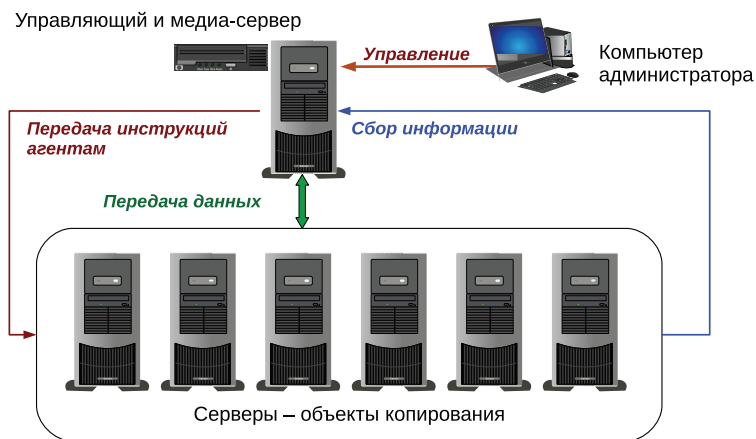


Рис. 4.7. Система резервного копирования среднего предприятия

При всех своих минусах из серии «ни рыба ни мясо» эти системы получили довольно широкое распространение, в первую очередь из-за ошибочной оценки их возможностей в более отдаленной перспективе. В то же время неоправданное применение подобных средств в малом бизнесе, в особенности на стартовом уровне, приводит к внушительным и часто совершенно ненужным расходам. Получается, что на стартовом уровне внедрение такой архитектуры вводит предприятие в ощутимые затраты, что побуждает его держаться до последнего за уже внедренное решение даже в ущерб актуальности и эффективности.

На мой взгляд, применение систем резервного копирования среднего уровня уместно, когда есть некая уверенность, что бизнес не вырастет больше определенного «потолка». Например, отдельный самостоятельный филиал крупной компании со своим штатом управления и специфичными бизнес-процессами. Или торговая сеть в небольшом городе вдали от транспортных артерий и крупных населенных пунктов.

Ниже рассматривается наиболее перспективный вариант для постепенного развития системы резервного копирования по принципу «от простого к сложному».

### 4.3. Проблема выбора при организации системы резервного копирования

Разумеется, идеальным решением является с самого начала приобрести специализированное решение резервного копирования от известного производителя, с централизованной топологией, возможностью делегирования полномочий бэкап-оператору и т. д.

К сожалению, бизнес, находящийся в стадии первоначального развития, может не располагать необходимыми финансовыми средствами для приобретения такой серьезной и дорогой системы.

Поэтому чаще всего в самом начале используют некую систему начального уровня с децентрализованной или смешанной топологией, построенную на бесплатном решении. Впоследствии планируется переход на систему резервного копирования более высокого уровня развития уже с централизованной топологией.

Если же говорить о выборе системы на момент такой модернизации, то я настоятельно рекомендую сразу сделать предпочтение в пользу Enterprise-решения. Да, вот так сразу: от системы резервного копирования начального уровня сразу до уровня Enterprise. Дело в том, что переход от решения среднего уровня до системы уровня крупного предприятия не менее сложен, чем любой другой. При этом нахождение на данном этапе может затянуться непростительно долго из-за расплывчатой границы между этими уровнями. Если уход от решений начального уровня происходит по причине врожденных недостатков децентрализованной топологии, проще говоря, когда администратор резервного копирования начнет терять управление своей системой, определить точную дату, когда необходим переход от среднего уровня до уровня крупного предприятия, бывает довольно трудно. Чаще всего администраторы резервного копирования и руководство ИТ действуют, исходя из соображений: «работает – не трогай» и «пока вроде все устраивает».

К сожалению, ждать, когда нынешняя система резервного копирования безнадежно устареет, чревато серьезными последствиями. Чем дольше затягивается «ожидание», тем вероятнее ИТ-инфраструктура останется без рабочей резервной копии в момент аварии.

Помимо всего прочего, чем позднее осуществляется переход с одного уровня на другой, тем дороже он обходится. В итоге все происходит по принципу «скупой платит дважды».

В то же время при покупке Enterprise-решения обычно не требуется производить некую массовую закупку лицензий для программ-агентов «на вырост», большого количества медиа-серверов и т. д. Практически для любой системы такого уровня можно вначале купить ядро системы резервного копирования из управляющей части, одного медиа-сервера и небольшого числа программ-агентов для критичных сервисов, а уже потом постепенно расширять архитектуру по мере роста ИТ-инфраструктуры в целом. В этом случае стоимость полноценного Enterprise-решения будет не намного выше покупки системы среднего уровня. Мало того, на данном этапе можно скрепя сердце пойти на компромисс и организовать вначале смешанную топологию, при которой копирование критичных сервисов выполняли бы специализированные агенты, а некритичных – встроенные средства резервного копирования. Главное – заложить нормальную масштабируемую архитектуру с учетом дальнейшего развития бизнеса.

Поэтому лучше не искушать судьбу и сразу внедрять систему крупного предприятия, как только бизнес покажет устойчивый рост и выйдет из состояния «стартапа».

## 4.4. Рекомендации по выбору системы резервного копирования

И вот настал тот светлый день и час, когда системного администратора вызывают к начальству и он слышит нечто вроде:

«Наша компания переходит на новый уровень, когда недопустимы даже небольшие простои работы. Поэтому надо подумать о том, как нам сделать нормальное резервное копирование. Не “на коленке”, как делается сейчас, то есть копия, то нет, а чтобы все было надежно. Подумай, может, нам стоит купить какую-то коммерческую систему, лучше раз потратиться, зато потом чтобы все работало».

О каких качествах будущей системы резервного копирования можно узнать из такого сообщения?

- Во-первых, предполагается рост бизнеса, следовательно, увеличится объем копируемых данных. При этом данный рост

может происходить длительное время, что потребует придания системе функций масштабирования.

- Во-вторых, возросли требования к надежности системы. Предполагается, что новая система обеспечит безошибочное копирование и восстановление данных, а также сама не будет сбоить во время работы.
- И в-третьих, данная система должна работать достаточно быстро, чтобы свести время простоя к минимуму.

Чтобы удовлетворить все три условия, необходимо выбрать систему резервного копирования с централизованной топологией, использовать сочетание двух систем: Disaster Recovery и архивной копии.

Желательно сразу ориентироваться на уровень Enterprise и трехзвенную архитектуру.

Планируемая система была достаточно универсальной: имела возможность записи как на магнитные ленты, так и на дисковое хранилище, содержала в себе массу программ-агентов для копирования большинства известных систем, также имела дополнительные возможности в виде дедупликации, шифрования, сжатия данных и т. д. Все эти функции не обязательно покупать сразу, но сама возможность их реализации должна быть предусмотрена.

Ко всему прочему,купаемая система должна быть не очень дорогостоящей, так как бизнес только планирует интенсивный рост и не в состоянии выделить значительные средства.

**Как удовлетворить все эти требования?** Разумеется, это непросто. Решению подобных вопросов и посвящена оставшаяся часть книги.

## 4.5. Заключение

В этой главе читатель познакомился с классификацией систем резервного копирования по уровням. Разумеется, организовать некую систему сохранения данных можно только в привязке к нуждам бизнеса и спецификой предприятия. В то же время нельзя полностью идти на поводу «экономии на спичках». Своевременная модернизация способна предотвратить будущие проблемы, а бесконечное откладывание принятия решения о переходе на систему, адекватную растущему предприятию, не только может быть причиной ситуации, когда бизнес останется без резервных копий, но и потеряет крупные суммы денег из-за необходимости проводить миграцию уже разросшейся инфраструктуры.

# Вопросы к первой части для закрепления материала

1. Сформулируйте своими словами, почему важно создать условия для сохранения данных.
2. Что означают сокращения: RTO, PRO, MTPOD?
3. Что означают термины: «backup window», «off-site»?
4. О каких уровнях RAID вы узнали из данной книги?
5. Чем отличается RAID5 от RAID10?
6. Что такое RAID 2.0+?
7. В чем отличие технологии copy-on-write от redirect-on-write?
8. Какие существуют основные направления топологии резервного копирования?

Часть II

.....

# РЕЗЕРВНОЕ КОПИРОВАНИЕ КАК ОНО ЕСТЬ

# Глава 5

## Виды резервного копирования

*Объявление на дверях бухгалтерии: «Сисадмин уволился. Резервной копии нет. Зарплаты тоже сегодня не будет».*

### 5.1. Зачем все так усложнять, если нужно просто сохранить данные?

Вряд ли кто-то стал бы отрицать, что нужно всегда иметь надежную актуальную резервную копию. И никто не спорит, что гораздо надежнее иметь две копии! Например, за вчера и за сегодня. А еще лучше иметь целый архив за требуемый период. Все зависит от требований бизнеса. Но вот незадача: если инфраструктура достаточно большая, а требования серьезные, то сохранение всех данных превращается в весьма недешевое удовольствие. И дело не только в стоимости носителей для хранения копий, специфического программного обеспечения для спасения данных. Сам по себе процесс резервного копирования способен создавать весьма внушительную нагрузку на ИТ-инфраструктуру. Поэтому крайне важно максимально оптимизировать сам процесс выборки, передачи и хранения данных.

Это нужно для двух целей: без проблем попадать в «окно бэкапа» и не хранить колоссальных объемов не особо нужной информации, тратя на это деньги, человеко-часы и нервные клетки при обслуживании таких «суперхранилищ». Любая система резервного копирования – это уникальная экосистема, которая при неправильном подходе из ангела-хранителя превращается в кошмарное чудовище, пожирающее денежные и другие ресурсы, в итоге не приносящее ничего, кроме головной боли.

## 5.2. Планирование работы системы резервного копирования

Вот тут и начинается самое интригующее. Как оно обычно бывает: по-совещались, обозначили, какие данные нужно копировать, разошлись по кабинетам, подсчитали приблизительный объем и время на создание копии и тихо упали в обморок. Стало ясно, что при прямолинейном подходе «сохранять все нужное на неопределенный срок» не получится никогда. Прежде чем создавать систему сохранения данных, необходимо выяснить критерии ее эффективной работы и сопоставить их с возможными материальными и временными затратами.

Вот несколько характеристик, которые влияют на эффективность резервного копирования:

- глубина хранения данных;
- ограничение времени;
- ограничение объема;
- ограничение времени восстановления.

Остановимся по порядку на каждом пункте.

### 5.2.1. Глубина хранения данных

Это период, за который хранятся резервные копии. В предыдущих своих работах я писал о необходимости четкого разделения между резервным копированием, выполняемым с целью создания архива, и резервным копированием для последующего быстрого восстановления системы (Disaster Recovery). Но любой архив имеет замечательное свойство увеличиваться в размерах: чем дольше срок, за который собирались и поступали на хранение данные, тем больше размер архива. Если хранить абсолютно все копии, начиная со дня основания компании, то тут никакого бюджета не хватит – все уйдет на покупку средств хранения и на оплату услуг по обслуживанию такого «монстра». Поэтому разумное решение в данном случае – ограничить срок хранения некими рамками, что называется, глубиной хранения.

### 5.2.2. Ограничение времени создания резервной копии

Как уже неоднократно упоминалось выше, создание резервной копии – процесс достаточно ресурсоемкий. Нельзя взять и запустить копирование большого объема данных, не согласуя своих действий с другими участниками бизнес-процесса. Мало того, довольно часто

возникает ситуация, когда также нельзя запросто прервать процесс резервного копирования. Поэтому временные ограничения на создание резервной копии являются одним из решающих факторов при проектировании соответствующих систем.

### **5.2.3. Ограничение объема**

Пожалуй, этот параметр не нуждается в детальном описании. Рано или поздно любой ИТ-специалист сталкивается с дефицитом дискового пространства или нехваткой съемных носителей для хранения большого объема данных. Проблема часто заключается даже не в отсутствии денежных средств на приобретение нового оборудования, а в необходимости кардинальной переделки всей инфраструктуры или хотя бы одного из ее фрагментов. При этом имеющееся оборудование может быть уже не востребовано. Хороший пример: замена ленточных библиотек, при которой может потребоваться и модернизация всего аппаратного обеспечения, закупка новых кассет и т. д. Поэтому ситуация «надо записать, да некуда» встречается довольно часто и не приносит ничего хорошего.

### **5.2.4. Ограничение времени восстановления**

Этой важной характеристике, как правило, уделяется меньше всего внимания. Обычная практика при проектировании систем сохранения данных: «Главное, чтобы проверенная актуальная копия имелась в наличии (!), а сколько времени потребуется на его восстановление – это уже не так важно».

К сожалению, имеются грустные основания для подобного подхода – во многих российских компаниях до сих пор резервное копирование рассматривается как нечто не слишком обязательное, и ИТ-специалисты не склонны уделять вопросам сохранения данных слишком пристальное внимание. Там, где традиционно надеются на «русский авось», гораздо больше времени уделяют настройке какой-нибудь необязательной функции, например красивому отображению списка абонентов в адресной книге, нежели созданию основы для сохранения всей информационной структуры предприятия. Зато в аварийной ситуации за такое пренебрежительное отношение платить приходится сполна: и временем при простоях, и потерей данных, и потерей денег, репутационными рисками и т. д. Поэтому если вы не знаете, сколько времени вам потребуется на восстановление данных из резервной копии, можно с уверенностью сказать, что у вас на самом деле нет уверенности, что данные вообще будут восстановлены.

## 5.3. Оптимизация резервного копирования

После того как мы определились с характеристиками, позволяющими оценить эффективность системы резервного копирования, можно более предметно рассмотреть вопросы ее оптимизации.

Есть два основных направления для такой оптимизации: архивировать хранимые данные, постоянно совершенствуя алгоритмы сжатия, и... собирать и хранить далеко не всю информацию.

«Этого не может быть!» – воскликнет нетерпеливый читатель: «Как же так, ведь резервное копирование изначально ставит своей целью спасение всей информации!»

В реальности дело обстоит именно так. Мало того, второй способ с самого начала представляется гораздо предпочтительнее. Ведь сохраняемая информация различается не только по объему, но и по степени важности и уровню повторяемости. Нет смысла без конца клонировать одни и те же файлы, например ядра операционной системы или встроенной системы помощи, которые крайне редко изменяются и несут вспомогательную роль в работе информационных систем. Но можно пойти дальше – время от времени копировать важную информацию целиком, а в промежуточные временные участки только регистрировать изменения. Для того чтобы такое рискованное дело принесло положительный эффект и были сохранены именно нужные данные, были разработаны специальные виды неполного резервного копирования, о которых будет рассказано немного позже.

Виды резервного копирования:

- полное резервное копирование (Full backup);
- различные разновидности неполного резервного копирования:
  - инкрементальное резервное копирование (Incremental backup);
  - дифференциальное резервное копирование (Differential backup);
  - обратное инкрементальное резервное копирование (Reverse Incremental backup).

## 5.4. Полное резервное копирование (Full backup)

Является методом создания резервных копий, при котором выбранный массив данных копируется целиком. Своего рода главный и основополагающий вид резервного копирования. Если у вас нет ни

одной полной резервной копии, можно смело сказать, что резервное копирование как таковое отсутствует, потому что невозможно гарантированно восстановить ту или иную информацию.

Полные резервные копии полноценны и независимы, это самое простое решение для сбережения данных. Процесс восстановления также не причиняет каких-либо хлопот, потому что резервная, как уже говорилось выше, содержит любой фрагмент данных, который имелся и был скопирован в определенную дату. И конечно, полное резервное копирование незаменимо в случае, когда нужно подготовить резервную копию для быстрого восстановления системы с нуля. То есть если вам нужен Disaster Recovery – значит, без Full backup не обойтись.

Помимо собственно копии всех данных, это своего рода точка отсчета для применения других видов резервного копирования (об этом мы поговорим немного позже, когда рассмотрим другие виды резервного копирования и схемы ротации).

В чем же недостаток Full Backup? Почему не стоит каждый раз создавать полную резервную копию? Дело в том, что это не только наиболее полный и надежный вид резервного копирования, но и самый затратный. В случае необходимости сохранить несколько копий данных общий хранимый объем будет увеличиваться пропорционально их количеству. На сбор и передачу большого количества информации нужно длительное время и соответствующее техническое обеспечение: высокоскоростное сетевое оборудование, производительная дисковая подсистема, – и, самое главное, собранные копии надо где-то хранить. В итоге объем хранимых в резервных копиях данных будет многократно превышать объем данных, необходимый для работы. Рано или поздно такая схема окажется неподъемной для бюджета, и от нее придется отказаться.

Чаше всего нет возможности выделить необходимый промежуток времени, когда можно пожертвовать производительностью тех или иных фрагментов инфраструктуры во имя полного резервного копирования всех данных.

Чтобы избежать такого расточительства, используют алгоритмы сжатия и применяют неполное резервное копирование, о котором пойдет речь ниже.

### ***Краткое резюме по Full backup***

Преимущества:

- содержит все данные, которые занесены в настройки задания для копирования;
- для восстановления нужна только одна полная копия;

- быстрое восстановление.

Недостатки:

- создание такой копии может занять много времени;
- занимает много места при хранении.

## 5.5. Неполное резервное копирование и его виды

Для экономии материальных средств при организации резервного копирования, а также по возможности не выходить за рамки «окна бэкапа» разработано несколько технологий, которые применяют в зависимости от конкретной ситуации.

Очевидно, что применение неполного резервного копирования не может быть полноценной и независимой реализацией процесса сохранения данных, а только дополнением к полному резервному копированию. То есть какую бы гениальную схему для неполного резервного копирования мы не придумали, она в любом случае будет опираться на тот факт или предположение, что ранее была создана полная резервная копия, которая хранится в надежном месте.

Для реализации неполного резервного копирования используется метод создания некоего признака, или маркера, которым помечаются данные, прошедшие (или, наоборот, не прошедшие) процедуру полного резервного копирования после их изменения. Например, в операционных системах семейства Windows для этих целей используется соответствующий атрибут файла (архивный бит), который устанавливается, когда файл был изменен, и сбрасывается программой резервного копирования при выполнении Full backup. В других операционных системах, например в некоторых Unix-подобных платформах, для этих целей используются другие признаки, например время изменения файла. Время изменения сравнивается с последней датой создания полной копии, и на основании результата принимается решение относительно необходимости копирования файла.

На таком же принципе основано неполное резервное копирование баз данных, корпоративной почты и других систем хранения информации.

### 5.5.1. Инкрементальное копирование (Incremental backup)

Как я уже писал выше, при неполном резервном копировании система ориентируется на некий признак изменения файла, например на

архивный бит при работе на платформах Windows. При инкрементальном копировании этот признак сбрасывается после копирования. То есть скопировали и сами себе сделали пометку, что объект был скопирован. Если до следующего запуска процесса резервного копирования объект успеет измениться, ПО резервного копирования его сохранит. Если изменения не производились, признак остался в прежнем состоянии «без изменений», объект в новую резервную копию не попадет. Так будет, либо пока он снова не изменится, либо он все-таки будет скопирован при выполнении Full backup (см. рис. 5.1).

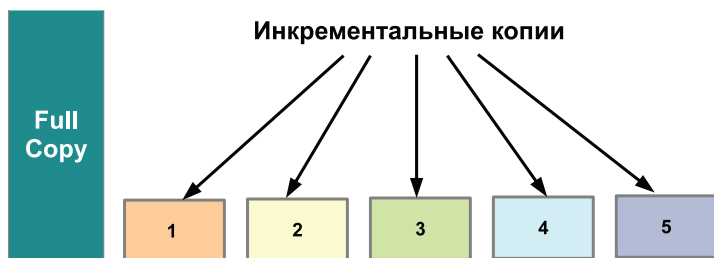


Рис. 5.1. Инкрементальный вид резервного копирования



Там, где нет явного признака изменения файла, используется алгоритм сравнения времени модификации файла или любое другое условие. Но суть остается такой же: объект менялся – нужно копировать, не менялся – ничего не нужно делать.

В этом случае копируются не все данные (файлы, секторы и т. д.), а только те, что были изменены с момента последнего копирования. Еще раз повторю, схема с применением любого вида резервного копирования будет неполноценной, если время от времени не делать Full backup. При полном восстановлении системы нужно провести восстановление из последней полной копии, а уже потом поочередно извлекать данные из каждой последующей инкрементальной копии в порядке их создания. Сначала извлекаем данные из самой первой и старой копии, потом из более свежей – и так до конца периода. Изменять порядок чередования при полном восстановлении данных строжайше запрещено.

Другое дело, когда нужен конкретный объект данных на момент определенной даты создания копии. Некоторые системы резервного копирования позволяют извлечь отдельный фрагмент данных, не прибегая к восстановлению всей цепочки данных. Например, используется система резервного копирования с пофайловым доступом

к данным, и нужен текстовый документ, который редактировался в прошлый месяц. Тогда достаточно взять одну инкрементальную копию за указанный период и найти в ней искомый файл.

Для чего используется этот вид копирования? В случае создания архивных копий он необходим, чтобы сократить расходуемые объемы на устройствах хранения информации (например, сократить число используемых ленточных носителей). Также это позволит минимизировать время выполнения заданий резервного копирования, что может быть крайне важно в условиях, когда приходится работать в плотном графике 24×7 или прокачивать большие объемы информации.



Иногда инкрементальное резервное копирование называют «инкрементным». В литературе встречаются оба термина. В нашем случае мы будем ориентироваться на английское прочтение данного термина – «*incremental*».

У инкрементального копирования есть один нюанс, который нужно знать, – это так называемый «инкрементальный парадокс». О нем пойдет речь в следующем разделе.

### ***Краткое резюме по Incremental backup***

Преимущества:

- минимальные затраты времени на создание резервной копии;
- размеры копий минимальные.

Недостатки:

- восстановление из такого бекапа занимает больше времени (задействованы все промежуточные фрагменты);
- для восстановления нужны предшествующая полная и все неполные резервные копии за требуемый период;
- имеет место явление под названием «инкрементальный парадокс», о котором речь пойдет ниже.

### **5.5.2. Инкрементальный парадокс**

Инкрементальное резервное копирование при полном восстановлении может сыграть своего рода забавную шутку. Дело в том, что в наборе из полной и соответствующих промежуточных копий хранятся все объекты, которые существовали за данный период. Не важно, были ли они впоследствии удалены, переименованы или перенесены в другое.

Рассмотрим следующий пример (см. рис. 5.2).

Допустим, у нас есть некий файл-сервер с общим ресурсом, с которым работают пользователи.

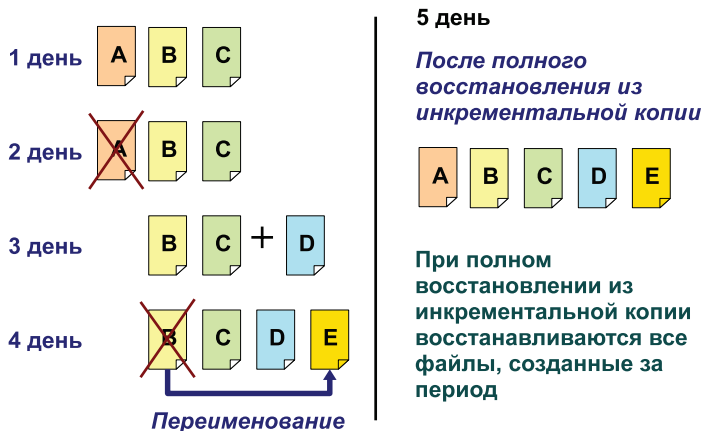


Рис. 5.2. «Инкрементальный парадокс»

В 1-й день рабочего периода пользователи создали три файла: «А», «В», «С».

Во 2-й день удалили файл «А».

В 3-й день создали еще один файл – «D».

В 4-й день файл «В» переименовали в «Е».

На 5-й день дисковый массив на сервере вышел из строя, и после поэтапного восстановления из инкрементальных копий мы получили... все пять файлов: А», «В», «С», «D», «Е».

На самом деле это явление не таит в себе никаких чудес. Ведь мы вначале добросовестно сохраняли все файлы пользователей, а потом так же добросовестно все их восстановили при извлечении данных из полной и всех промежуточных резервных копий.

Принимая во внимание данное явление, необходимо учитывать две вещи:

1. Общий объем данных после восстановления может оказаться больше, чем до восстановления. И поэтому при использовании инкрементального резервного копирования необходимо при восстановлении резервировать до одной трети дискового пространства.
2. Данные после восстановления могут не всегда соответствовать ожиданиям пользователей. Например, пользователь считает, что раз и навсегда удалил файл «А», а он снова «всплывает» после восстановления. Поэтому если позволяют технические и временные ресурсы, крайне желательно сначала восстановить

данные на дополнительный носитель, а потом предоставить пользователям возможность самим перенести необходимые данные в нужное место. И, разумеется, быть готовым ответить на «неудобные вопросы» по поводу ранее удаленных или переименованных объектов данных.

### 5.5.3. Дифференциальное резервное копирование (Differential backup)

Отличается тем, что тот самый признак обновления файла, который в случае с Incremental backup сбрасывался при каждой копии, теперь остается неизменным вплоть до создания полной резервной копии. То есть один и тот же объект будет копироваться вплоть до следующего создания полной резервной копии. Поэтому Differential backup вмещает все файлы, измененные со времени последнего Full backup. Данные при этом помещаются в архив «нарастающим итогом» (см. рис. 5.3).

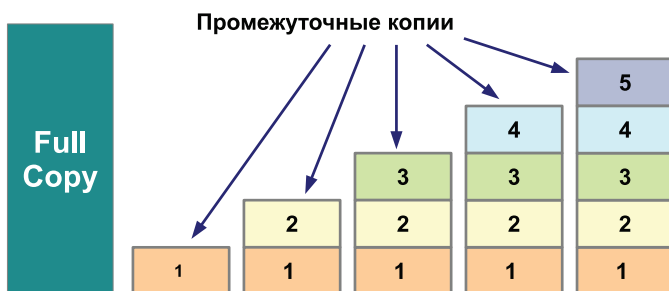


Рис. 5.3. Дифференциальный вид резервного копирования

В силу того, что каждая новая копия, созданная таким образом, содержит данные из предыдущей, это удобно при полном восстановлении данных на момент аварии. Для реанимации системы нужны всего лишь две копии: полная и последняя неполная (дифференциальная), поэтому вернуть к жизни поврежденную ИТ-инфраструктуру или отдельный ее участок можно гораздо быстрее, чем последовательно извлекать данные из всех инкрементальных фрагментов. Данный вид резервного копирования хорошо подходит для Disaster Recovery.

К тому же Differential backup позволяет сэкономить время при розыске объекта в веренице резервных копий, и ему чуждо такое явление, как «инкрементальный парадокс».

Но дифференциальное копирование значительно проигрывает инкрементальному в экономии требуемого пространства. Так как в каждой новой копии хранятся данные из предыдущих, суммарный объем зарезервированных данных может быть сопоставим с полным копированием. И, конечно, при планировании расписания (и расчетах, поместится ли процесс бэкапа во временное «окно») нужно учитывать время на создание последней, самой объемной промежуточной копии.

### ***Краткое резюме по Differential backup***

Преимущества:

- требует меньше времени для полного восстановления в сравнении с Incremental+Full backup;
- не возникает «инкрементальный парадокс».

Недостатки:

- за счет растущего объема данных создание копии в среднем происходит медленнее, чем у Incremental backup;
- занимает больше места на диске, чем при инкрементальном копировании, поэтому не так удобно для систем архивного хранения за длительный период;
- если выполнять данный вид резервного копирования слишком длительное время без создания полной резервной копии, размеры при хранении могут значительно вырасти.

## **5.5.4. Обратное инкрементальное резервное копирование (Reversed incremental backup)**

Эта разновидность резервного копирования отличается от своего описанного ранее собрата тем, что при создании инкрементальных копий изменения вносятся еще и в полную резервную копию. Если при обычном способе полная резервная копия является неизменяемым эталоном, относительно которого регистрируются изменения в неполных копиях, и одновременно гарантом восстановления данных хотя бы на начало периода, то в случае с обратным резервным копированием полная резервная копия обновляется с учетом последних изменений (см. рис. 5.4).

Основная цель Reversed incremental backup – минимизировать время полного восстановления. Теперь нет необходимости после восстановления из полной копии дополнительно еще что-то делать с целой вереницей промежуточных фрагментов. Это очень удобно при полном восстановлении после аварии. Идеальный способ сохранения

данных для Disaster Recovery. В то же время он неплохо подходит для работ по подготовке к различным миграциям данных, перевозке серверов и т. д. Достаточно в самый последний момент сделать очередную инкрементальную копию, и впоследствии можно в кратчайшие сроки восстановить всю систему.

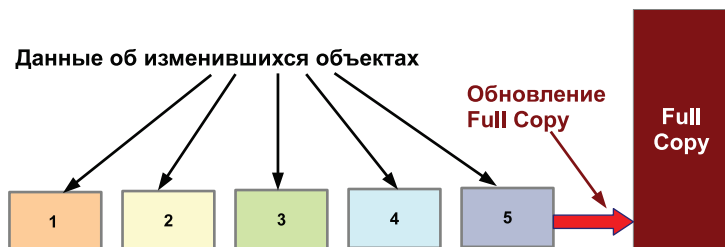


Рис. 5.4. Обратный инкрементальный вид резервного копирования (Reversed incremental backup)

Однако за все приходится платить. В первую очередь приходится попрощаться с уверенностью в том, что все не так страшно, когда процедура создания Incremental backup дала сбой, ведь есть нетронутая полная копия и предыдущие фрагменты. Увы, бывают случаи, когда модификация исходной резервной копии при Reversed incremental backup может разрушить все надежды на восстановление. Например, из-за аппаратной ошибки или некорректной работы программного обеспечения. Не забываем также про различные вирусы, действия третьих лиц. В случае с обычным инкрементальным копированием такие риски маловероятны, особенно когда ленточный картридж с полной копией преспокойно пылится в запортом шкафу на охраняемой территории.

Чтобы этого избежать, нужно или почаще выполнять Full backup, или использовать добавочные полные резервные копии (Additional Full Backup), о которых пойдет речь ниже.

Еще одна интересная деталь. Если обычный incremental backup может быть с легкостью использован как на дисковом накопителе, так и на ленточном, то Reversed incremental backup рекомендуется использовать именно на дисковом хранилище. Связано это как со специфическим предназначением в первую очередь для Disaster Recovery, так и с логикой работы, ведь после записи изменений нужно разыскать и модернизировать полную копию. Гораздо удобнее это делать, когда все вместе хранится на одном разделе дискового хранилища.

### *Краткое резюме по Reversed Incremental backup*

Преимущества:

- быстрое восстановление, удобство при миграции;
- небольшой объем позволяет быстро выполнять создание промежуточных копий.

Недостатки:

- риск повредить полную копию при модернизации;
- трудность при использовании ленточных хранилищ.

## 5.6. Добавочная, или дополнительная, резервная копия

Здесь все просто. Добавочная копия практически ничем не отличается от обычного Full backup, за исключением одного пункта: не сбрасывается признак изменения файла. Это значит, что добавочная копия не разрывает логическую цепочку Incremental или Differential backup и позволяет этому процессу выполняться дальше в том же порядке (см. рис. 5.5).

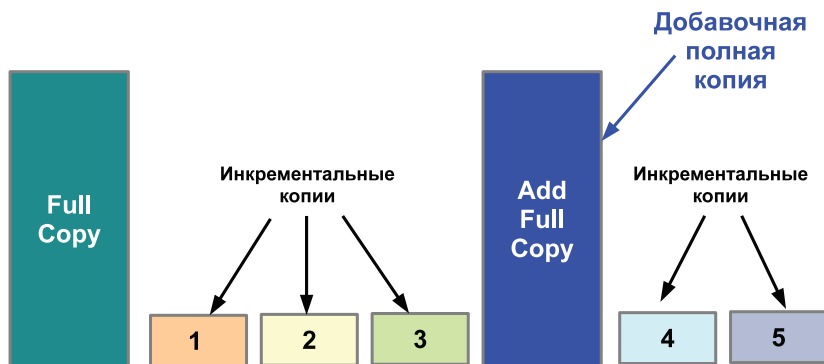


Рис. 5.5. Добавочное полное резервное копирование

Где можно использовать такой вид резервного копирования? Там, где нет возможности или желания возиться с восстановлением из промежуточных копий. В первую очередь он применяется для создания носителей с данными для перемещения за территорию (off-site) на случай форс-мажорных обстоятельств. Также служба безопасности может запросить внеочередную полную копию данных для ана-

лиза информации и просто «потому, что могу получить». Полные копии данных используются при аудите и формировании отчетов при управленческом учете.

Преимущества и недостатки у него такие же, как и у полного резервного копирования, за исключением одной немаловажной детали – добавочная копия делается либо в «окно», предназначенное для создания одной из промежуточных, либо вообще в любое время по срочной заявке. А времени в этом случае будет гораздо меньше, и нужно принять соответствующие меры, чтобы не помешать бизнес-процессам. Если такая копия к тому же записывается на то же хранилище, что и все остальные, нужно внимательно отнестись к объему выделяемого пространства, чтобы его хватило для записи фактически еще одной полной копии.

## 5.7. Заключение

В данной главе описывались различные виды резервного копирования и возможности, которые предоставляются проектировщикам и администраторам систем резервного копирования. Как видите, простыми организационными методами можно значительно уменьшить размер хранимых резервных копий и время их создания. Далее мы еще не раз вернемся к видам резервного копирования, когда будем говорить о схемах ротации, методах копирования и ошибках администраторов.

# Глава 6

## Схемы ротации носителей

*Прошлое – это read-only memory, а будущее – write-only memory.*

(Найдено на Bash.im)

### 6.1. Схемы ротации

Эта глава является продолжением предыдущей, и речь идет все о том же: как эффективно, с минимальными временными и финансовыми затратами организовать работу резервного копирования.

Использование различных видов резервного копирования предполагает их правильное чередование и взаимодействие. В итоге приходится прибегать к выработке определенного порядка. Об этом и пойдет речь в этой главе.



В тексте часто используется термин «носитель». Читатели, знакомые со специализированным оборудованием для резервного копирования, могут вполне справедливо связать это понятие с ленточными кассетами (картриджами), например стандарта LTO. Первоначально при создании схем ротации и методик расчета используемых накопителей подразумевались именно такие системы. Но с удешевлением систем хранения на базе жестких дисков, и в особенности их недорогого варианта NAS, термин «носитель» стал применяться и к этим устройствам тоже. Сейчас он просто обозначает некий конечный ресурс, куда сохраняются данные и который, в принципе, может быть сменен (например, администратор выберет для размещения новой копии другую сетевую папку).

### 6.2. А для чего вообще необходимы схемы ротации?

Любая система резервного копирования имеет два главных ограничителя: время и деньги. Анализ любой характеристики (в том числе

перечисленных в предыдущей главе) так или иначе приводит к этим двум факторам. Носители данных не бесконечны. Не важно, какого объема ленточный накопитель вы приобрели или какой гигантский дисковый массив организовали – его ресурсы рано или поздно закончатся. Мало того, любой носитель подвержен выходу из строя. В то же время мы не можем позволить себе каждый раз записывать свежую копию на новую ленту или диск, а потом хранить это бесчисленное множество лет. Поэтому хотим мы этого или нет, но рано или поздно придется перезаписывать старый носитель новой информацией, а также иметь представление, когда он в очередной раз был очищен, в течение какого периода на нем размещались данные и через какое время эти операции повторятся вновь. И лучше, если этот процесс будет происходить через заведомо определенные промежутки времени и организован будет по хорошо понятной системе. Вот так незаметно мы и подошли к необходимости использовать схему ротации носителей.

Существует специальный термин «глубина хранения», или «глубина бэкапа», обозначающий предельное время хранения резервной копии.

## 6.3. Когда схема ротации не используется

Как ни странно, метод бессистемной замены носителей до сих пор активно применяется. В этом случае администратор резервного копирования самолично решает, какой носитель будет перезаписан, а какой – отложен на более длительное хранение. И не всегда причина кроется в нежелании навести порядок с применением научного подхода. В некоторых ситуациях для этого могут быть вполне объективные причины. Например, когда время от времени данные на резервной копии нужно сохранять более длительный срок, скажем, для последующего анализа. Тогда разумнее выбрать для перезаписи другой носитель, что, естественно, не подпадает под какую бы то ни было схему ротации. Или имеет место скачкообразное увеличение объема данных, предназначенных для копирования, после чего следует их уменьшение. В таком случае администратор резервного копирования вообще не может предугадать, какой объем ему потребуются для размещения копий, а создавать систему хранения с запасом «на всякий случай» нет финансовой возможности.

Но все же такой «управляемый хаос» является скорее вынужденным, в первую очередь из-за необходимости постоянно «держат руку на пульсе». И любые неожиданные события: болезнь или увольнение ответственного сотрудника, какие-то личные обстоятельства – могут оставить ИТ-инфраструктуру без актуальной резервной копии. А это уже прямая угроза существованию любой организации. Поэтому крайне важно максимально автоматизировать процесс резервного копирования. Что же касается необходимости решения неординарных задач, то для этой цели неплохо подходит создание добавочной полной копии.

## 6.4. Полное резервное копирование

Наиболее простой способ организации резервного копирования – всегда создавать полную резервную копию. Если есть средства и время для создания Full Backup – это самый хороший вариант. Например, закупить 60 носителей и сохранять данные в течение двух месяцев. В этом случае, помимо скорости восстановления данных, существует возможность поднять данные на любой день за указанный период. При этом любая копия может быть использована для off-site, то есть попросту храниться в другом месте, подальше от основной ИТ-инфраструктуры на случай форс-мажора.

Главный недостаток данного метода состоит в ощутимых финансовых и временных затратах. Рано или поздно наступит тот час, когда объем хранимой информации увеличится настолько, что задания резервного копирования не будут успевать выполняться в отведенное время, а результаты их выполнения – помещаться на предоставленные носители. В качестве наиболее популярного выхода из подобной щекотливой ситуации служит переход на схему ротации с использованием сочетания полных и неполных резервных копий.

## 6.5. Самая популярная схема ротации – «ОТЕЦ–СЫН»

В принципе, сама схема рассматривалась еще в предыдущей главе (см. рис. 6.1).

Основной принцип: в самом начале периода выполняется полное резервное копирование, а каждый последующий этап создается неполная копия: инкрементальная или дифференцированная. Нетруд-

но догадаться, что под понятием «отец» подразумевается Full backup, а «сыном», точнее «сыновьями», являются промежуточные копии.

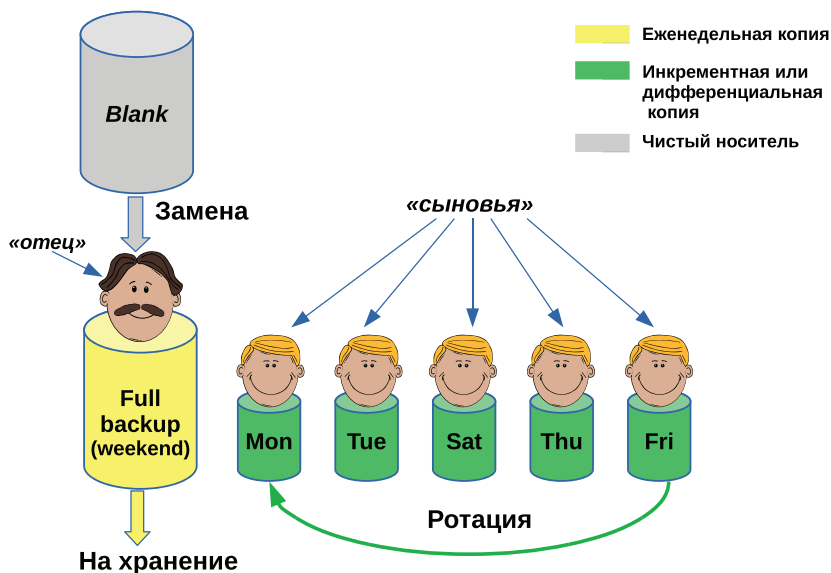


Рис. 6.1. Схема ротации «отец–сын».  
В этой схеме полная резервная копия всегда поступает на хранение, неполные – перезаписываются

Особенно эта схема популярна в организациях, работающих по обычному недельному графику с выходными в конце недели, при котором не требуется режим работы 24×7. В этом случае полная копия создается по выходным, а в течение недели создается ежедневная промежуточная копия. В результате к концу периода администратор резервного копирования получает 6–7 копий (в зависимости от длины рабочей недели), из которых он может поэтапно восстановить данные за период.

### **Методика расчета количества накопителей для схемы «отец–сын»**

Предположим, что полная и промежуточная копии каждый раз записываются на отдельный носитель. В случае с пятидневной рабочей неделей для одного цикла нам необходимо иметь один накопитель для Full backup и еще пять – для промежуточных копий. Всего шесть копий.

В совсем небольших организациях зачастую так и поступают: сохраняют на протяжении одной недели. Нетрудно заметить, что после выполнения полного резервного копирования предыдущие промежуточные копии бесполезны. Поэтому если очередной процесс полного резервного копирования не выполнится, то бизнес останется без какой бы то ни было резервной копии вообще. В случае аварийной ситуации, как говорится, «все пропало». В идеале имеет смысл сохранять данные в течение как минимум двух таких циклов, то есть за две недели, чтобы использовать ранее записанные носители. Но, в принципе, можно оставлять нетронутой только полную копию от предыдущего периода. В этом случае даже при провале Full Backup все равно остается шанс восстановить данные из комплекта: полная копия за предыдущий период и «оставшиеся в живых» промежуточные копии. Если они «умирают» за счет затирания новыми инкрементальными копиями, наверное, имеет смысл отметить, что повторное использование носителей происходит от конца к началу – в сегодняшний понедельник за прошлую пятницу.

Проще говоря: есть некая цепочка, состоящая из полной копии и промежуточных. В момент создания новой полной копии затерли носитель, но бэкап так и не сделался. Получается, что промежуточные копии также бесполезны. И если авария, например сбой серверов с базами данных, произойдет в понедельник, то восстанавливать их неоткуда. Но если сделать Full backup на чистый носитель, то даже при провале полной копии все равно можно поднять данные хотя бы на конец прошлой недели, воспользовавшись нетронутой «цепочкой».

Таким образом, минимальное количество носителей рассчитывается по методике:

Количество накопителей для всех инкрементных копий +  
количество накопителей для текущей полной копии +  
количество носителей для прошлой полной копии.

Таким образом, количество носителей для Full Backup просто удваивается.

Допустим, и для полной, и для промежуточной копий расходуется по одному носителю.

В нашем случае получается 7 носителей:

5 промежуточных + 1 полная новая + 1 полная за прошлый период = 7.

Если же администратор спохватился слишком поздно и, кроме носителя с Full copy, были перезаписаны и последующие инкрементальные или дифференцированные копии, то восстановить данные можно будет в лучшем случае только на момент создания прошлой полной копии. Конечно, это очень плохо, но даже резервная копия недельной давности все же больше, чем вообще ничего.

Поэтому куда безопаснее иметь два полных комплекта «отец–сын»: за текущий и прошлый периоды.

Но тогда понадобится уже 12 носителей:

$(5 \text{ промежуточных} + 1 \text{ полная}) * 2 \text{ периода} = 12$ .

В этом случае расходуется больше носителей, зато гарантированно сможем восстановить данные на любой день за прошедшие две недели.

При использовании обратного инкрементального копирования – Reversed Incremental backup – возникает другая ситуация. Напомню, что особенностью этого вида резервного копирования является внесение изменений в полную копию при создании инкремента. С одной стороны, это выглядит неплохо, так как полная копия всегда отражает текущее состояние. И может быть использована для полного восстановления на момент, предшествующий потере данных. С другой – она же может быть повреждена при сбое процесса копирования. Поэтому данный вид резервного копирования плохо подходит для организации архивного хранения за длительный период. Основная его задача – создавать основу для быстрого восстановления инфраструктуры после аварии (Disaster Recovery). Если же необходимо всенепременно использовать Reversed Incremental backup для создания архивных копий, нужно перед запуском каждого процесса продублировать существующую полную копию на другом носителе. В этом случае даже при самом серьезном сбое все равно остается хотя бы один работающий экземпляр Full copy, который можно использовать для восстановления системы и данных.



Следует отметить, что приведенная методика расчета не является прямой инструкцией по закупке нужного количества носителей. Здесь не учитывается необходимость приобретения запасных носителей на случай выхода из строя, а также создания добавочных полных копий, например с целью передачи данных для анализа службе безопасности. Конечное число имеющихся накопителей должно быть всегда больше расчетного.

Почему я назвал схему «отец–сын» самой популярной? Все очень просто – она применяется в небольших организациях, которых, как известно, куда больше, чем крупных корпораций.

## 6.6. «Дед–отец–сын» как классическая схема ротации

Вышеописанная схема ротации «отец–сын» работает довольно неплохо, но все же не лишена недостатков. Например, она не рассчитана на длительное хранение данных. Разумеется, остается возможность просто умножить количество периодов, чтобы «закрыть» требуемый промежуток времени. Но слишком уж затратной выглядит подобная схема. Кроме того, схема «отец–сын» не рассчитана на изъятие отдельных носителей с целью передачи их на хранение.

Поэтому для организации процесса сохранения данных крупной компании используются другая схема: «дед–отец–сын» – и ее разновидности.

Принцип работы здесь во многом напоминает «отец–сын», повторяемый в течение нескольких периодов. За одним небольшим исключением – в схему внесен избыточный элемент в виде дополнительной полной копии, которая по истечении периода перемещается за периметр ИТ-инфраструктуры (см. рис. 6.2).

На рисунке изображен наиболее популярный вариант использования данной схемы ротации – за месячный период. В течение каждой недели выполняется резервное копирование по схеме «отец–сын». Каждую новую неделю носитель для хранения промежуточных копий перезаписывается, а полные копии сохраняются на протяжении месяца. В конце месяца создается дополнительная полная копия – «дед», предназначенная для длительного хранения. С началом нового периода носители для хранения еженедельных полных копий «отцы» перезаписываются. Таким образом в течение недели данные могут быть восстановлены за каждый день, в течение оставшегося месяца – с интервалом в одну неделю, а в течение более длительного срока – помесечно, из дополнительных полных копий, расположенных, как правило, в специальном хранилище.

Ниже приведен расчет минимального количества носителей для годового цикла:

Количество носителей для полной копии  $\times 12$  + количество носителей для полной копии  $\times 4$  + количество носителей для промежуточных копий.

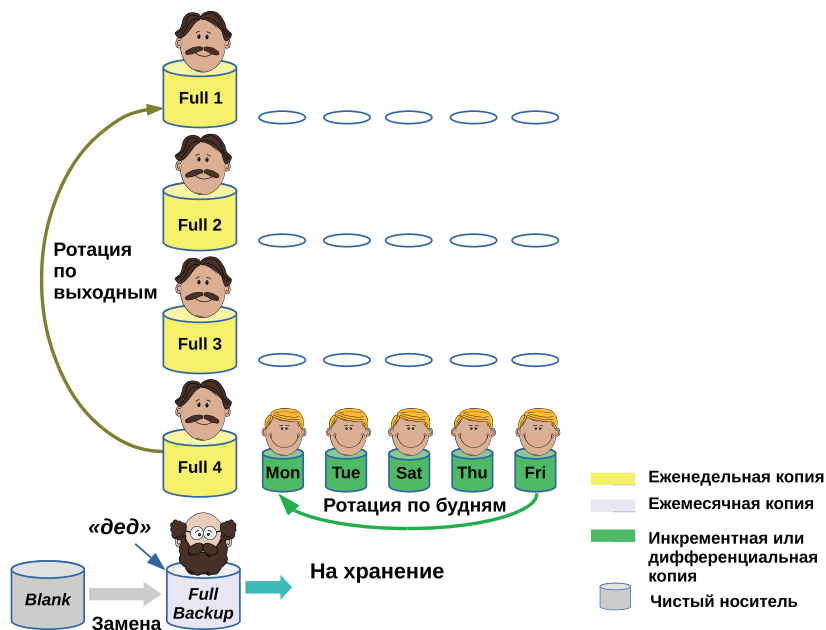


Рис. 6.2. Схема ротации «дед–отец–сын».  
В этой схеме одна полная резервная копия  
поступает на хранение, остальные – перезаписываются.  
Неполные копии всегда перезаписываются

В нашем примере с пятидневной рабочей неделей это составляет:

$$1 \times 12 + 1 \times 4 + 1 \times 5 = 21 \text{ носитель.}$$

(Напомним, что для приводимых примеров используется допущение, что для создания всех типов копий каждый раз нужно по одному носителю. При других условиях количество используемых единиц хранения будет больше. Также следует помнить, что в месяце не 28 дней или 4 недели, поэтому иногда заказывают на несколько носителей больше, на случай, если потребуются изменить график.) Соответственно, их самих надо чаще заменять. Однако если принять во внимание, что схема «дед–отец–сын» унаследовала те же недостатки, что и «отец–сын», то имеет смысл сохранять промежуточные копии не только за последнюю, но и за предпоследнюю неделю. Улучшенная формула будет выглядеть так:

Количество носителей для ежемесячной полной копии  $\times 12$  + количество носителей для еженедельной полной копии  $\times 4$  + количество носителей для промежуточных копий  $\times 2$ .

Для нашей «пятидневки» это еще пять дополнительных носителей:

$$1 \times 12 + 1 \times 4 + 2 \times 5 = 26.$$

Как видим, в случае со схемой ротации «дед–отец–сын» количество носителей увеличилось незначительно, зато это прибавило надежности нашей схеме.

В случае, когда необходимо хранить данные в течение двух и более лет, в хранилище оставляют данные за последний месяц года, а все остальные перезаписывают. Таким образом на каждый год. То есть если предполагается хранить данные 5 лет, то дополнительно понадобится еще 4 носителя для обеспечения ежегодных копий за предыдущие периоды.

Как уже было сказано выше, в наших расчетах используется некая идеальная схема. На самом деле обязательно нужно делать поправку еще и на износ носителей. Постоянное промежуточное копирование изнашивает ленточный картридж или жесткие диски в хранилище гораздо сильнее, нежели ежемесячное создание полной копии. Поэтому при проектировании и внедрении систем резервного копирования все это нужно учитывать и закупать носители «с запасом». (Я бы сказал, с очень хорошим запасом, чтобы не остаться посреди бюджетного года без возможности сделать резервную копию.)

При всех достоинствах описанной схемы ротации у нее есть один существенный недостаток: в течение последних чисел месяца нужно выполнить дополнительную полную копию, а временной интервал, когда это можно выполнить, не мешая основной работе бизнеса, так называемое «окно бэкапа», может оказаться недостаточным.

Приходится либо захватывать рабочие часы, либо пожертвовать частью сохраняемых данных. Чтобы этого избежать, иногда идут на хитрость и вместо добавочной полной месячной копии отдают на хранение одну еженедельную. Проще говоря, вместо «деда» отдают какого-нибудь «отца» и просто добавляют чистый носитель в схему ротации. Такой вариант вполне допустим, хотя бывает довольно неудобно обращаться за изъятием нужной «отцовской» копии из хранилища при необходимости восстановить данные за ближайший период. Но тем не менее такая схема вполне работоспособна.

## 6.7. Операция «слияние», или Как пожертвовать историей на благо экономики

Описываемые выше схемы ротации предназначены для хранения архивных копий за определенный период, чаще всего – довольно длительный. Но иногда возникает ситуация, когда не требуется или просто нет возможности хранить большое количество копий, например для Disaster Recovery. Достаточно поддерживать в актуальном состоянии несколько рабочих копий. Или используемая система является резервной. Такой подход применяется при организации филиальной инфраструктуры, когда «на местах» необходимо хранить данные для быстрого восстановления ИТ-инфраструктуры, а весь объем информации за прошлые периоды размещается в центральном офисе (штаб-квартире).

Для подобных ситуаций неплохо подходит схема ротации, называемая Backup Merge, что по-русски означает «слияние» (см. рис. 6.3).

Как видно по рисунку, данная схема ротации очень напоминает рассмотренную выше «отец–сын», но с небольшим отличием, которое затрагивает алгоритм создания неполных копий. Ниже описан примерный порядок работы системы резервного копирования на основе Backup Merge.

- Создается полная резервная копия.
- Создаются промежуточные резервные копии, пока их число не достигнет установленного предельного значения.

После этого работа происходит по следующему алгоритму:

- создается новая промежуточная резервная копия;
- данные в Full Copy синхронизируются с самой старой промежуточной копией. Или еще можно сказать, что самая старая промежуточная копия сливается с полной;
- удаляется самая старая промежуточная копия.

Этапы 3–5 повторяются сколько угодно долго, пока не понадобится реорганизовать работу системы резервного копирования.

Данная схема работы во многом схожа с Reversed Incremental backup, за исключением одной детали: при обратном резервном копировании в полную копию вносятся самые последние изменения, а при Merge Backup процесс обновления Full Copy выполняется за счет самой старой неполной копии. Соответственно, главный недостаток один и тот же: риск повредить Full Copy при модификации. И схемы

решения данной проблемы также совпадают: сохранять полную копию на другом носителе перед ее обновлением.

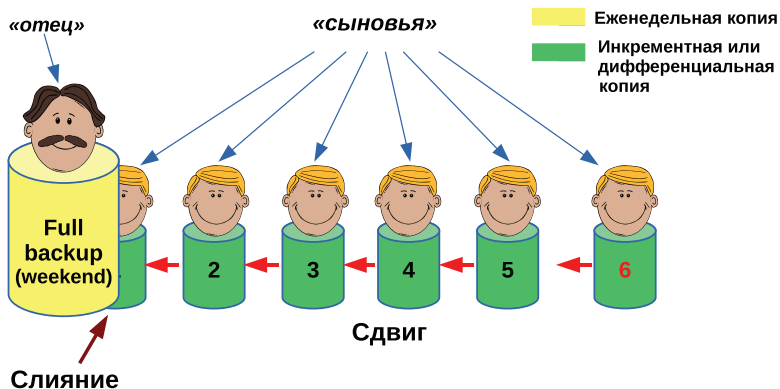


Рис. 6.3. Схема ротации Backup Merge.

В этой схеме полная резервная не стирается, но объединяется с самой ранней неполной копией.

Таким образом, количество копий всегда остается неизменным

Расчет количества используемых носителей для данной схемы ротации производится таким образом:

Количество накопителей для хранимых инкрементных копий + накопители для новой копии + количество накопителей для текущей полной копии + количество носителей для прошлой полной копии.

Для пятидневной рабочей недели с одним носителем для любого вида копии это будет выглядеть как

$1 \times 5 + 1$  (для свежей) + 1 (для полной) + 1 (для резервной полной) = 8 носителей.

Если сравнить расчет количества носителей с классической схемой ротации «отец–сын», можно заметить, что добавился один комплект носителей для свежей неполной копии. Это связано с тем, что в грамотно организованных системах сначала записывается новая копия, а потом происходит слияние. Если процесс сбора и записи промежуточных данных не увенчался успехом, его можно повторить, не теряя при этом уже существующей «цепочки».

Необходимость всегда дублировать полную резервную копию перед запуском нового процесса из кажущегося недостатка легко превращается в достоинство, когда заходит речь о необходимости off-site. В отличие от громоздкой схемы «дед–отец–сын», когда для выноса резервных данных за периметр нужно дожидаться записи добавочной полной копии, в данном случае такую операцию можно проводить на любом этапе. На рис. 6.4 показан вариант с использованием облачной системы хранения данных, когда информация из полной копии синхронизируется с удаленной системой.

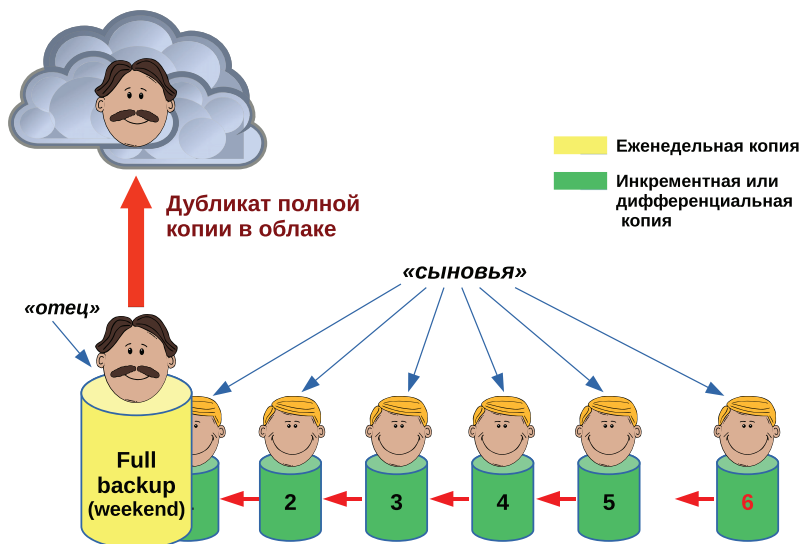


Рис. 6.4. Перенос данных в облако при использовании схемы ротации Backup Merge – off-site с применением облачного копирования

С использованием технологии дедупликации данных процесс не занимает много времени, а учитывая тот факт, что при этом не затрагивается основная ИТ-инфраструктура, то может проводиться и вне «окна бэкапа», то есть в любое удобное время (если это позволяет канал передачи данных, разумеется).

## 6.8. Постоянное резервное копирование, или Continuous backup

До сих пор мы рассматривали системы ротации, которые основаны на единовременном запуске заданий резервного копирования, например в ночное время, когда бизнес-процессы имеют минимальную активность. Но такой ненавязчивый подход имеет и отрицательные стороны, в первую очередь из-за возможности потерять изменения за прошедший день. Представьте себе ситуацию, когда сбой системы произошел в конце рабочего дня. В этом случае системный администратор будет располагать резервной копией, созданной в прошедшую ночь, а вся работа целой организации за нынешний день пойдет прахом.

Для предотвращения подобных ситуаций существует метод, называемый Continuous backup. Ему соответствует довольно приблизительный термин «точки мгновенного восстановления».

При использовании данной технологии резервные копии создаются не по определенному расписанию, а непрерывно, по итогам модификации данных в файлах. Когда данные записываются на диск, допустим, при закрытии файла, они одновременно сохраняются на устройстве резервного копирования.

Если быть предельно точным, то эта система больше похожа на синхронизацию данных, например с применением DFS от Microsoft. Отличие заключается в том, что обновление данных носит непрерывный и локальный характер.

Существует множество различных решений для такого рода синхронизации. Например, для Windows-систем создана программа NTI Shadow for ReadyNAS, которую можно получить бесплатно с хранилищами NETGEAR. Для Unix-систем хорошую службу может сослужить Vox Backup. Различные программы для continuous backup могут использовать файловый или блочный метод резервного копирования, а также системы дедупликации. При этом возможна хорошая экономия дискового пространства на устройстве хранения копий.

Данная система оказывается просто незаменимой для ситуаций, когда пользователи вынуждены держать большие объемы данных на локальных дисках. Например, при работе с большими графическими файлами или системами обработки аудио- и видеоконтента. Хранить данные на центральном сервере и постоянно «тянуть» их от сервера до рабочей станции и обратно не представляется возможным. При сбое локальной сети шанс потерять файл большого размера очень велик.

В то же время рабочая станция – вещь куда менее надежная, чем сервер, и постоянно хранить на ней данные небезопасно, к тому же остальные участники проекта должны как-то знакомиться с работой друг друга. Для таких ситуаций continuous backup просто незаменим как средство сохранения и дублирования данных для дальнейшей работы.

Continuous backup также применяется, когда необходима работа в режиме non-stop и даже небольшая задержка на несколько минут может привести к высоким финансовым потерям. Например, при работе с денежными потоками. В этом случае полезно синхронизировать имеющиеся данные с дополнительной системой хранения, а уже оттуда снимать резервную копию. При таком подходе администратор резервного копирования не привязан к пресловутому «окну бэкапа» и может построить более удобный график создания резервных копий.

Зачем все это нужно, если есть встроенные решения?

В некоторых системах присутствуют попытки создать некие заменители continuous backup, например Shadow Copy в системах от Microsoft. Но основное отличие «непрерывного бэкапа» от «теневого копирования» заключается в том, что Shadow Copy создает скрытые резервные экземпляры файлов на том же дисковом томе и по расписанию. Сразу обнаруживаются два недостатка: резервная копия хранится на том же массиве, что и данные, и это само по себе весьма опасно. К тому же копии создаются не непрерывно, что не позволяет восстанавливать данные на момент сбоя или на другой произвольный момент времени. То же самое можно сказать и о других системах синхронизации данных, основанных на запуске заданий по графику, например о DFS. В случае репликации, посредством механизма DFS в системах файлы синхронизируются на определенный момент времени. Это значит, что нельзя восстановить данные в произвольный момент времени, например созданные пятнадцать минут назад.

У Continuous backup есть еще одна дополнительная «профессия», о которой не любят распространяться. На сегодняшний момент это прекрасный инструмент для контроля действий сотрудников. При условии, что все файлы, созданные на компьютере, синхронизируются с файловым хранилищем, можно не бояться, что сотрудники при угрозе увольнения уничтожат ценные файлы. Руководство компании также может наблюдать, как продвигается работа над тем или иным проектом, не на словах, а на деле, анализируя файлы, создаваемые сотрудниками.

## 6.9. Заключение

В этой главе мы познакомились с различными схемами ротации носителей и узнали о механизмах, позволяющих улучшить и расширить область применения резервных копий. Помимо экономии средств, этот простой механизм позволяет упорядочить сам процесс резервного копирования и облегчить процесс восстановления.

# Глава 7



## Цели и технологии резервного копирования. О различиях в технологии резервного копирования

*Если ваш системный администратор – восторженный оптимист, это значит, что в скором времени все остальные сотрудники станут угрюмыми пессимистами.*

(Народная примета)

### 7.1. Дуализм систем резервного копирования

При создании систем резервного копирования огромное количество времени уделяется техническим факторам, таким как объем файлов, время копирования, точка восстановления. Но если не ответить на один-единственный вопрос, все может пойти совсем не так, как задумывалось. Этот вопрос: «А с какой целью выполняется резервное копирование?»

«Что значит с какой целью? Чтобы был бэкап!» – возмутится нетерпеливый читатель. И, в принципе, он будет прав, за исключением одного маленького нюанса: «А что и как мы будем восстанавливать?»

У резервного копирования цели и задачи могут быть абсолютно разные. Например, можно быстро восстановить почтовый сервер для выполнения функций приема-передачи сообщений и не спеша реанимировать почтовую базу. Или наоборот, развернуть бухгалтерскую базу данных на другом сервере СУБД и не спеша заниматься переустановкой системы.

Именно поэтому различают две цели резервного копирования: архивное резервное копирование и подготовка к Disaster Recovery.

### 7.1.1. Подробнее о задачах резервного копирования

Как я уже писал выше, различают две основные задачи: восстановление инфраструктуры при сбоях (Disaster Recovery) и ведение архива данных с целью последующего обеспечения доступа к информации за прошлые периоды. Классическим примером Disaster Recovery является образ системной партии сервера, созданный соответствующей программой: Acronis Backup and Restore, Part Image или любой другой.

Примером архива может выступить ежемесячная выгрузка баз данных из 1С с последующим хранением на кассетах. Иногда возможно совмещение этих задач, например годовой «набор» ежемесячных полных «снимков» файлового сервера посредством все того же Backup and Restore с инкрементными копиями в течение недели для полного восстановления сервера при полном выходе из строя.

Самое главное здесь – четко понимать, для чего делается резервирование. Приведу пример: вышел из строя критичный SQL-сервер по причине отказа дискового массива (одновременный сбой 2 дисков в RAID5). На складе есть подходящее аппаратное обеспечение, поэтому встал вопрос только в восстановлении программного обеспечения и данных. Руководство компании обращается с понятным вопросом: «Когда заработает?» – и удивляется, узнав, что на восстановление уйдет минимум 4 часа. Дело в том, что на протяжении всего времени осуществлялось резервное копирование исключительно баз данных без учета необходимости восстановить сам сервер со всеми настройками, включая программное обеспечение самой СУБД.

Другой пример. Молодой специалист на протяжении всего периода своей работы создавал посредством программы ntbackup одну-един-

ственную копию файлового сервера под управлением MS Windows 2003 Server, включая данные и System State в общую папку другого компьютера. На данном сетевом ресурсе имелся дефицит дискового пространства, и эта копия постоянно перезаписывалась. Через некоторое время его попросили восстановить предыдущий вариант многостраничного отчета, который был поврежден при сохранении. Понятное дело, что, не имея архивной истории, с выключенным Shadow Сору он не смог выполнить этот запрос.

Нужно ясно осознавать, для чего выполняется резервное копирование. Не стоит смешивать бэкап для полного восстановления при сбоях и создание архива за период. И под каждую задачу лучше использовать свой инструмент, хотя есть и универсальные решения. Например, Symantec Backup Exec имеет функции восстановления «на голое железо», или Acronis Backup and Restore позволяет вести архивное копирование, но это, скорее, частные случаи общего правила. По сути, такая универсальность может означать только одно: в одном продукте собраны два инструмента для разных ситуаций.

Основным определяющим критерием для формирования целей и задач резервного копирования являются требования бизнеса. И если нужно восстановить критичный сервис за кратчайшее время, необходимо подумать не только об архивации данных, но и о наличии средств быстрого восстановления операционной системы и другого установленного программного обеспечения, например посредством снятия образов системной партии. Пример ведения архивной копии показан на рис. 7.1.

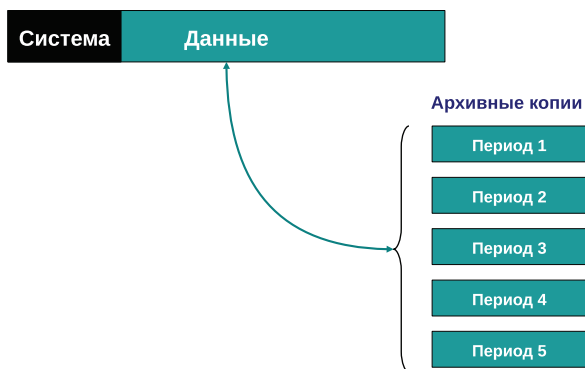


Рис. 7.1. Создание архивной копии

Как видим, при создании архивной копии собираются только данные, которые хранятся в течение длительного времени. Операционная система, особенности настроек профилей пользователей и другие будничные вещи из работы системных администраторов исторической ценности не представляют.

А вот при создании копии для Disaster Recovery поступают совсем по-другому. Резервному копированию подлежит вся информация без исключения. Основная задача здесь – максимально быстро восстановить систему до аварии «все как было» (см. рис. 7.2).

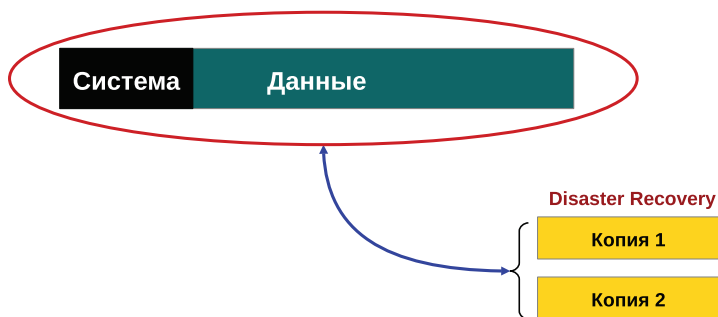


Рис. 7.2. Копирование для Disaster Recovery

Обратите внимание, что при копировании для Disaster Recovery нет смысла хранить историю бэкапов за длительный период. Основная обязанность администратора резервного копирования – позаботиться о том, чтобы у него были две полные копии работающей системы. Почему нужны именно две копии? Если из-за аварии одна не успеет завершиться, всегда в наличии вторая копия для восстановления.

Принято считать, что копии Disaster Recovery необходимо держать как можно ближе к объекту восстановления, в то время как архивные копии обычно помещают в специальное хранилище вне территории предприятия.

Однако с развитием систем восстановления от этого правила все чаще стали отходить. Во второй главе «Термины и определения» я рассказал о различных уровнях Disaster Recovery. Например, в случае применения системы восстановления 3-го уровня резервные копии регулярно перемещаются на резервную площадку, чтобы при необходимости (например, при аварии) запустить копию основной системы. Резервные копии служат при этом основным источником данных для восстановления.

Основные различия между этими целями сведены ниже в табл. 7.1:

**Таблица 7.1. Сравнение целей и задач резервного копирования**

	<b>Архивная копия</b>	<b>Disaster Recovery</b>
Резервная копия содержит	Только данные, созданные пользователями	Все необходимое для полного восстановления: данные, системные файлы, файлы настроек и т. д.
Срок хранения	В течение длительного периода	Очень короткий, при этом копия постоянно обновляется
Местоположение копии	Вдали от основной ИТ-структуры	Всегда под рукой
Основное предназначение	При анализе за прошлый период	При быстром восстановлении всей системы



В самом понятии Disaster Recovery с самого начала заложена некоторая путаница. В одних случаях под этим термином понимают восстановление системы на «голое железо» (bare metal). В другом варианте под данным процессом понимается возможность «вернуть» настройки на только что установленную систему, как, например, копирование конфигурационных файлов из папки / etc и других в Unix-like-системах (в Windows этому примерно соответствует копирование и восстановление System State). Конечно, при таком раскладе сервер будет введен в работу не ранее, чем будет проинсталлирована операционная система и восстановлены необходимые установки, что займет гораздо более длительное время. Но в любом случае решение, каким быть Disaster Recovery, проистекает из потребностей бизнеса.

## 7.2. Технология резервного копирования

Резервное копирование данных производится при помощи двух основных технологических направлений:

- на уровне файлов;
- на уровне блоков.

### 7.2.1. Технология файлового копирования

Как это предполагается из названия, обычно используется при резервировании информации на уровне файлов (см. рис. 7.3). Основной плюс данной технологии заключается в возможности гибкого отбора сохраняемых файлов. Например, только один каталог, файлы по шаблону или изменившиеся за определенный период времени.

Минусом технологии является довольно долгий процесс копирования. Это связано с необходимостью каждый раз получать доступ к объекту на уровне файловой системы. Также замедление работы может быть связано с длительной процедурой поиска и отбора файлов по признакам, указанным в задании резервного копирования.

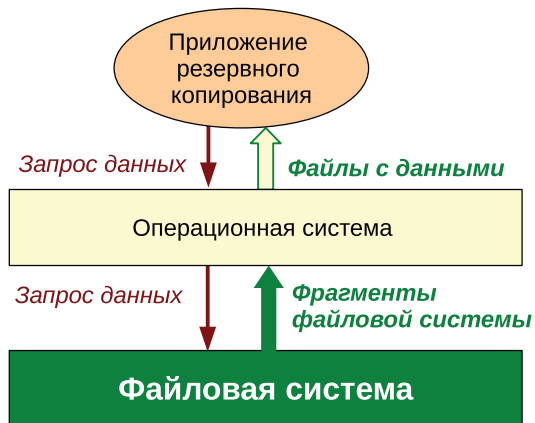


Рис. 7.3. Пофайловая технология резервного копирования

Еще одна неприятная особенность данной технологии – необходимость бороться с блокировкой открытых для записи файлов. Обычно, когда файл открыт для записи, операционная система не дает получить доступ к его содержимому другой программе. В результате если процесс резервного копирования «натывается» на такой открытый файл, он в зависимости от настроек либо останавливается, либо пропускает данный файл. Существуют различные методы обхода данной проблемы, например повторная попытка получить доступ через некоторый временной интервал или использование службы теневого копирования томов VSS в Windows. Но факт остается фактом – необходимо осуществлять дополнительные действия по преодолению трудностей.

### 7.2.2. Технология блочного копирования

Значительно отличается от файловой и на первый взгляд не имеет недостатков. В случае с блочным бэкапом доступ к данным осуществляется посредством специального драйвера с прямым доступом к диску в обход файловой системы (см. рис. 7.4). Сам блок представляет собой наименьший адресуемый элемент диска. Такие вот «кирпичики», из которых состоит файловая система.

Особенности технологии блочного резервного копирования:

- данные копируются одним потоком;
- файловая система не задействуется, или ее участие минимально;

- нет понятия открытых файлов;
- при инкрементном и дифференциальном копировании сохраняются только занятые измененные блоки, что значительно меньше, чем резервировать файлы целиком.

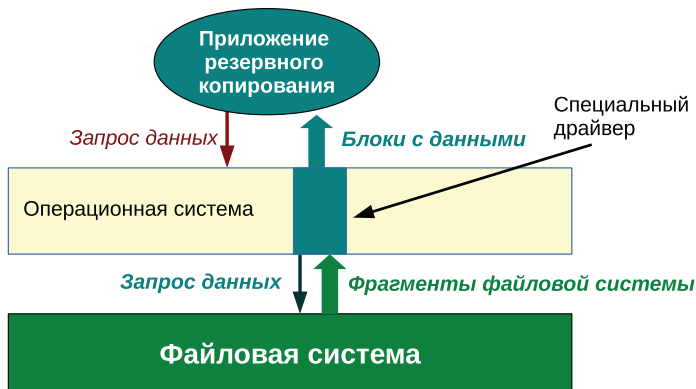


Рис. 7.4. Технология блочного резервного копирования

Обычно технологию блочного копирования используют для копирования дисковых разделов целиком, но возможны и исключения. Например, программа Acronis Backup and Restore позволяет исключить из процесса резервирования файл подкачки, а также файлы, соответствующие определенному шаблону, например с расширением \*.bak, \*.tmp и др. В этом случае система резервного копирования все равно вынуждена обращаться к файловой системе, но не для считывания файлов, а для получения информации о том, какие файлы (а следовательно, блоки, из которых они состоят) следует пропустить.

Строго говоря, существуют два режима блочного копирования. В одном случае происходит побитовое копирование дискового раздела на другой диск или в файл образа. В этом случае программу абсолютно не интересует ни тип файловой системы, ни занимаемый объем. Естественно, получившийся объем резервной копии будет равен полному объему дискового раздела, включая свободное пространство. При этом процесс протекает довольно медленно. Единственным плюсом в данном случае является возможность копировать любые дисковые разделы с самыми экзотическими файловыми системами, незнакомые ни для программы резервного копирования, ни для операционной системы, в которой

эта программа работает. Также можно сохранять зашифрованные тома (разумеется, без возможности доступа к самим данным).

В другом случае программа блочного копирования знает, как обратиться к файловой системе за информацией о свободном (незанятом) пространстве и файлах, подлежащих исключению из конечной копии. В итоге сам процесс проходит значительно быстрее, и конечный объем резервной копии может быть значительно меньше. Но программа должна уметь работать с файловыми системами копируемых разделов. Разумеется, данный метод бесполезен, если раздел зашифрован. Причиной такого ограничения является все то же считывание копируемой информации «в обход» операционной системы, сразу с тома, содержащего необходимые данные. Например, Acronis Backup and Restore не может работать с файловой системой при блочном бэкапе разделов, зашифрованных программой Secret Disk. Приходится копировать раздел побитно, один к одному, или воспользоваться файловым методом.

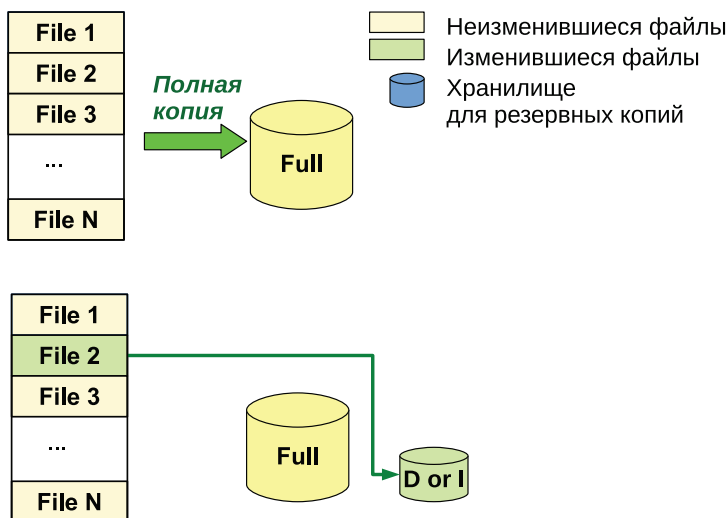


Рис. 7.5. Копирование изменений по файловой технологии (измененные файлы копируются целиком)

Технология блочного резервного копирования также позволяет создавать дифференциальные и инкрементные резервные копии, что может значительно сократить объем копий и ускорить процесс ре-

зервного копирования. При этом стоит отметить, что на самом деле инкрементные и дифференциальные копии создаются быстрее, так как копируется не каждый измененный файл целиком, а только изменившиеся небольшие блоки.

Минус у блочного метода в чистом виде только один: недостаточно гибкости при определении объектов резервного копирования. Нельзя указать в задании: «Вот эту директорию мы копируем, а эту нет, а здесь выбираем только несколько файлов». Приходится копировать либо весь раздел целиком (за исключением некоторых файлов), либо воспользоваться услугами «старого доброго» файлового метода. Именно поэтому в последнее время все чаще встречаются программные продукты, сочетающие в себе оба метода. Такая стратегия позволяет использовать плюсы обоих методов.

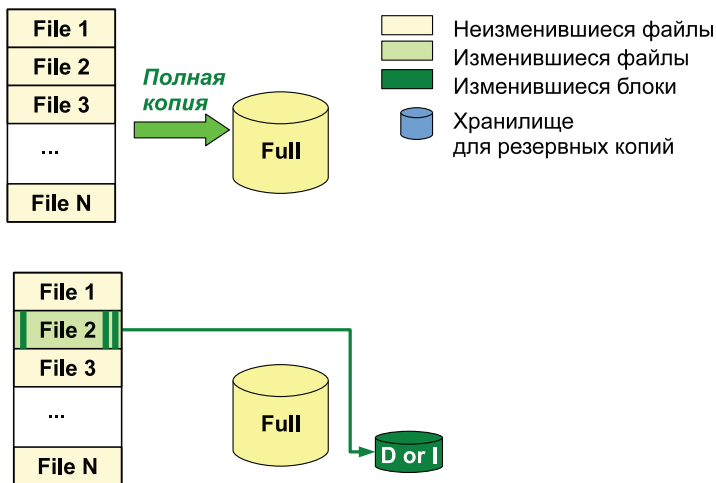


Рис. 7.6. Копирование изменений по блочной технологии (копируются только измененные блоки)

### 7.2.3. В каких случаях применяют ту или иную технологию?

Блочный метод хорошо подходит для копирования в целях аварийного восстановления системы, когда необходимо получить полностью работоспособную систему в сжатые сроки. В таких случаях всегда полезно иметь под рукой полный «слепок» файловой системы рабочего сервера, чтобы провести быстрое восстановление.

Также данная технология хороша, когда на дисковом томе содержится только информация, подлежащая резервному копированию согласно текущему плану.

Таким примером может послужить отдельный раздел файлового сервера, на котором хранятся исключительно рабочие данные пользователей. В этом случае можно сделать имидж всего раздела и быть уверенным, что в него попали все подлежащие копированию файлы.

Также блочный бэкап используется при конверсии физических машин в виртуальные. Сначала создается полный образ системы, после чего он разворачивается на базе заранее созданной виртуальной машины. Плюсом данного метода, в отличие от стандартных утилит конверсии, является независимость от гипервизора. Созданный имидж системы можно развернуть одинаково хорошо в виртуальной среде WMvare ESX/ESXi, Citrix XenServer и т. д.

Есть еще одна сфера применения блочного бэкапа: расследование инцидентов, в том числе инициированных службой безопасности. В этом случае очень удобно иметь под рукой образ системы за определенную дату, чтобы была возможность посмотреть, что же на самом деле хранилось на исследуемом томе.

В свою очередь, файловое копирование подходит для случаев, когда нужно провести сложный селективный отбор копируемых или невозможна работа специальных драйверов, работающих в обход файловой системы. (Например, конфликт с другим программным обеспечением).

В случае, когда необходимо обеспечить быстрое аварийное восстановление, иногда поступают следующим образом: загрузившись с другого носителя, например с CD-диска, однократно создается образ системной партиции и других дисковых разделов, подлежащих быстрому восстановлению в режиме «один к одному», после чего сервер снова возвращается в нормально работающий режим, а файловое резервное копирование используется для сохранения изменений. Конечно, существует очень большая вероятность, что подобную операцию придется повторять, чтобы избежать серьезных расхождений, особенно если система претерпевала изменения. Но в любом случае лучше запланированная остановка на время выполнения резервной копии, нежели не определенный по времени простой по причине серьезного сбоя. Особенно часто такой комбинированный метод используется на период внедрения и апробирования продуктов, технологий и других нововведений, когда может потребоваться быстро вернуть систему к первоначальному состоянию.

Таким образом, при серьезном сбое сначала производится восстановление из раннего образа, а после «накатываются» изменения из файлового бэкапа.



В случае, когда резервному копированию подлежит работающая база данных в режиме «чтение-запись» – не подходит ни файловый, ни блочный метод резервного копирования. Это связано с тем, что часть измененных данных, подлежащих записи на диск, может находиться во временной памяти, например в ОЗУ или файле подкачки. Поэтому необходимо либо использовать встроенные средства данной СУБД для организации безопасной выгрузки данных, например в отдельный файл, либо специальные программы-агенты, которые умеют работать с базой данных. Такой метод позволяет получить данные напрямую и передать их системе для дальнейшего копирования.

Примеры программ, реализующих ту или иную технологию резервного копирования:

- классическим примером программы для файлового копирования являются `ntbackup` для Windows-систем и `tar` для систем Unix;
- в качестве примера инструментов для блочного копирования можно привести небезызвестную утилиту `dd` для Unix-систем. Более современными средствами являются программа `Partimage` и созданные на ее основе Live CD-дистрибутивы `Clonezilla` и `Redo Backup`. Более подробно мы познакомимся с `Redo Backup` в главе 18.

## 7.3. Заключение

Как видим, нельзя рассматривать процесс резервного копирования в отрыве от восстановления данных. Важно не только знать, что и когда нужно сохранять, но и для каких целей.

# Глава 8

## Резервное копирование виртуальных систем

*«Объявление:*

*Серьезной хостинговой компании требуется квалифицированная уборщица. Требования: знание современных вычислительных платформ, способность восстановить систему после своих действий».*

(Вакансия из секретной базы  
крупного кадрового агентства)

### 8.1. Предисловие

Виртуальные системы незаметно завоевали свою весьма значительную долю в сфере ИТ. Уже не только системные администраторы и инженеры, но и программисты, веб-дизайнеры и даже художники (те, кто рисуют элементы для сайтов) знают о таких системах не понаслышке. Появление облачных систем всех мастей расцветок постепенно стирает грань между виртуальными и традиционными (физическими) вычислительными машинами. Сейчас можно наблюдать обратную тенденцию, когда виртуализируется все, что попадает под руку, включая файл-сервер общим объемом в несколько терабайт.

И, как любая вычислительная система, виртуальные среды нуждаются в правильно организованном налаженном сервисе сохранения данных. При этом действовать нужно еще более осторожно и предусмотрительно, чтобы не затронуть работу других гостевых систем. Помимо резервной копии в виде служебных файлов, неплохо бы сохранить в рабочем виде и ее «богатый внутренний мир»: файловую

систему, целостность базы данных, доступность зашифрованных разделов и т. д.

Сохранять копию резервной машины, не заботясь о целостности заключенных в ней данных, – все равно, что спасти шкатулку, потеряв при этом все драгоценности из нее.

Существует множество различных систем виртуализации. Например, виртуализация серверов, виртуализация десктопов, виртуализация приложений и др. Для каждой системы существуют свои оригинальные методы сбережения данных.

В этой главе мы говорим именно о визуализации серверов, то есть об эмуляции работы физического сервера средствами специального программного обеспечения – гипервизора. Эти системы являются самыми известными и применяются практически повсеместно – от небольшого офиса до крупной облачной компании. Несмотря на то что существует множество различных программных продуктов для решения подобных задач, на самом деле не так уж и важно, какая система виртуализации серверов используется. Для подобных систем для резервного копирования и восстановления не так уж просто выдумать что-то принципиально новое. Большинство используемых методов схоже между собой. Достаточно усвоить общие принципы, и читатель сможет сам грамотно организовать сохранение данных для вверенной ему ИТ-инфраструктуры.

Ниже приводятся два основных направления для копирования виртуальных машин (серверов).

## **8.2. Вариант 1 – виртуальная машина создает свою копию самостоятельно**

Применение данного метода основано на традиционном подходе, когда специальная программа, работающая в операционной системе виртуальной машины, выполняет резервное копирование подобно обычному физическому серверу.

Например, если на предприятии используется ARCServe Backup, то нужно просто установить программу-агента и создать соответствующее задание. Если используются специализированные СУБД или почтовые системы, опять же можно использовать соответствующих агентов и получить резервную копию. Для копирования всей системы целиком в целях Disaster Recovery (аварийного восстановления) можно использовать любую доступную программу для снятия образа дисков.

### 8.2.1. Преимущества данного метода

Не нужно закупать специальных средств. То же самое оборудование и программное обеспечение можно использовать для копирования и физических, и виртуальных машин.

Гарантированное сохранение целостности данных при наличии специального программного обеспечения. Например, копирование данных SQL-сервера средствами самой СУБД проходит абсолютно одинаково как в физической, так и в гостевой системе.

Важное замечание. Следует тщательно ознакомиться с требованиями производителя программного обеспечения и применять специальные рекомендованные средства. Например, при прямом копировании файла почтовой базы данных MS Exchange можно получить просто здоровенный файл без возможности дальнейшего использования.

### 8.2.2. Ограничения данного метода

Допустим, мы сумели сохранить образ системного диска и данные пользователей. Но, для того чтобы это все восстановить, нужна ни много ни мало новая виртуальная машина. При этом необходимо учесть не только общие характеристики: количество процессоров, объем ОЗУ (RAM), размер виртуального жесткого диска, – но и гораздо более тонкие нюансы. Например, в какой resource pool входила виртуальная машина, был ли настроен overcommitting и с какими параметрами, где находился файл подкачки, был ли настроен автостарт этой виртуалки и в какую очередь и т. д.

В итоге для выполнения всего процесса восстановления потребуется значительное время.

Пока речь идет только об одной виртуальной машине, такой способ кажется сложным, но приемлемым. Но что будет, если выйдет из строя дисковое хранилище, на котором размещено большое количество «виртуалок»? Восстанавливать все гостевые системы вышеописанным способом фактически будет означать создание всей виртуальной структуры с нуля.

## 8.3. Копирование виртуальной машины целиком

Подход напоминает блочное резервное копирование. И в том, и в другом случае создается образ нужного раздела целиком. Соответственно, при развертывании не нужно тратить времени на установку операци-

онной системы, прикладного программного обеспечения и выполнять общую настройку сервера. При этом копирование виртуальной машины происходит на более «низком» уровне – средствами самого гипервизора, что позволяет выполнять такие операции быстрее.

Гостевая система сама по себе представляет набор файлов, расположенных на хранилище, к которому имеет доступ гипервизор. Поэтому системе резервного копирования достаточно получить доступ к файловой системе хранилища и выборочно сохранить ее содержимое в другом месте. Благодаря унификации оборудования в рамках решений от одного вендора восстановить саму виртуальную машину не составляет труда.

Но, несмотря на кажущуюся простоту, неправильное использование этого метода может преподнести неприятные сюрпризы. Например, если просто сделать копию файлов виртуальных машин, то пропадет информация, хранящаяся на данный момент в оперативной памяти. Также файлы, в которые в данный момент происходит запись, могут попасть в резервную копию уже с повреждениями. Для таких систем, как сервер MS Exchange или базы данных на SQL-сервере, это может оказаться дорогой к смерти.

### 8.3.1. Механизмы для копирования виртуальных машин целиком

При сохранении виртуальных машин «как есть» нужен некий механизм, позволяющий, во-первых, избежать записи на диск в момент обращения, во-вторых, который мог бы гарантировать соответствие между файловой системой и данными в оперативной памяти. То есть обеспечил бы своего рода заморозку системы в момент создания копии. Запомним этот термин: «заморозка», в дальнейшем он нам еще понадобится.

Первое, что приходит на ум, – просто запретить в момент создания резервной копии запись в данные для гостевой системы. Выглядит это, грубо говоря, так: начался процесс резервирования – ни одна программа, включая служебные модули операционной системы, не имеет возможности писать на диск. Хорошо бы при этом еще и остановить работу всех приложений, которые даже чисто теоретически могут изменять содержимое диска. А еще неплохо бы принудительно закрыть все открытые файлы и сбросить содержимое оперативной памяти на жесткий диск. Вам это ничего не напоминает? Неполный аналог этому состоянию – спящий режим на персональных компьютерах. А для полного удовлетворения вышеописанных условий полноценного безопасного резервного копирования необходимо просто временно выключить виртуальную машину.

В простых случаях так и поступают – запускается специальный скрипт, делающий shutdown гостевой системы, после чего просто копируют необходимые файлы. Хороший пример – прокси-сервер небольшой компании. Согласитесь, что вероятность нахождения кого-то из сотрудников глубокой ночью в выходной день на территории офиса не слишком велика. А если при этом именно для работы требуется доступ во Всемирную паутину – это вообще редчайший случай. И даже если найдется такой желающий, его можно предупредить заранее о том, что придется потерпеть недолгий промежуток времени, пока делается бэкап и снова запускается прокси-сервер. Другое дело, когда речь идет о сервисах, которые нельзя останавливать таким образом. Например, в случае с серверами высокой доступности или при наличии открытых пользователей файлов на сетевом ресурсе. Да мало ли по какой причине нельзя вот так запросто выключить «ту самую виртуалку, на которой все крутится».

Так что же нужно, чтобы в режиме штатной работы с виртуальным диском и оперативной памятью создать резервную копию, сохранив целостность данных?

Очевидно, необходимо предоставить возможность для приложений и операционной системы временно записывать данные в другое место, чтобы после окончания бэкапа переместить эти данные на основное местоположение. Мы не зря вспомнили блочный бэкап – там эта задача была решена еще до появления систем виртуализации. Например, такие системы, как Acronis Backup & Recovery или Paragon Drive Backup, умеют снимать образ системного диска во время штатной работы компьютера. В этом случае данные временно пишутся на другой участок раздела, а после окончания процесса резервного копирования выполняется слияние (merge) с фрагментами, запись в которые была запрещена во время резервного копирования.

В ситуации с виртуальными машинами все еще гораздо проще. Если успех блочного бэкапа зависит от множества факторов, например совместимости драйверов и агентов резервного с другим ПО, то в случае с резервным копированием виртуальных машин можно попросить гипервизор просто создать временный файл и писать данные туда. Такой механизм называется «моментальный снимок системы», или «снапшот» (snapshot). А процесс остановки записи на основной виртуальный диск называется «freezing» (заморозка).

Внимательный читатель вспомнит, что о снапшотах уже шла речь в 3-й главе, когда разбирали методы избыточности для повышения отказоустойчивости дисковых систем. Рассматривались методы сору-

on-write и redirect-on-write. Так вот, в виртуальных системах почти всегда используется метод redirect-on-write. Это связано с необходимостью получить слепок нетронутой системы. Ситуация, когда основные данные «замораживаются» на определенный момент времени, а все изменения записываются в специально отведенную область, по сути, и есть работа алгоритма redirect-on-write. Вторая причина – необходимость поддерживать уровень быстродействия. У redirect-on-write скорость работы при выполнении рутинных операций гораздо выше из-за отсутствия необходимости делать «двойное копирование». Но когда происходит слияние снимка и основной системы (эту процедуру называют «merge»), тут уж мало не покажется, бывает, что система «замирает» на некоторое время. Но в целом суммарные затраты быстродействия при redirect-on-write все равно ниже.



Не забываем, что метод создания снимков системы используется при обслуживании не только виртуальных, но и других систем. Например, в аппаратных файловых хранилищах. При этом под снимкты обычно резервируется 20–25% от всего объема хранилища, что уменьшает полезное место.

Существуют две разновидности снимктов. В одном случае просто создается временный диск («дельта-диск»), а запись в основной прекращается. В другом – помимо дельта-диска, создается еще и файл дампа оперативной памяти, в который сбрасывается содержимое ОЗУ гостевой системы. Как правило, наличие или отсутствие того или иного механизма зависит не только от версии, особенностей реализации гипервизора и системы резервного копирования, но и от стоимости используемой лицензии. Чем дороже лицензия – тем больше функций. В самых дорогих редакциях (они обычно носят громкие названия вроде «Gold», «Enterprise», «Platinum») включены все доступные на данный момент возможности.



Как бы чудесно не обстояли дела с созданием снимктов, самый надежный способ резервного копирования виртуальных машин – сохранять гостевую систему в выключенном состоянии. В этом случае всегда есть очень хороший шанс восстановить все в целости и сохранности.

### 8.3.2. Особенности резервного копирования виртуальной машины с использованием снимка

Как я уже писал выше, стандартная процедура сохранения виртуальной машины выглядит так: сначала создается моментальный снимок (иногда еще и дамп памяти), потом производится копирование гостевой системы в замороженном состоянии (в это время все изменения пишутся в дельта-диск снимка), после чего снимок удаляется (см. рис. 8.1).

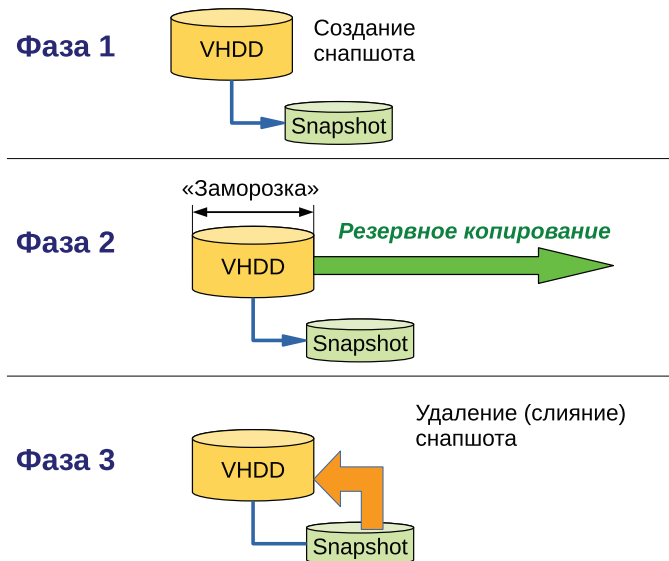


Рис. 8.1. Использование моментальных снимков при копировании виртуальных машин

На деле все оказывается не так просто. Во-первых, наличие снимка снижает быстродействие гостевой и всей хост-системы. Во-вторых, каждый моментальный снимок увеличивает процент отказа виртуальной машины. Если программа резервного копирования не умеет аккуратно подчищать за собой временные снимоты, то существует очень большая вероятность, что через какое-то время вы останетесь и без бэкапа, и без работающей виртуальной машины. Что делать в таких случаях?

Для начала нужно попытаться удалить временные снимки вручную через стандартный механизм управления снимотами. Если уже имеется несколько снимков, убирать их необходимо строго по одному (!), начиная с самого последнего. Не следует пытаться удалить сразу все снимоты за один раз или начинать с тех, которые находятся в середине «цепочки» (см. рис. 8.2).



Ни в коем случае нельзя пытаться напрямую удалять файлы моментальных снимков на хранилище host-сервера. Любая попытка такого вмешательства в работу виртуальной среды приведет к потере работоспособности гостевой системы. В «особо счастливых» случаях таким способом можно убить не одну, а сразу несколько виртуалок. Вообще, при работе с виртуальными

системами следует руководствоваться принципом: «запрещено все то, что явно не разрешено», – и прежде чем что-то предпринять, сначала много читаем по данному вопросу, затем тщательно все анализируем и только потом что-то делаем.

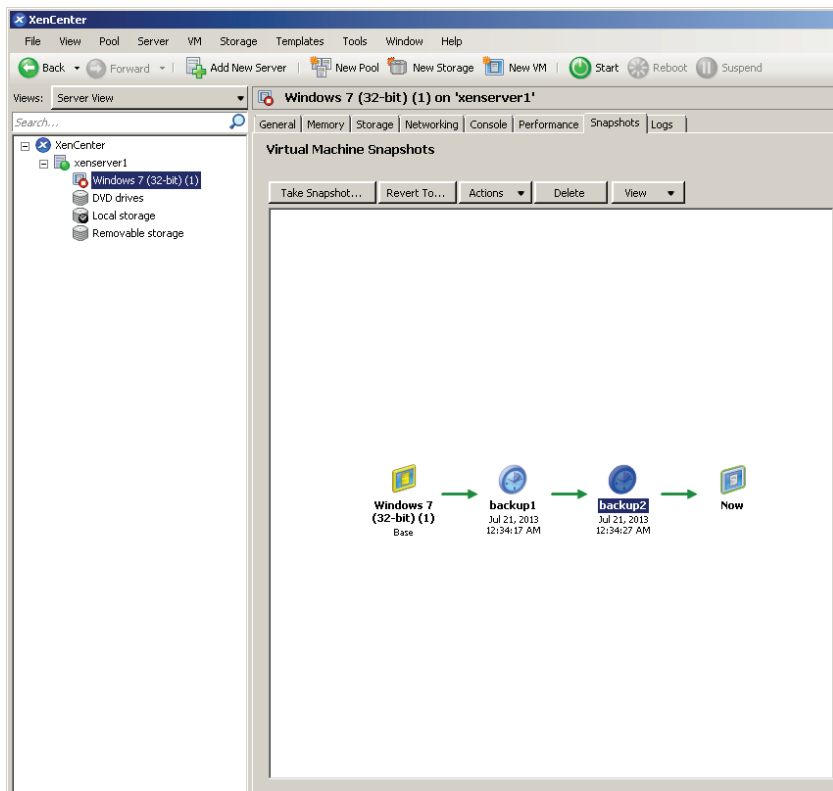


Рис. 8.2. Менеджер управления снимками Citrix XenCenter показывает несколько моментальных снимков

А что делать, если из-за сбоя моментальный снимок не удаляется? К сожалению, такая ситуация – далеко не редкость. В этом случае крайне желательно разобраться, почему это произошло. Если причина – некорректная работа ПО при создании снимкота, для начала нужно попробовать просто удалить снимок вручную, используя привычные инструменты управления снимкота. Если это не получается, то можно воспользоваться следующим приемом: создать штат-

ными средствами гипервизора еще один снимок, после чего выбрать пункт «Удалить все» (см. рис. 8.2). Такую процедуру желательно делать только в случае, если моментальных снимков немного – не больше трех. И хотя правильнее удалять снапшоты по одному, в данном случае имеет место форс-мажорная ситуация, и для ее устранения все средства хороши.

Другое дело, когда неудаляющиеся снимки появляются по вине ошибок самого хранилища. Например, из-за некорректной работы дискового массива. В этом случае необходимо выполнить миграцию виртуальной машины на другое хранилище.

По моему личному опыту, если доступна функция «Live Migration» (в разных системах виртуализации она может называться по-разному), то использовать стоит именно ее. Дело в том, что ошибки на дисковом хранилище также приводят к чудесной ситуации, когда выключенная виртуальная машина больше не включается. «Live Migration» помогает избежать необходимости выключать гостевую систему.

И наконец, если было испробовано все возможное, а «воз и ныне там», то можно просто клонировать виртуальную машину (сделать копию) средствами управления гипервизора. При этом новая гостевая система не будет иметь снапшотов. Проводить такую операцию во включенном или выключенном состоянии – необходимо смотреть по обстоятельствам. Вариант с выключенной гостевой системой, как правило, более надежен и имеет больше шансов на успешный запуск. И всегда стоит лишний раз задуматься над вопросом: «Возможно ли будет запустить исходную виртуальную машину после выключения?»

Лично я в любом случае, перед тем как начинать что-либо делать с гостевой системой, обязательно клонирую ее для создания полной актуальной резервной копии. После этого проверяю работоспособность вновь созданной «виртуалки» в режиме с отключенными виртуальными сетевыми интерфейсами. И только после успешного результата приступаю к манипуляциям с исходной системой.

«Зачем же было мучиться, если можно было сразу клонировать исходную систему и запустить новую?» – спросит настороженный читатель. Дело в том, что копия всегда хоть чем-то отличается от оригинала. Например, может смениться сетевой MAC-адрес или серийный номер BIOS. В результате все рано приходится перенастраивать вновь созданную копию того, что ранее работало. На моей памяти, как правило, приходилось переделывать сетевые настройки UNIX-like операционных систем. В общем, если удастся избавить гостевую

систему от ненужных снимков, не прибегая к процедуре клонирования, желательно действовать именно так.

И конечно же, самое лучшее решение – не использовать программного обеспечения, создающего такие проблемы. Поэтому, прежде чем что-то приобретать, необходимо тщательно протестировать ПО (пусть даже усеченную или пробную версию) на виртуальной системе для определения возможных сбоев. Использование дополнительных тестовых систем виртуализации, работающих отдельно от production, только поначалу кажется лишними расходами. На деле это здорово помогает экономить деньги, страхуя бизнес от бесполезных расходов в случае отказа рабочих систем.

## 8.4. Виртуальный шторм и его последствия

Все это время мы говорили о копировании одной виртуальной машины. Но что происходит, когда таких машин несколько? Процесс создания резервной копии любит транжирить ресурсы системы. Резко возрастает нагрузка на дисковую подсистему, плотным потоком данных забивается сетевой интерфейс, оперативная память занимает дополнительные блоки данных, да и процессор в этот момент не скушает. На моей практике были случаи, когда приходилось отключать задание резервного копирования, чтобы приложение по обработке данных успело завершить очередной цикл обработки.

А теперь вернемся к нашей виртуальной среде. Сейчас принято размещать на одном host-сервере от десяти и более виртуальных машин. В результате, если в один короткий период запустить процесс резервного копирования нескольких гостевых систем, можно не только не завершить бэкап, но и вызвать зависание копируемых виртуалок, а если очень «повезет», то и напроць повалить всю хост-систему (см. рис. 8.3).

Как можно избежать подобной ситуации? Во-первых, нужно более чутко и внимательно относиться к составлению расписания. Нужно знать максимальное и среднее время выполнения каждого задания, чтобы не допускать одновременной работы критического количества процессов.

Во-вторых, необходимо серьезно относиться к определению максимальной нагрузки на гостевые и host-системы, а также уделить время вдумчивой настройке распределения ресурсов. Большинство

современных систем виртуализации имеют замечательные средства для этого. Если не использовать эти возможности, понадеявшись на настройки по умолчанию, виртуальный шторм рано или поздно произойдет, причем в самый нежелательный момент. Очень часто системный администратор, отвечающий за работу виртуальных систем, распределяет ресурсы виртуальной системы, руководствуясь средними показателями работы в штатном режиме. Толку от такого подхода немного, все равно что от вычисления средней температуры по больнице. Нужно учитывать возможные требования пиковой нагрузки, а также особенности прикладного программного обеспечения по созданию этих самых «пиков». Например, если точно известно, что в последнюю пятницу каждого месяца финансовый отдел формирует месячный отчет, который «под завязку» нагружает сервер баз данных, стоит перенести еженедельный полный бэкап этого сервера на сутки позже, чтобы и отчет успеть сгенерировать, и резервному копированию ничего бы не помешало.

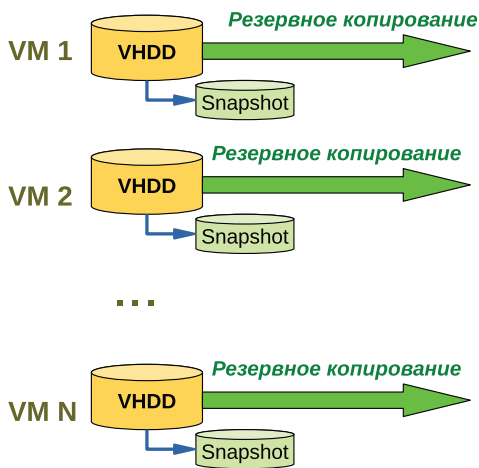


Рис. 8.3. Схема возникновения виртуального шторма

И, конечно, при просмотре результатов резервного копирования нужно обязательно обратить внимание не только на параметры «выполнился – не выполнялся» и сколько места осталось на носителе, но и к какому временному падению быстродействия системы это привело на тот момент.

## 8.5. Заключение

В завершение мне хотелось бы дать несколько полезных советов по эффективной организации резервного копирования виртуальных машин.

Сочетайте несколько вариантов. Допустим, необходимо организовать сохранение SQL-сервера. Для полной уверенности нужно не только провести резервное копирование всей виртуальной машины целиком, но и отдельно выполнить бэкап баз данных штатными средствами самого SQL-сервера или специального ПО. В противном случае есть риск оказаться «у разбитого корыта», когда виртуальная машина будет восстановлена, а данные – нет. Особо хитрые и расчетливые системные администраторы хранят отдельно образ заранее настроенных гостевых систем и отдельно – копии данных. Это позволяет им быстро восстанавливать систему и при этом не хранить дублирующие гигабайты. Для ленивых сисадминов можно предложить вариант, когда предварительно делается копия данных (например, дамп базы SQL-сервера) на тот же виртуальный диск гостевой системы, а потом все вместе копируется средствами бэкапа виртуальных машин. Это тоже помогает сохранить целостность информации в резервной копии, но все-таки вариант с отдельным хранением данных и образа системы для Disaster Recovery выглядит более надежным.

Избегайте больших нагрузок как на гостевую, так и на host-систему в целом. Регулярно следите за нагрузкой во время резервного копирования. Следует также помнить, что после окончания резервного копирования системе может потребоваться некоторое время, чтобы «прийти в себя». В таких случаях показатели высокой загруженности еще некоторое время остаются выше нормы. Это следует учитывать как при составлении расписания, так и при мониторинге нагрузки.

Проводите регулярные учения по восстановлению из резервной копии. Иначе можно оказаться в критический момент с «пустыми руками», но в полной уверенности, что «с бэкапом все в порядке». Лучше лишний раз все досконально проверить. Исходя из моего опыта, хороший системный администратор – это человек, который предпочитает лишний раз удостовериться в отсутствии проблем.

# Вопросы ко второй части для закрепления материала

1. Чем отличаются инкрементальный, дифференциальный и обратный инкрементальный методы резервного копирования?
2. Для чего применяются схемы ротации носителей?
3. Что такое Continuous backup?
4. Что является причиной дуализма в резервном копировании?
5. Чем отличаются «пофайловая технология резервного копирования» и «технология блочного резервного копирования»?
6. Когда начинается «виртуальный шторм»?

Часть III



# ОБОРУДОВАНИЕ ДЛЯ СИСТЕМ РЕЗЕРВНОГО КОПИРОВАНИЯ

# Глава 9

## Системы хранения резервных копий – единство и борьба противоположностей

*– Нет, все-таки жесткие диски лучше, чем эти ваши ленты. В харде есть магнетики, а из пластин получаются красивые блестящие подставочки под чашку с кофе. А с ленты что взять?*

*– Ну-у, например, можно сделать елочную гирлянду на Новый год...*

(Из разговоров в отделе системного администрирования)

### 9.1. Предисловие

В наше время ИТ-специалисты сталкиваются подчас с довольно непростой задачей: что выбрать для записи и хранения резервных копий. Это традиционно является одной из главных задач при создании систем резервного копирования. Долгое время данную нишу традиционно занимали ленточные накопители. Выражение «записать на ленту» стало практически синонимом слов «сделать резервную копию». Но прогресс не стоит на месте, и жесткие диски становились все вместительнее и быстрее в работе, а стоимость хранения единицы информации на них уменьшалась. Сейчас можно приобрести дисковую подсистему дешевле, чем стоит ленточная библиотека с комплектом кассет того же объема. Возникают резонные сомнения: а что лучше и использовать? И есть ли какие-либо нюансы в данном вопросе?

Чтобы получить ответ на данный вопрос, нужно внимательнее рассмотреть особенности каждой системы.

## 9.2. Что из себя представляют дисковые хранилища

Устройства данного класса представляют собой отдельные диски или дисковые массивы с возможностью записи по сети или через локальное подключение.

При работе с ними можно использовать простые протоколы доступа, такие как CIFS, FTP, WebDAV.

На сегодняшний день это наиболее распространенные носители для систем начального уровня. То есть если администратор самостоятельно мастерит систему резервного копирования из подручных средств, вероятнее всего, это будет некий файл-сервер или другое устройство, целью создания которого является запись резервных копий на жесткие диски.

Это также создает поле для маневра в маркетинговой политике, и уже многие вендоры поставляют современные системы резервного копирования, основанные именно на дисковых системах.

Свидетельством того, что дисковые подсистемы хранения все более стремительно набирают популярность, может послужить функция эмуляции ленточного накопителя посредством создания специального файла-контейнера. Примером таких систем может послужить технология Virtual Tape Library (подробнее о ней будет рассказано ниже), когда функция записи на ленту является всего эмуляцией, созданной на дисковом массиве. Сейчас практически все системы резервного копирования могут записывать копии собранных данных на дисковые массивы подобным образом.

### 9.2.1. Преимущества дисковых подсистем

Простота организации и относительная дешевизна. На начальном этапе не требуется дорогостоящее программное обеспечение для резервного копирования (достаточно операционной системы). Как я уже писал выше, можно сделать самую простую бэкап-систему, используя штатные средства операционной системы и простой компьютер с общей папкой для размещения копий.

На современных системах, основанных на жестких дисках, могут быть реализованы современные методы оптимизации работы, такие

как сжатие данных «на лету», дедупликация, моментальные снимки (snapshots).

### 9.2.2. Общие черты для всех дисковых хранилищ

Отсутствие возможности быстрого перемещения носителя.

Да, взять и просто так вытащить жесткий диск из дискового массива далеко не всегда безопасно. Во-первых, далеко не все дисковые подсистемы имеют функцию «горячего извлечения» (hotswap). Во-вторых, диски чаще всего объединяются в массивы. Ну, вытащите вы пару дисков из RAID-5, что это вам даст, кроме головной боли? А раз нельзя вытащить диск и унести его за периметр, осложняется выполнение процедуры off-site.

Вытащить из ленточного накопителя кассету и перенести ее в безопасное место вне основной территории предприятия гораздо проще, чем демонтировать, перевозить, заново монтировать и подключать целое хранилище.

Поэтому в случае использования дисковой системы за периметр перемещается не носитель с данными, а сами данные. Например, при копировании по защищенному каналу в резервный ЦОД (центр обработки данных) или облачное хранилище. Разумеется, сеть передачи данных для реализации таких задач должна иметь очень хорошую пропускную способность. На практике такие услуги стоят очень дорого, поэтому на удаленную площадку удастся скопировать только самую критичную информацию.

Еще одна интересная особенность дисковых подсистем – сохраняемый контент постоянно доступен, в том числе и для злоумышленника. Если ленточный картридж можно извлечь из накопителя и положить в сейф, то с дисковым хранилищем такой номер не пройдет. Теоретически можно предположить, что, разделив все количество дисков на массивы RAID1 (зеркало), можно изъять один из дисков, избежав краха системы. На практике такой сценарий мало осуществим из-за риска уничтожить данные – несмотря на все заверения вендоров, потеря диска в работающей системе может вызвать случайный сбой в работе даже у самых надежных систем. Не стоит забывать также о потере производительности на перестройку RAID-массива и о высокой стоимости жестких дисков. Поэтому будем считать, что информация на дисковых хранилищах доступна постоянно и, соответственно, может быть испорчена в любой момент времени, например компьютерным вирусом или же разрушением файловой системы при отключении питания.

Невозможность заменить дисковый носитель приводит к другой проблеме. Какие бы эффективные процедуры экономии дискового пространства не применялись, администратор резервного копирования рано или поздно столкнется с нехваткой дискового пространства. Из этого положения есть два выхода: постоянно наращивать объем, что трудновыполнимо с финансовой точки зрения. Или же надо урезать период хранения данных, ограничив его, например, двумя месяцами вместо желаемых двух лет. Не стоит забывать также о том, что данные имеют свойство увеличиваться в объеме. Например, бухгалтерская база данных в январе и декабре одного и того же года может различаться в 12 раз (по числу отчетных месяцев). Поэтому рано или поздно наращивать объем все равно придется. То, что в случае с ленточными накопителями решается приобретением нескольких картриджей, для дисковых хранилищ выливается в закупку и монтаж нового серверного оборудования.

В то же время бывает удобно держать данные для восстановления «всегда под рукой», особенно если, согласно принятому SLA, время простоя должно быть минимально.

Таким образом, вырисовывается своеобразная специализация дисковых систем — размещение копий для Disaster Recovery. Именно для систем быстрого восстановления подобные системы хранения подходят практически идеально. Они находятся в режиме постоянной доступности, чаще всего на той же самой территории, что и основная инфраструктура. Поэтому восстановление полного объема данных после случайного сбоя можно провести в самые сжатые сроки. Сложности с наращиванием объема не являются большой помехой, ведь для Disaster Recovery рекомендуется сохранять две копии данных: проще говоря, прошлый и позапрошлый Full backup всей системы.

### 9.2.3. Разнообразие дисковых подсистем

В прошлые времена системные администраторы, создающие систему резервного копирования на базе дисковой подсистемы, обычно не слишком мучились с выбором решения. Просто высвобождали самый старый сервер, на него устанавливали хорошо изученную операционную систему (чаще всего это была Windows, настолько древняя, чтобы могла работать с этой старой аппаратной конфигурацией) и настраивали копирование данных по сети по децентрализованной топологии.

Сейчас к услугам администраторов резервного копирования имеется широкий спектр самого разнообразного оборудования: от мощных промышленных систем до небольших сетевых хранилищ. Разумеется,

«невозможно объять необъятное» (К. Прутков), и в рамках одной главы не получится описать все используемые варианты. Ниже пойдет речь об основных разновидностях устройств хранения резервных копий на дисках.

### 9.2.4. Сетевая система хранения данных – NAS

По своей сути это обычный файл-сервер, у которого зачастую «отрезаны» другие функции. Оставлены только операции чтения-записи по сети при помощи популярных протоколов: CIFS, NFS, FTP и т. д. Управление производится удаленно, с помощью веб-интерфейса или просто командной строки. Из-за узкой специализации в качестве управляющего ПО используется усеченная версия операционной системы, обычно бесплатной, например Debian Linux или FreeBSD.

Помимо покупки готовых устройств, существует хорошая возможность создать NAS из обычного компьютера при помощи специального дистрибутива, например FreeNAS, NAS4free, OpenMediaVault и др.

#### **Преимущества NAS:**

- готовый файл-сервер. Может использовать множество протоколов и интегрироваться в различные системы централизованного управления, например Active Directory;
- доступ к данным не зависит от ОС и платформы;
- удобство администрирования;
- максимальная простота установки.

#### **Ограничения в использовании:**

- стек протоколов TCP/IP, используемый в большинстве NAS, был разработан для соединений в любых условиях. Но в то же время это приводит к высоким накладным расходам (затраты ресурсов на инкапсуляцию пакетов, обмен квитанциями и т. д.). Для скоростных сетей передачи данных такой вариант не слишком хорошо подходит. В то же время при передаче данных на большие расстояния альтернативы у TCP/IP практически нет;
- трудности масштабируемости. Недорогие готовые NAS ограничены одним корпусом с жесткими дисками. Промышленные NAS состоят из нескольких дисковых хранилищ и позволяют подключать дополнительные модули-расширения («Expansions»). Но и цена такого оборудования приближается к ленточным библиотекам и хранилищам SAN. Поэтому нельзя сказать: «Поставьте NAS-хранилище – и получите колоссальную экономию». Здесь нужен взгляд скептика, особенно

при оценке предложений с точки зрения экономической эффективности. Не секрет, что многие поставщики NAS уровня Enterprise, в том числе очень известные западные вендоры, заведомо вводят потребителей в заблуждение относительно экономических последствий внедрения таких систем.

### 9.2.5. Сеть хранения данных – SAN

Storage Area Network или SAN (дословно – сеть хранения данных, СХД) представляет собой систему передачи информации для подключения внешних устройств хранения, таких как дисковые массивы, ленточные библиотеки, при этом существующее программное обеспечение использует подключенные ресурсы как локальные. В отличие от сетевых устройств – NAS, дисковые массивы, подключенные по SAN, «видны» системе как локальные тома.

Если сетевые хранилища NAS создавались как недорогое решение, в случае с SAN ситуация полностью противоположная. Эта технология изначально позиционировалась для сектора Enterprise. Соответственно, и оборудование, работающее в подобной среде, стоит всегда довольно дорого. Отчасти это плата за отказоустойчивость и точное соблюдение заявленных характеристик, отчасти – просто дополнительный налог за громкий бренд.

Можно ли использовать дисковые массивы SAN для резервного копирования? Конечно, можно. На них точно так же можно складывать файлы резервных копий. Существует специальная технология LAN-free, позволяющая выполнять резервное копирование в обход локальной сети, используя SAN в качестве транспорта. Об этом можно прочесть в главе 12.

#### Преимущества SAN:

- большая скорость передачи данных;
- удобное централизованное управление;
- резервирование данных без загрузки локальной сети и серверов;
- хорошая масштабируемость;
- высокая готовность и отказоустойчивость.

#### Ограничения в использовании:

- высокая цена;
- необходимость использовать специализированное аппаратное обеспечение, которое может оказаться в дефиците, – далеко не все продавцы оборудования могут поставлять соответствующую технику в сжатые сроки.

## 9.3. Ленточные системы хранения

Представляют собой совокупность специальных накопителей на основе магнитной ленты и устройств чтения-записи.

Существует несколько стандартов ленточных устройств, самый распространенный из которых – LTO Ultrium.

Являются наиболее безопасными в плане снижения рисков ошибочного стирания (перезаписи), а также вирусной активности.

На основе ленточных накопителей создано множество различных устройств, от одиночных накопителей до огромных ленточных библиотек размером с серверную стойку.

### 9.3.1. Преимущества ленточных систем хранения

**Гибкость при наращивании объема** (достаточно просто докупить картриджей).

**Устойчивость к вирусам и вредоносному ПО.** В отличие от дискового массива, который постоянно доступен для чтения-записи штатными средствами операционной системы, ленточный картридж может быть извлечен из библиотеки и помещен в безопасное место. Сам факт такого изъятия предполагает, что на отдельно хранящемся носителе невозможно изменить данные. То есть вредоносная программа или злоумышленник не смогут повредить или исказить информацию на ленте. В отличие от дисковых массивов, для записи на ленточный носитель необходимо специальное программное обеспечение. Таким образом, даже если картридж все еще находится в библиотеке, повредить на нем информацию, не имея доступа к программе резервного копирования, весьма проблематично. Другое дело – NAS или файл-серверы, данные на которых подчас доступны не только для легального использования, но и для вирусов, троянских программ и других вредителей.

**Легкость переноса носителей в безопасное место.** Выше уже говорилось о том, что использовать отдельный диск из дискового хранилища практически невозможно. А ленточный носитель может быть легко извлечен из библиотеки и прочитан в другом устройстве, поддерживающем данный стандарт записи.

**Элементы системы шифрования встроены сразу в картридж.** Начиная с поколения LTO-4 в ленточный картридж встраивается специальный чип, облегчающий шифрование и дешифрование данных. Более подробно об этом будет рассказано в следующей главе.

**Наследование поколений.** Современный стандарт дисковых накопителей – LTO – позволяет производить модификацию дисковых

хранилищ, создавая и поддерживая доступность старых версий. Например, если у предприятия вся информация записана на накопители 4-го поколения (стандарт LTO-4), то возможно обновить оборудование чтения-записи до 5-го и даже 6-го поколения, прежде чем старые ленточные картриджи уже не смогут быть прочтены на новом оборудовании. При модернизации ленточной библиотеки до следующего поколения уже купленные носители прекрасно продолжают работать с новым оборудованием. При этом в одной и той же библиотеке можно использовать картриджи разных поколений.

**Характеристики ленточных систем приближаются к жестким дискам.** Устройства, поддерживающие современный стандарт LTO. У жесткого диска HDD SAS скорость обмена – до 600 Мб/с, у самого последнего стандарта ленточного массива скорость обмена может достигать 400 Мб/с.

Но существует еще один показатель – internal transfer rate, то есть скорость обмена данными между физическим носителем (дисковыми пластинами с нанесенным магнитным покрытием) и контроллером жесткого диска. Эта величина в разы медленнее. Например, для одного жесткого диска скорость обмена с внешним интерфейсом 300 Мб/с (External transfer rate) и internal ~174 Мб/с. Разумеется, можно объединить отдельные жесткие диски в единый массив, что позволит увеличить скорость обмена данными. Но тут вступает в действие другой фактор: взаимодействие даже с самым быстрым дисковым массивом ограничено пропускной способностью сети LAN или SAN, а также возможностями клиентского устройства (см. рис. 9.1). В конечном итоге становится очевидно: чтобы сравнить на практике быстродействие дисковых массивов и ленточных библиотек, нужно создать некие особые условия, при которых не возникает других узких мест при обмене данными. В реальной жизни этого достичь практически невозможно, и, честно говоря, просто нецелесообразно с экономической точки зрения.

### 9.3.2. Ограничения использования ленточных накопителей

Необходимость приобретения специальных считывающих устройств (накопителей и/или библиотек) – когда я встречаю подобные фразы в технических обзорах, меня не покидает мысль: «А что, дисковые накопители не являются специальными устройствами, которые надо приобретать за деньги?» Вероятнее всего, предполагается, что дис-

ковый массив можно приспособить для других целей, а ленточную библиотеку – нет. Например, при нехватке места появляется соблазн «пустить под нож» резервные копии, а из самого дискового массива сделать очередной файлообменник для хранения личных файлов сотрудников.



Рис. 9.1. Потенциально узкие места системы резервного копирования

Увы! Такой подход действительно применяется на многих российских предприятиях. Там, где процветает наплевательское отношение к ИТ в целом. Именно поэтому «неуниверсальность» ленточных систем хранения может послужить гарантией от безалаберных действий некоторых сотрудников ИТ-подразделений, а также их руководителей.

Необходимость создания системы учета и хранения. Действительно, при хранении информации на внешних носителях неплохо бы при этом знать, какая информация на каком носителе записана. (Ну, конечно, если вы не хотите раз за разом перечитывать все кассеты из хранилища.) Для этого существует специальная система учета на основе базы данных. Чаще всего эти функции уже встроены в ПО для резервного копирования, но иногда предпринимались попытки

создать подобную систему самостоятельно. В принципе, для подобных целей подойдет любая программа для складского учета. Особенно такая система актуальна при регулярном перемещении накопителей за пределы ИТ-структуры, например в банковскую ячейку. Постоянно вставлять накопители, для того чтобы посмотреть, какие данные на них записаны, в таких условиях не получится. В то же время подобный подход не мешает и при работе с дисковыми хранилищами, только вместо идентификатора будет использоваться имя архивного файла. Такой подход позволяет сократить время восстановления и избежать случайного удаления резервных копий при очередной «чистке» из-за дефицита свободного пространства для новых копий.



Мне всегда были интересен этот момент – вот некий оператор системы резервного копирования видит набор безликих файлов на дисковом хранилище. Откуда он при этом узнает, какие из них действительно необходимы, а какие можно удалить. Если у него нет системы учета – то никак. Получается, что вести учет надо при любой системе хранения.

Пожалуй, единственным серьезным «осложнением» можно считать наличие дополнительных механических устройств, осложняющих ремонт и обслуживание. Да, подобная проблема действительно существует. Наличие специальных механизмов автоподачи, извлечения картриджей из ячеек и механическая сущность ленточных накопителей требуют особых навыков и наличия запчастей для ремонта или замены. К сожалению, во всем мире неуклонно падает количество людей, умеющих работать руками, а не перекладывать бумажки с места на место. Поэтому найти грамотного ремонтника, способного починить ленточную библиотеку, очень и очень трудно.

Тут стоит отметить одну важную деталь. Пока устройство стоит на обслуживании у вендора – все вопросы решаются довольно оперативно. Если представители вендора отказываются от сотрудничества в данном направлении, например чтобы подтолкнуть клиента к покупке новой библиотеки, – вот тут при выходе из строя библиотеки начинаются проблемы с ремонтом. В принципе, если своевременно обновлять парк оборудования, в том числе и ленточную библиотеку, таких проблем не возникает.

Если быть справедливым, то следует помнить, что при использовании дисковых массивов могут возникнуть те же проблемы. При смене поколений дисковых накопителей организации вынуждены покупать новые дисковые массивы, потому что поставщики жестких дисков в погоне за прибылью перестают выпускать HDD прошлого модель-

ного ряда. В качестве такого негативного примера можно вспомнить переход от винчестеров UltraSCSI к стандартам SAS и SATA. Предприятия, не имевшие денег быстро выполнить переход, закупив за высокую стоимость новое оборудование, потом искали возможность за любые деньги приобрести HDD стандарта UltraSCSI, чтобы хоть как-то восстановить работоспособность своей инфраструктуры.

Поэтому выражение «Скупой платит дважды» как нельзя лучше подходит ко всей ИТ-отрасли в целом, вне зависимости от того, какая система хранения используется.



Исходя из моего опыта, все ленточное оборудование, нуждавшееся в восстановлении, пришло в нерабочее состояние из-за неаккуратной перевозки, эксплуатации в ненадлежащих условиях, неквалифицированных действий персонала и т. д. Стоит сказать, что встречаемые мной старые ленточные устройства, например библиотеки LTO-2 Adic от Quantum, работают до сих пор без какого-либо ремонта.

Подытожив все вышесказанное, можно сказать, что ленточные библиотеки хорошо подходят как для архивного хранения, так и для быстрого восстановления (Disaster Recovery) на новой площадке, например после пожара или другой катастрофы.

## 9.4. Диски и ленты работают вместе – технология D2D2T, или «Disk-to-Disk-to-Tape»

«Disk-to-Disk-to-Tape» (D2D2T) – это технология резервного копирования, когда данные сохраняются на диск, прежде чем они будут записаны на магнитной ленте.

D2D2T работает, когда данные пишутся на промежуточный диск, который имеет аналогичную или большую емкость (см. рис. 9.2).

Как правило, промежуточный диск хранит резервную копию в течение короткого срока: несколько недель или даже дней, при этом данные могут переноситься на ленту постепенно. Это позволяет процессу резервного копирования уложиться в «окно бэкапа», перенеся относительно медленное взаимодействие с ленточной библиотекой «на потом». При этом в случае сбоя ленточного носителя все равно остается резервная копия на временном дисковом хранилище, что позволяет повысить уровень отказоустойчивости.

В случае быстрого восстановления промежуточный диск обеспечивает более высокую скорость передачи данных (DTR), по сравне-

нию с устройством хранения на ленте. Это позволяет значительно ускорить процесс восстановления данных из самой последней копии.



Рис. 9.2. Технология «Disk-to-Disk-to-Tape» – D2D2T

## 9.5. Виртуальные ленточные хранилища – Virtual Tape Library

Технологический бум порой рождает странные вещи. Например, многие вендоры предлагают дисковые массивы, с логической точки зрения представляющие собой ленточные накопители. Речь идет о так называемых Virtual Tape Library VTL, что дословно означает виртуальные ленточные библиотеки. На первый взгляд, подобные накопители созданы с весьма благой целью: ускорить работу системы резервного копирования, чтобы легче попадать в «окно бэкапа». Вторая задача – избавить «бедного замученного системного администратора» от «ненавистных лент», при этом сохранив остальные компоненты системы резервного копирования, включая программное обеспечение и ее настройки.

На самом деле возможность эмуляции ленточного накопителя при помощи специального файла-образа существовала давным-давно – едва ли не в самых ранних версиях известных программ резерв-

ного копирования. Что же касается других особенностей ленточных хранилищ, например ведения учета, то вести его так или иначе все равно придется, чтобы обеспечить целостность и сохранность данных. В качестве дополнительного преимущества указывается Tiering, проще говоря, механизм перемещения данных, например с быстрого и дорогого хранилища на более медленное и дешевое. Но и здесь мало что нового – сейчас практически во всех современных системах резервного копирования присутствует технология D2D2T, или «disk-to-disk-to-tape», выполняющая практически то же самое, только без покупки лишнего оборудования.

Фактически концепция VTL Virtual Tape Library представляет собой очередной маркетинговый ход, в котором эксплуатируется модное слово «Virtual», и нежелание производителей оборудования поддерживать выпуск любых устройств с механикой чуть сложнее вентилятора.

## 9.6. Заключение

В правильно сбалансированной системе не возникает открытого противостояния. Поэтому для достижения каждой цели нужно использовать наиболее подходящий инструмент. В правильно организованной системе резервного копирования могут применяться различные механизмы: дисковые хранилища и ленточные библиотеки, чтобы выполнить и архивное копирование, и резервирование для Disaster Recovery, тем самым обеспечивая максимальную защиту данных.

# Глава 10

## Магнитные ленты и вездесущий LTO

*– Как передать поток Интернета около 10 Мбит на 1 км без проводов с наименьшими затратами? Тип аппаратуры?  
– Записать на самый дешевый съемный носитель и на велосипеде отвезти.*

(Из форумной переписки)

### 10.1. Предисловие

В этой главе пойдет речь о самых старых и заслуженных носителях информации в электронной форме.

«Как же так?!» – спросит современный читатель: «Ведь до магнитных лент в компьютерной индустрии использовались перфокарты и перфоленты!..» Дело в том, что до цифровых устройств обработки и хранения информации, к которым относятся компьютеры, были еще и аналоговые. Звук, записываемый на ленту, – это тоже информация. В научных исследованиях магнитофоны применялись не только для записи звука, но и с целью записи показаний аналоговых датчиков для фиксации результатов экспериментов. Магнитная лента (или специальная магнитная проволока) применялась в авиации, энергетике, оборонной промышленности и множестве других отраслей.

Прочтя главу 9, вы уже поняли, что за долгое время использования магнитных лент накопилось немало как их преданных сторонников, так и убежденных противников. Устройства на магнитной ленте используются до сих пор, и постоянно растущие «аппетиты» к объемам данных пророчат им еще долгую жизнь.

Сейчас маловероятно найти устройство для работы с магнитной лентой в ее классическом представлении – «на катушке». Место обычной ленты заняли сложные носители с высокоточной механи-

кой – ленточные картриджи (кассеты). Учитывая тот факт, что в такие носители встраивают специальный чип, который берет на себя служебные функции при работе с магнитной лентой, современные ленточные картриджи по своей сложной структуре и объемам хранения приближаются к жестким дискам.

## 10.2. Различные устройства для записи на ленту

Устройства для записи на ленту (или ленточные накопители) условно можно разделить на три основные разновидности: одиночные устройства – стримеры, накопители с автозагрузчиком («лоадеры», по-англ. *loaders*, от глагола *to load* – загружать) и ленточные библиотеки.

### 10.2.1. Одиночный ленточный накопитель

Стример (Streamer), или ленточный накопитель, по сути, представляет из себя специализированный магнитофон для записи данных на ленту. По традиции их относят к устройствам с последовательным доступом к данным. Часто эти накопители устанавливаются внутри сервера, например в 5-дюймовый отсек. Есть также модели, выполненные во внешнем форм-факторе, например в корпусе 1U или в настольном исполнении, чем-то напоминающем видеомагнитофон. Могут подключаться как к сугубо внешним интерфейсам, например USB, так и к специализированным адаптерам: SCSI, SAS или даже напрямую к сети SAN по шине Fiber Channel.

Главное отличие от других устройств – отсутствие механизма автоматической замены картриджа. Все функции контроля по загрузке, удалению и перезаписи носителей администратор резервного копирования выполняет вручную.



Такой, казалось бы, недостаток иногда может дать определенные преимущества. Если в программе резервного копирования включить функцию автоматического извлечения носителя, то замену картриджей можно поручить любому сотруднику, даже не входящему в ИТ-подразделение, например начальнику службы безопасности.

Существует довольно интересное решение – массив независимых накопителей из нескольких приводов лент в одном корпусе, причем каждый из них обслуживает один-единственный носитель. Схема работы таких массивов аналогична системам RAID для жестких дисков. По аналогии с RAID Redundant Array of Independent Disks – избыточ-

ный массив независимых дисков, такие объединения ленточных накопителей называют RAIT (Redundant Array of Independent Tapes – избыточный массив независимых лент). Использование таких решений значительно повышает производительность операций передачи данных, поскольку накопители работают параллельно. Кроме того, RAIT обеспечивает повышенную отказоустойчивость, так как он спроектирован аналогично избыточным схемам RAID. Основные недостатки ленточных массивов связаны с невысокой емкостью и невозможностью ротации носителей – при чтении данных каждый носитель может быть использован в совокупности со своими «собратьями» по массиву, и обязательно носитель должен быть помещен в «свой» накопитель, где он был записан. Технологию RAIT можно реализовать не только на базе одиночных накопителей устройств, но и приводов, встроженных в ленточные библиотеки.

### 10.2.2. Автозагрузчик

В принципе, это тот же самый одиночный накопитель, но снабженный механизмом замены картриджа в накопителе. Чаще всего автозагрузчики выполняются в виде устройств 1U для серверных стоек и содержат 2 контейнера (пенала) для размещения ленточных носителей: левый и правый. Каждый такой пенал обычно размещает 4 ячейки для магнитных лент. Итого 8 магнитных носителей. Операции производятся только с одним накопителем в один момент времени, то есть пока ленточный накопитель читает или записывает один носитель, остальные 7 ждут своей очереди. Встроенный механизм автоподачи используется не только для изъятия или загрузки картриджей в накопитель, но и для перемещения ленточных носителей между ячейками.

Иногда самую крайнюю ячейку выделяют исключительно для загрузки новых или изъятия уже записанных ленточных накопителей. Такую «освобожденную» ячейку называют mail-slot. Это помогает упростить операции замены кассет без выемки пеналов. Но при этом остается только семь ячеек, так как восьмая используется в качестве mail-slot. Функции перемещения кассет в накопителе целиком ложатся на механику лодаера. Этот вариант считается более безопасным, так как снижается риск уронить картридж и повредить его механизм, уязвимый к разрушительному воздействию (см. рис. 10.1).

Подключаются загрузчики чаще всего либо через интерфейс SAS (SCSI), либо посредством оптоволоконного кабеля, либо к HBA-адаптеру сервера, либо напрямую к сети хранения данных (SAN) через Fiber Channel.

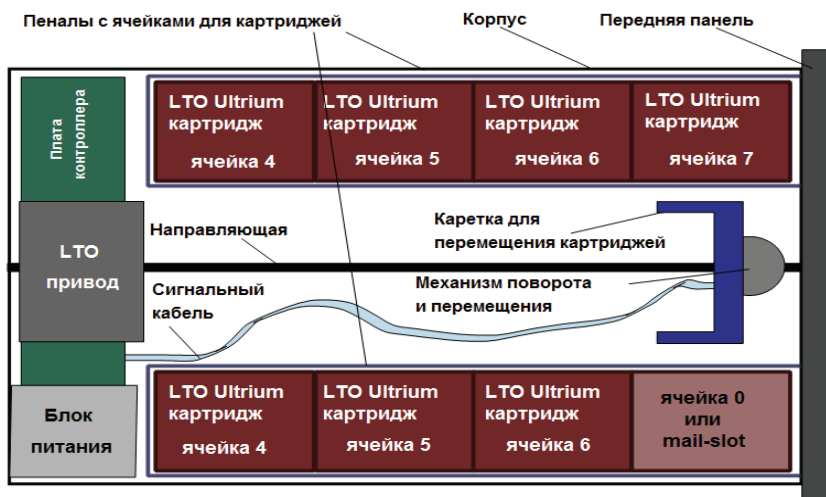


Рис. 10.1. Внутреннее устройство автозагрузчика на 8 кассет

### 10.2.3. Ленточная библиотека

По сравнению с отдельным автозагрузчиком, ленточная библиотека представляет устройство с несколькими накопителями и увеличенным количеством ячеек для размещения лент. По сути, это своего рода хранилище с функцией чтения-записи носителей. Из-за одновременной работы нескольких приводов скорость обмена информацией значительно повышается. Загрузка носителей внутрь может быть реализована несколькими вариантами. Это может быть специализированный mail-slot на несколько кассет или возможность открытия целиком передней панели для помещения накопителей непосредственно в ячейки библиотеки. Схемы обработки данных на накопителях тоже могут отличаться. Например, любой носитель может быть отработан в любом приводе, или же за накопителем закрепляется определенная часть общего хранилища носителей. Некоторые библиотеки организованы по модульному принципу и могут быть объединены в одно общее устройство.

### 10.2.4. Рекомендации по применению тех или иных устройств

Библиотеки больше подходят для крупных центров обработки данных (ЦОД). Основные причины для этого – довольно высокая ско-

рость копирования и восстановления информации (до сотен гигабайт в час), большая емкость (до десятков терабайт), надежность хранения и минимальная удельная стоимость хранения (в расчете на один мегабайт данных).

В то же время одиночные приводы и лoadеры больше подходят для решений среднего и малого бизнеса, а также для ИТ-инфраструктур с распределенным хранением данных.

Некоторые системные администраторы используют ленточные библиотеки и даже отдельные накопители как своего рода постоянные хранилища информации. Но не следует упускать из виду вероятность форс-мажорных обстоятельств и время от времени выносить данные в безопасное место за территорией предприятия или ЦОД. А ленточные накопители как нельзя лучше подходят для этих целей.

### 10.3. Стандарты хранения на ленточных накопителях. Вездесущий LTO

При работе с архивными данными очень важно получить ответ на вопрос: «А смогу ли я прочитать данные с ленты, спустя, скажем, 5 лет после выпуска устройства? И будут ли выпускаться устройства, поддерживающие данный стандарт?»

Благодаря популярности ленточных накопителей было выпущено множество модификаций различных моделей. При этом потребитель столкнулся с проблемой: ленточные картриджи от одного производителя могли быть прочитаны только в накопителях определенной модели. Если производитель по какой-то причине прекращал выпускать нужное оборудование, весь накопленный за многие годы архив данных превращался в свалку высокотехнологичного мусора.

Для ликвидации вышеописанных проблем крупные производители договариваются об используемых стандартах, и Linear Tape-Open, или сокращенно LTO, явился самым успешным воплощением идеи стандартизации.

Он появился как результат консолидации усилий нескольких крупных вендоров: IBM, Hewlett-Packard и Seagate Technology. Позже ленточное подразделение Seagate Technology купила компания Quantum, которая также внесла мощный вклад в развитие рынка устройств записи на магнитной ленте.

Вначале были разработаны два формата – Ultrium и Accelis. Формат Accelis изначально предназначался для быстрого чтения данных,

что само по себе неплохо при хранении больших объемов данных.

В то же время формат Ultrium позволял организовать более быструю запись данных, что дает преимущества для резервного копирования в условиях узкого «окна бэкапа».

Но так как целью разработки стандарта была полная унификация хранимых данных, «в живых должен был остаться только один», и использование формата Accelis было прекращено.

Поэтому все устройства и носители, соответствующие данному стандарту, обозначают как «LTO Ultrium» или даже просто «Ultrium».

Одни из самых известных особенностей LTO: поддержка большого количества параллельных каналов на ленте и пленка полудюймовой ширины. Своего рода «визитная карточка стандарта».

Помимо основной своей роли – специализированных систем хранения, устройства LTO Ultrium легко встраиваются в серверное и другое оборудование и поддерживают различные интерфейсы: SCSI, SAS, FC. На уровне операционной системы они совместимы с большинством известных платформ: семейством Microsoft Windows, семейством HP-UX, IBM-AIX, различными дистрибутивами Linux, Solaris и BSD.

Еще один важный момент: все устройства LTO поддерживают технологию WORM – однократной записи и множественного чтения. Это может быть очень полезно, чтобы предотвратить возможность уничтожения ценной информации злоумышленником на этапе архивного хранения.

### 10.3.1. Характеристики различных поколений LTO

Прежде чем начать сравнительное описание различных поколений, хотелось бы сказать несколько слов о емкости накопителей. В рекламных целях при описании объема принято удваивать реальную емкость носителя. Например, если накопитель LTO1 может вместить максимум 100 Гб данных без учета сжатия, производители и продавцы указывают «максимальную емкость» в 200 Гб. Это при условии, что конечному потребителю очень повезет, и его данные могут быть сжаты в два раза. Если же необходимо сохранять данные, которые уже прошли процедуру сжатия, например графические файлы в формате JPEG, то емкость хранимых файлов так и составит в лучшем случае 100 Гб. Аналогично обстоит дело с зашифрованными данными, которые не сжимаются в принципе. Поэтому нужно быть очень осторожным, оценивая соответствие носителя LTO объему сохраняемой информации. Часто можно встретить запись типа 100/200 Гб, обозна-

чающую сырую емкость накопителя и емкость при двукратном сжатии соответственно.

Примерно так же обстоит дело и с параметрами скорости передачи данных. Реальная скорость данных всегда раза в два ниже, чем «максимальная, с учетом сжатия». Учитывайте этот факт, чтобы поместиться в «окно бэкапа».

И не стоит забывать о накладных расходах. Опытные администраторы резервного копирования ориентируются на показатель, что на хранение 100 единиц информации на ленточных накопителях требуется примерно 110–120 единиц объема ленты без учета компрессии. Подобное несоответствие связано с ожидаемыми потерями на запись служебной информации. Наиболее хитрые, подозрительные и злопамятные администраторы закладывают полуторакратное превышение объема, то есть на каждые рабочие 100 Мб информации выделяется 150 Мб на ленте без сжатия. И, как правило, такая избыточность в итоге оправдывается.

Еще один часто встречающийся вопрос – об устройствах «полной» и «половинной» высоты. Устройство полной высоты равно примерно двум пятидюймовым отсекам (примерно, потому что каждый производитель склонен адаптировать свои устройства для применения в первую очередь в составе своего же оборудования в сборе). Данный тип устройств в основном предназначался для установки в крупные серверы и ленточные библиотеки. Сейчас в основном выпускаются устройства половинной высоты, то есть в два раза меньше по габаритам и позволяющие разместить в старых библиотеках и серверах больше приводов.



При закупке устройств хранения лучше выбирать оборудование одного производителя.

Для стандарта LTO существует очень простая формула: накопитель последней модели может читать и писать данные на носителях предпоследнего поколения, а с накопителями более раннего поколения («предпредпоследнего») он работает в режиме «только чтение».

**Пример:**

накопитель LTO 5 может работать с носителями LTO4 и LTO5 в режиме «чтение-запись», с кассет LTO3 он может только прочесть информацию.

Соответственно, LTO4 поддерживает режим R/W с картриджами LTO4 и LTO3, а с LTO2 – только RO. И т. д.

По замыслу разработчиков и производителей, из-за того, что объемы данных постоянно растут, нет смысла обеспечивать поддержку

совсем ранних моделей. Это позволяет удешевить оборудование. Еще одна причина – магнитная лента не позволяет хранить данные вечно. Несмотря на то что производители заявляют очень оптимистичный прогноз – сохранность информации в течение 10 лет, опыт показывает, что лучше на это не рассчитывать. На протяжении от двух до пяти лет желательно перенести информацию на новый носитель, а старый – уничтожить в целях безопасности. Поэтому нет никакого смысла обеспечивать возможность работы с очень старыми лентами, тем более что-то писать на них.

Если у вас вышла из строя старая ленточная библиотека, скорее всего, вы уже не найдете ни запчастей, ни специалистов для ее ремонта. Поэтому своевременно проводите модернизацию системы резервного копирования, не дожидаясь момента, когда старый аппаратно-программный комплекс перестанет справляться со своими задачами.



Производя модернизацию своей серверной или ЦОД, не выбрасывайте, а сохраните на складе хотя бы одно работающее устройство и хотя бы одну копию программного обеспечения, при помощи которых делались старые резервные копии. Дело в том, что некоторые топ-менеджеры или руководители служб безопасности иногда предпочитают создавать свои собственные архивы. И рано или поздно возникает необходимость прочесть с них информацию или хотя бы выдать заключение о состоянии резервной копии.

### 10.3.2. Сравнительное описание поколений LTO

#### *LTO-1 (Generation-1) (100/200 Гб)*

Сейчас накопители этого поколения вероятнее всего встретить только в «частных коллекциях». Они появились в конце 2000 года, имели емкость 100 без сжатия и 200 Гб, большинство устройств было оснащено только интерфейсом SCSI LVD/SE.

#### *LTO-2 (Generation-2) (200/400 Гб)*

Стримеры первого поколения LTO-2 Ultrium 460 полной высоты (30/60 Мб/сек, 200/400 Гб), оснащены интерфейсом Wide Ultra 160 SCSI LVD/SE, FC. Также известна вторая модификация стандарта для стримеров LTO-2 Ultrium 448 с кэшем в 64 Мб, скоростью записи-чтения 24/48 Мб/сек, 200/400 Об. Устройства имели интерфейсы Wide Ultra 160 SCSI LVD/SE, а также 3 Об/с SAS.

#### *LTO-3 (Generation-3) (400/800 Гб)*

Стримеры первого поколения LTO-3 Ultrium 960, кэш 128 Мб полной высоты (160/320 Мб/сек, 400/800 Гб), оснащены интерфейсом

SCSI LVD/SE, FC, а также у фирмы Quantum существует вариант LTO-3A Drive Gigabit Ethernet (NAS). Второе поколение стримеров LTO-3 Ultrium 920, кэш 64 Мб (160/320 Мб/сек, 400/800 Гб), оснащено интерфейсом Wide Ultra 320 SCSI LVD/SE, 80-контактным SCA (SCSI и питание), 3 Гб/с SAS или подключались по интерфейсам Fibre Channel.

### ***LTO-4 (Generation-4) (800/1600 Гб)***

Стримеры первого поколения LTO-4 Ultrium с кэшем 128 Мб полной высоты (160/320 Мб/с, 800/1600 Гб) оснащены как интерфейсом Ultra320 SCSI LVD/SE, FC, так и интерфейсом Гб/с SAS.

Второе поколение стримеров LTO-4 половинной высоты (160/320 Мб/с, 800/1600 Гб) пока оснащено только интерфейсом 3Гб/с SAS.

Это поколение ознаменовало старт встроенной поддержки шифрования. В стримеры LTO-4 уже встроено кодирование данных на базе аппаратных средств с использованием 256-битной системы AES.

### ***LTO-5 (Generation-5) (1500/3000 Гб)***

Устройства LTO 5 имеют емкость хранения данных 1500 Гб без сжатия и 3000 Гб в режиме компрессии данных и поддерживают скорость передачи до 140 Мбайт/с LTO 5) (со сжатием до 280 Мбайт/с). Это второе поколение LTO Ultrium, имеющее встроенную систему шифрования данных на аппаратном уровне.

Начиная с этого поколения возможна реализация системы хранения на базе специализированной файловой системы LTFS (подробнее об этом будет рассказано ниже).

### ***LTO-6 (Generation-6) (2,5/6,25 Тб)***

Поддерживающие данную технологию устройства могут сохранять 2,5 Тб данных без учета компрессии. Максимальный размер данных составляет 6,25 Тб. Такой рост стал возможен за счет улучшения алгоритма сжатия, поэтому усредненный коэффициент теперь берется равным 1 к 2,5. К сожалению, скорость передачи данных без компрессии сопоставима с поколением LTO-5.

Также поддерживается функция шифрования на основе алгоритма AES 256 бит.

Хочу еще раз напомнить об условности любых коэффициентов сжатия. При расчете требуемых объемов данных нужно всегда ориентироваться на физическую емкость носителей без учета сжатия.

## 10.4. Поддержка на уровне файловых систем

Компанией IBM была представлена файловая система Linear Tape File System (LTFS) для работы с ленточными носителями. Эта файловая система позволяет работать с картриджами LTO-5 и LTO-6 на внешних ленточных приводах как с блочным устройством, например с USB-флэшкой. Такая функция особенно полезна для крупных хранилищ данных. Разумеется, если в библиотеке или одиночном накопителе нет соответствующего картриджа, то прочесть ничего не получится, поэтому вопрос об инвентаризации и хранении данных остается открытым. В своей работе LTFS использует первые дорожки ленты для индекса файловой системы.

## 10.5. Загадочный чип LTO-CM

Каждый ленточный накопитель LTO при создании получает специальный встроенный чип Cartridge Memory (LTO-CM). В первом, втором и третьем поколениях он служил для доступа к 128 блокам памяти, по 32 байта каждый, то есть к 4096 байтам. В LTO-4 его емкость увеличена до 8192 байт. Данные могут считываться или записываться поблочно при помощи специального бесконтактного считывателя. Содержимое памяти используется для идентификации лент и для опознавания накопителем поколения LTO.

Существуют различные считыватели: внешние – для использования в составе ленточных библиотек – и автономные. Некоторые из них позволяют временно заблокировать средствами LTO-CM доступ к данным на картридже до ввода разрешающего доступ ключа, что может использоваться, например, на время перевозки картриджей с конфиденциальной информацией с одной доверенной территории на другую.

## 10.6. Дополнительно о шифровании данных на ленте

Часто очень важно обеспечить безопасность клиентских данных, особенно если они могут носить конфиденциальный характер. В принципе, резервное копирование уже само по себе является хорошим средством для кражи информации. Добавьте к этому необходимость переноса данных за периметр предприятия (off-site), и мы получаем

просто лакомый кусочек для злоумышленника. Но существует риск не только кражи, но и случайной утраты носителя, например по причине ДТП на дороге. Чтобы защититься от таких нежелательных случайностей, нужно использовать технологию шифрования ленточных устройств. Это обеспечивает простую эффективную защиту конфиденциальных данных и предотвращает несанкционированный доступ к хранилищу картриджей.

Технология шифрования данных использует более высокий уровень, для которого требуются 256-разрядные ключи шифрования стандарта AES (Advanced Encryption Standard). Ключи передаются на накопитель менеджером ключей, чтобы производить шифрование и расшифровку данных.

Для обеспечения шифрования можно использовать различные уровни и возможности аппаратно-программного обеспечения.

### ***Шифрование на уровне программы***

Практически любая современная программа резервного копирования позволяет использовать функцию шифрования данных. Например, CA BrightStor ARCserve, HP Data Protector и многие другие. Преимуществом данного метода является простота реализации, минусом – большее требование к ресурсам и снижение работы программы резервного копирования.

### ***Шифрование на уровне операционной системы***

Известный пример – шифрование в операционной системе AIX. Ключами шифрования, которые передаются накопителю, управляет драйвер устройства или операционная система, а хранятся они в менеджере ключей шифрования. Основное преимущество данного метода – большее быстроедействие при выполнении операций кодирования, ведь работа ведется на уровне операционной системы.

### ***Шифрование на уровне библиотеки***

В этом случае функциями шифрования управляет аппаратное обеспечение, например блок управления крупной ленточной библиотеки уровня Enterprise. Для работы программе резервного копирования (или операционной системе) достаточно только передать соответствующий ключ.

Несомненным преимуществом такого подхода является перенос функции кодирования с сервера резервного копирования на саму ленточную библиотеку.

Несомненным минусом, или ограничением, является тот факт, что данная функция поддерживается очень ограниченным количеством устройств от известных (и дорогих) вендоров, например IBM. Вероятнее всего, при восстановлении данных с нуля придется покупать точно такое же устройство, а спустя несколько лет это может быть весьма проблематично.

В принципе, не особенно важно, на каком этапе происходит шифрование данных. Гораздо важнее иметь возможность расшифровать записанные данные. Для этих целей необходимо обязательно иметь резервные копии конфигурации аппаратных и программных средств резервного копирования, чтобы при необходимости с легкостью воссоздать нужную инфраструктуру и восстановить данные, например после пожара. И уж конечно, к этим копиям не нужно применять тех же средств шифрования и хранения, которые используются для основных данных.

## 10.7. Заключение

За долгий путь развития устройства хранения данных на ленте претерпели колоссальное развитие и видоизменились практически до неузнаваемости. Последние поколения ленточных накопителей по скорости работы и объемам хранения практически сопоставимы с современными жесткими дисками. В то же время хранение данных на ленте имеет неоспоримые преимущества. Более подробно об этом, а также о сравнении их с жесткими дисками мы поговорим в следующей главе.

# Глава 11

## Дисковая система хранения на базе NAS

*Из журнала «Популярная механика» (США),  
1949 год: «В будущем компьютеры будут ве-  
сить не более полутора тонн...»*

### 11.1. Описание принципов работы NAS

Вначале разберем устройство сетевого хранилища. Традиционно NAS представляет собой усеченную версию компьютера на Intel- или ARM-процессоре. ARM-процессоры принято использовать в устройствах для домашнего использования, в корпоративном сегменте они практически не встречаются из-за малой вычислительной мощности.

Как уже упоминалось в главе 8, основное управление производится посредством сетевого интерфейса.

Основной поток данных «во внешний мир» также проходит через сетевую подсистему.

Пользовательская информация хранится на жестких дисках. Для обеспечения надежности рекомендуется объединять жесткие диски в отказоустойчивые массивы. Операционная система чаще всего устанавливается на выделенный носитель: флэш-накопитель или отдельный винчестер. С общими принципами внутреннего устройства можно познакомиться на рис. 11.1.

В принципе, работа с сетевым хранилищем не составляет особого труда, и большая часть настроек может быть выполнена даже начинающим системным администратором. В этом мы убедимся на примере реально работающего NAS.

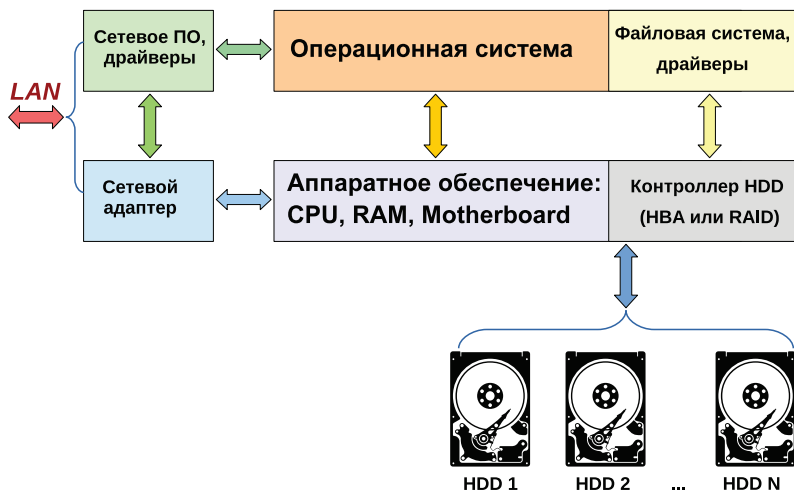


Рис. 11.1. Схема внутреннего устройства сетевого хранилища (NAS)

## 11.2. Работа с NAS на примере NETGEAR ReadyNAS Pro

### 11.2.1. О выборе RAID

Стоит отдельно сказать о технологии X-RAID, применяемой в хранилищах ReadyNAS.

X-RAID – это разработка, запатентованная фирмой Infrant. Данная технология организации RAID-массивов значительно упрощает процесс управления RAID. Вы можете установить в хранилище, к примеру, два диска по 1 Тб. В этом случае вы получаете «зеркало» из двух дисков, аналог RAID1. Потом, если понадобится дополнительное дисковое пространство, можно добавить еще один диск указанной емкости и получить размер массива, равный суммарному объему двух дисков, аналогично ситуации с RAID5. Точно таким же образом добавляются и другие диски, увеличивая тем самым объем используемого пространства, пока не заполнятся все отсеки. Но даже когда все отсеки заполнены, X-RAID позволяет увеличить объем и дальше. Если заменить все диски, то после замены последнего диска том автоматически расширится за счет дополнительной емкости новых дисков. Естественно, заменять диски нужно по одному, чтобы успешно выполнялась операция rebuilding по перестройке массива.



После каждой смены диска нужно обязательно дождаться окончания синхронизации. Это может занять продолжительное время (для винчестера в 1 Тб процедура может занять до 10 часов). Доступ к данным возможен без каких-либо ограничений.

Надо отметить, что большинство дисковых хранилищ, в том числе ReadyNAS, поддерживают и обычный RAID5. В случае замены обычного офисного файлового сервера этот вариант является даже предпочтительнее, так как X-RAID оптимизирован для операций чтения больших блоков данных. Такая оптимизация хороша, например, при воспроизведении потокового видео или восстановления системы при помощи программ блочного резервного копирования, например Acronis Backup & Restore, но теряет смысл при обычных операциях чтения-записи при работе с офисными приложениями.

При прочих равных условиях лучше сделать выбор в пользу технологии X-RAID. Например, ради оптимизации затрат и возможности в дальнейшем делегировать полномочия другому лицу. В этом случае появляется возможность наращивать объем хранилища постепенно, не прибегая к услугам высококвалифицированного системного администратора. Кроме того, система X-RAID имеет возможность в автоматическом режиме перестроить RAID-массив при замене вышедшего из строя жесткого диска.

## 11.2.2. Первоначальная настройка

Установка дисков в хранилище производится обычным путем. Диск устанавливается в корзину, корзина с диском вставляется в хранилище. Здесь, как говорится, ничего нового не придумано (см. рис. 11.2).

После первом включении хранилища сразу появляются надпись **Ready NAS** и полоска индикатора. Следом идет сообщение: **Factory Reset installing**. После этого через несколько минут на индикаторе должно появиться: **Create disk C:**, – с указанием процентов от выполненной операции. Далее система выдаст: **Requesting IP**. По умолчанию дисковые хранилища ReadyNAS настроены на получение IP-адреса по DHCP. Далее следует синхронизация дисков.

Самое время приступать к установке программы для настройки дискового хранилища, не дожидаясь конца этого процесса.

Нужно отметить следующую ситуацию: Linux-версия программы RAIDar, предназначенная для поиска сетевых хранилищ и их настройки, работает далеко не на всех дистрибутивах. Для таких случаев на прилагаемом CD присутствует документация, где есть раздел, связанный с работой на Linux. В подобных случаях нужен

веб-интерфейс программы: [http://\\_адрес\\_хранилища\\_/admin/](http://_адрес_хранилища_/admin/). (Веб-интерфейс NETGEAR ReadyNAS называется «FrontView».) Логин для входа по умолчанию *admin*, пароль по умолчанию *netgear1*.



Рис. 11.2. Сетевое хранилище NETGEAR ReadyNAS Pro с наполовину извлеченной корзиной



Веб-интерфейс NETGEAR ReadyNAS Pro имеет 2 режима: режим мастера настройки (Wizard) и основной. Первый раз автоматически предлагается Wizard. Но пользователь в любой момент может переключиться в основной режим по кнопке **Advanced Control**. После первоначальной настройки при каждом новом сеансе будет автоматически предложен основной режим, но точно так же можно легко переключиться в режим мастера настройки, нажав кнопку **Setup Wizard**.

Чтобы не утомлять читателя излишними подробностями, постараюсь описать процесс первичной настройки как можно короче.

Итак, при входе запускается мастер настройки (**Setup Wizard**), который выдаст первое окно **System**, вкладку **Clock**, где предлагается установить системное время.

Далее следуют следующие окна и вкладки:

- **Alerts** – где предлагается ввести адрес E-mail для приема уведомлений;

- окно **Network** – в котором можно либо задать получение IP-адреса по DHCP – **Use values from DHCP server**, либо выбрать пункт **Use values below** и вручную установить IP-адрес и маску подсети. У хранилища два сетевых интерфейса, такая возможность появится дважды: для первого и второго сетевых подключений;
- **Global Settings** – здесь задаются такие важные параметры, как имя хоста (**Hostname**), название рабочей группы (**Workgroups**), а также указываются шлюз по умолчанию (**Default Gateway**) и настройка DNS (**DNS Setting**) (см. рис. 11.3). Последние два параметра доступны для ввода только в случае, когда сетевые настройки устанавливаются вручную, а не присваиваются посредством DHCP;

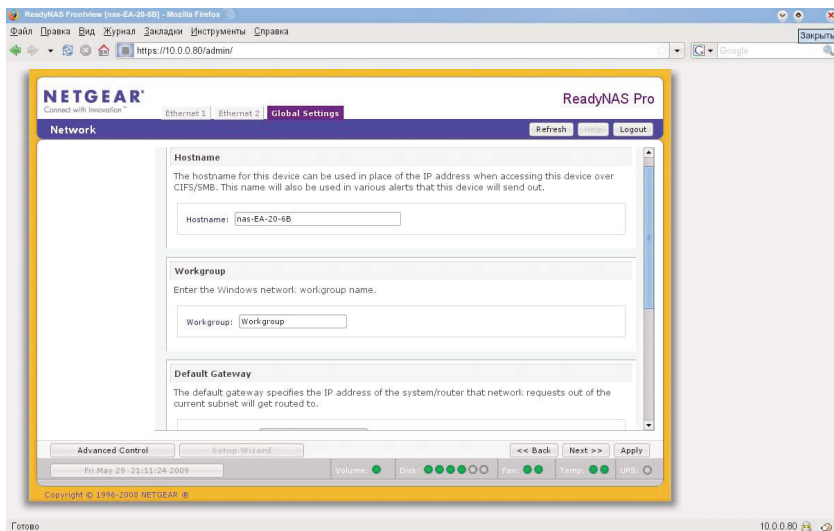


Рис. 11.3. Окно настройки **Network**, вкладка **Global Setting**

- **Security** – где можно сменить пароль администратора, а также задать реквизиты для восстановления пароля (см. рис. 11.4). В случае, если реквизиты восстановления пароля заданы корректно, для выполнения данной операции достаточно зайти на страничку [http://ip\\_address\\_of\\_readynas/password\\_recovery](http://ip_address_of_readynas/password_recovery). Если данные были введены неправильно или данная функция не была включена, то процедура восстановления несколько

усложняется – потребуется переустановка ПО, которая сбросит пароль, но сохранит основные настройки;

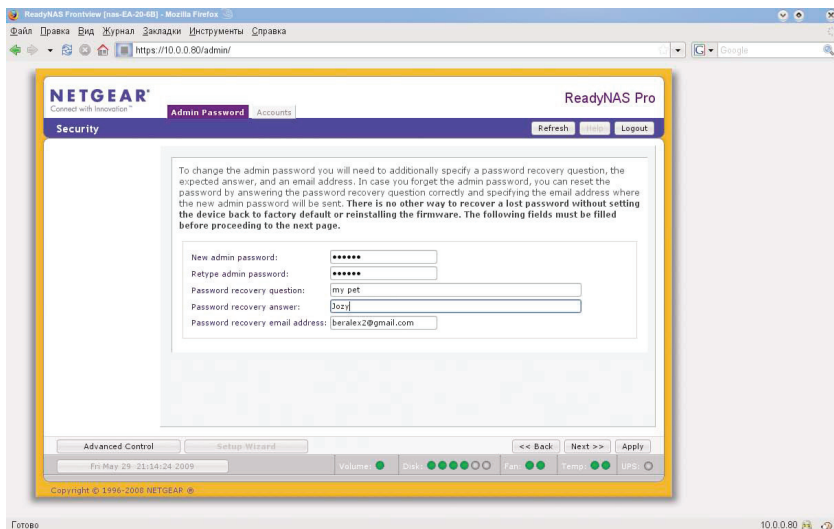


Рис. 11.4. Окно **Security**, вкладка **Admin Password**

- **Accounts** – здесь можно создать учетные записи остальных пользователей;
- окно **Services**:
  - **Standard File Protocols** – представлен перечень протоколов, по которым будет осуществляться доступ к хранилищу. Поддержку неиспользуемых протоколов можно отключить, сняв соответствующие галочки (см. рис. 11.5);
  - **Streaming Services** – для мультимедийных приложений и в данном случае абсолютно неактуальна;
  - **Installed Add-Ons** – показывает установленные дополнения. Так как устройство только что запущено и дополнений нет, список пуст;
- окно **Shares**:
  - **Share List** – список общих ресурсов и возможность изменить параметры, например по каким протоколам будет предоставляться доступ. Следует также добавить, что ReadyNAS позволяет предоставлять в общий доступ не только внутренние диски, но и сменные носители, подключенные по USB (см. рис. 11.6);
  - **Add Shares** – позволяет создавать новые общие ресурсы.

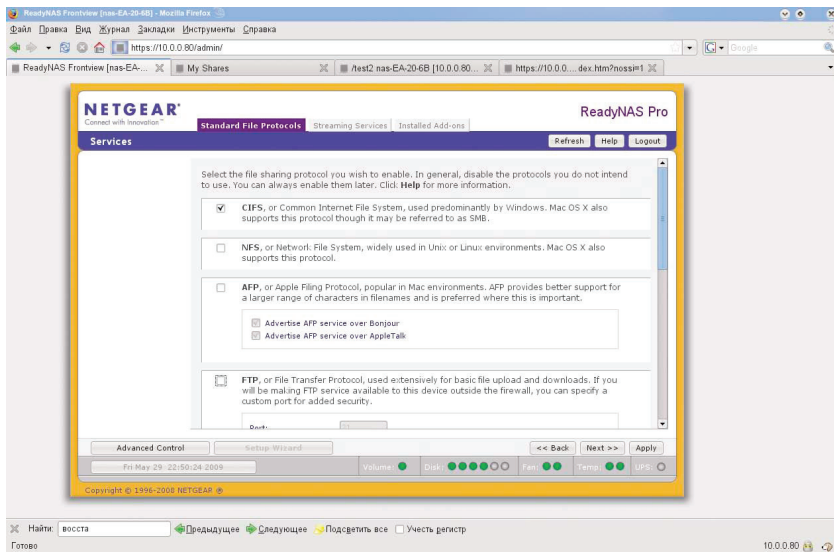


Рис. 11.5. Окно **Services**, вкладка **Standard Files Protocols**.  
Отключаем все, кроме CIFS и HTTPS, по умолчанию

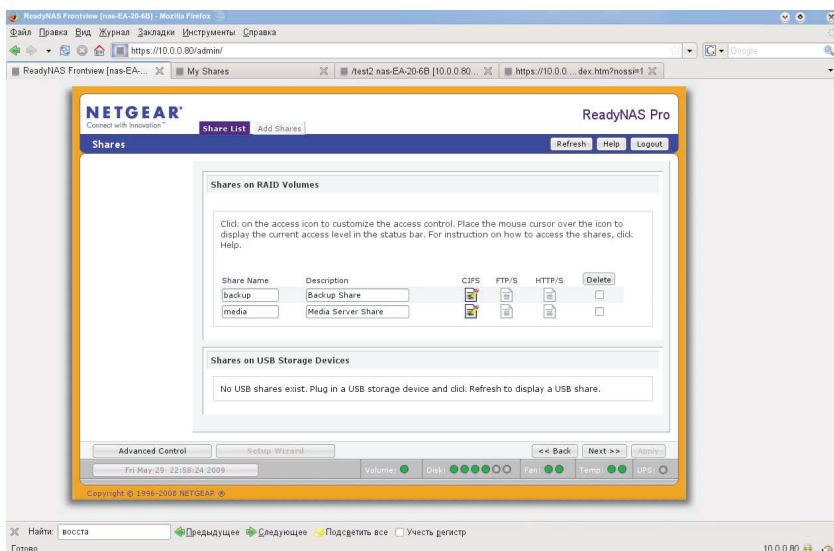


Рис. 11.6. Окно **Share**, вкладка **Share list** – список ресурсов

В итоге появляется окно **Registration**, где нужно зарегистрировать сетевое хранилище для получения поддержки со стороны производителя.

Что же касается Windows-платформ, для этой среды существует программа RAIDar. Просто необходимо запустить инсталляционный файл (на тот момент это был RAIDar\_Win\_4\_1\_5.exe) и по итогам установки получить рабочее окно программы. Данное приложение не только помогает осуществить поиск сетевых хранилищ ReadyNAS в сети, но и облегчает доступ к административному веб-интерфейсу соответствующих устройств (хранилищ), а кроме того, служит для мониторинга их состояния (см. рис. 11.7). После выбора соответствующего хранилища и нажатия на кнопку **Setup** запустится уже знакомый нам веб-интерфейс программы настройки хранилища.

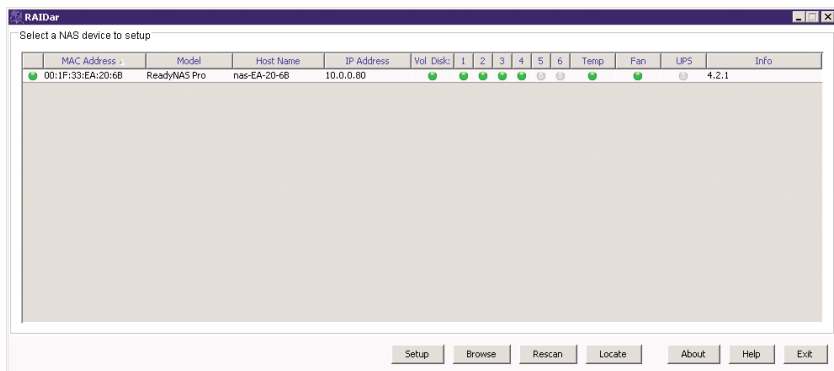


Рис. 11.7. Запущенная программа RAIDar

### 11.2.3. Учетные записи пользователей

При работе с мастером (Wizard) настройки мы произвели только первичную настройку сетевого хранилища. Но нам необходимо создать из этого агрегата полноценный файловый сервер, завести несколько пользователей и предоставить в общий доступ несколько каталогов.

Прежде чем приступить к настройке прав, необходимо понять идеологию работы сетевого хранилища. Встроенная операционная система NETGEAR ReadyNAS Pro базируется на ядре Linux. Обычно схема распределения прав доступа наследует принятую в UNIX-подобных системах модель: *владелец – группа – все (owner – group – public)*. Например, может быть использована такая схема работы:

владелец каталога имеет право читать и писать в него, прикрепленная группа пользователей – только читать, а все остальные вообще не имеют доступа. Или такая: владелец и ассоциированная группа могут и читать, и писать, а все остальные – только просматривать. Модель *владелец – группа – все* может создавать некоторые неудобства при использовании NAS в качестве офисного файл-сервера, но для хранения резервных копий это не особо важно.

Для примера создадим учетную запись пользователя, который обладает правами на чтение-запись, и дадим ему соответствующее имя «readwriter».

Для этого в браузере переходим на страницу управления [http://\\_адрес\\_хранилища/\\_admin/](http://_адрес_хранилища/_admin/), вводим имя пользователя **admin** и соответствующий пароль, в появившемся окне с левой стороны выбираем пункт меню **Security**, далее подпункт **User & Group Account**. Переходим на вкладку **Add User** и заполняем поля, необходимые для создания новой учетной записи:

- **User** – имя пользователя;
- **E-mail** – почтовый адрес, используется для отправки уведомлений, например о заполнении выделенного дискового пространства;
- **UID** – уникальный идентификатор пользователя. Данное поле лучше оставить пустым, программа сама проставит нужное значение;
- **Primary Group** – первичная группа пользователя. В нашем примере оставлено значение по умолчанию;
- **Password** – пароль пользователя;
- **Quota** – выделенная квота для данного пользователя.

Похожим способом создается пользователь «reader», который в дальнейшем будет иметь права только для чтения (см. рис. 11.8).

После заполнения реквизитов учетной записи пользователя нажимаем кнопку **Apply**, и новая учетная запись будет создана.

Аналогичным образом создается соответствующая запись для группы. Для создания группы необходимо посредством ниспадающего меню в первом верхнем углу окна выбрать соответствующий раздел.

Далее в появившемся окне следует заполнить следующие поля:

- **Group Name** – имя группы;
- **GID** – уникальный идентификационный номер группы. Лучше оставить пустым, тогда система присвоит ему необходимое значение;
- **Quota MB** – квота для группы.

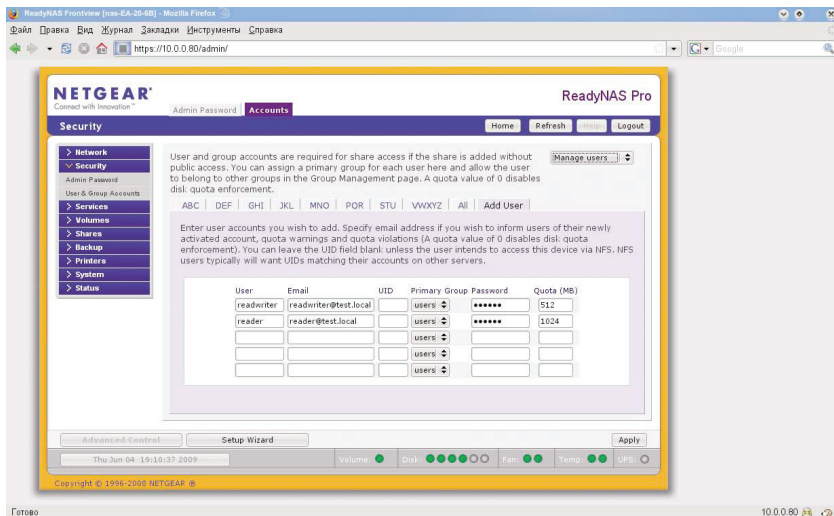


Рис. 11.8. Создание нового пользователя на сетевом хранилище

Когда учетная запись для группы создана, требуется приписать к данной группе необходимых пользователей. Для этого, ориентируясь по алфавитному перечню, перейдем на вкладку настройки свойств данной группы и введем имена пользователей в поле **Secondary Members**.

Теперь создадим общий ресурс. Сам процесс создания проходит довольно просто. Переходим в раздел **Shares**, добавляем раздел **Add Share**. Теперь необходимо заполнить форму, в которой указать имя ресурса и примечание к нему. После нажатия кнопки **Apply** ресурс будет создан.

#### 11.2.4. Настройка доступа по протоколу CIFS

Для настройки доступа клиентов сети Windows по протоколу CIFS переходим в раздел **Shares – Shares List**. Кликаем мышкой по значку протокола CIFS созданного нами общего ресурса и попадаем в окно настройки доступа по CIFS. Данное окно хоть и небольшое, но содержит весьма внушительное число параметров и снабжено полосой прокрутки (см. рис. 11.9).

В целях безопасности определим доступ по умолчанию (**«Default Access»**) к нашему ресурсу как «Disabled», то есть доступа к нему без аутентификации не будет. А пользователей и группы, которым необходимо иметь доступ на чтение или чтение-запись, укажем отдельно, заполнив поля **Read-only users**, **Read-only groups**, **Write-**

**enabled users**, **Write-enabled groups** соответственно. Следует также упомянуть о поле **Hosts allowed access**, в котором задается список IP-адресов или имен хостов для разрешения доступа к ресурсу только этим клиентам. Если же данное поле не активировано, доступ осуществляется с любого адреса.

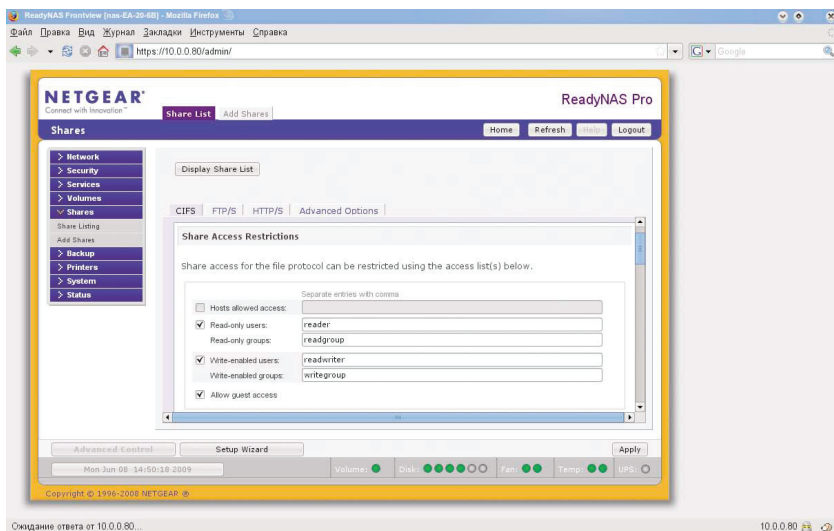


Рис. 11.9. Окно настройки доступа к общему ресурсу по протоколу CIFS

Следующий пункт, очень важный в организации файловых ресурсов: **Share Display Option**. Если сделать активным параметр «Hide this share when a user browses the ReadyNAS for available shares» («Скрывать этот ресурс, когда пользователь просматривает ReadyNAS в поиске общих ресурсов»), то данный общий каталог становится невидимым для клиентов.

**Настройка Recycle Bin, то есть Корзины**, заключается в первую очередь в возможности активации настройки корзины, а также в указании временного периода в днях, по истечении которого нужно удалять файлы («Remove files older than: ... days»), и размера самой корзины в мегабайтах («Limit Recycle Bin to: ... MB»).

Еще один раздел, заслуживающий внимания: **Advanced CIFS Permission**. Здесь предлагается настроить права доступа для вновь создаваемых файлов.

**Установка флажка *Automatically set permissions on new files and folders*** разрешает установку разрешений по умолчанию для вновь создаваемых файлов или каталогов.

**Do not allow ACL changes to be more restrictive than this** – «Не позволять ACL устанавливать более ограниченные параметры, нежели эти».

Далее идет краткое пояснение, основной смысл которого заключается в том, что когда создается новый файл посредством доступа через протокол CIFS, то устанавливаются следующие типы разрешений: разрешения для владельца файла или каталога, разрешения для группы и разрешения для всех остальных (данная схема организации прав доступа является стандартной для Unix-систем).

После установки флажка **Automatically set permissions on new files and folders** становится доступным меню выбора для групп (Group rights), где предлагается выбрать следующие значения: чтение-запись (Read/write), только чтение (Read-only) и доступ запрещен (Disabled). Список с такими же значениями доступен и для всех остальных (Everyone).

В нашем случае установим следующие значения: чтение-запись для группы и запрет на доступ для всех остальных.

Полностью аналогичные действия предлагается произвести и для вновь создаваемых каталогов.

**Еще один интересный параметр – Opportunistic Locking.** Увеличивает производительность CIFS, позволяя использовать кэширование файлов на Windows-клиентах.

Флажок **Enable oplocks for this share**, который разрешает кэширование, включен по умолчанию.

## 11.3. Continuous Backup с рабочих станций на сетевое хранилище

По правилам все рабочие файлы размещаются на выделенном файл-сервере. Но реализация данной политики целиком и полностью зависит от множества факторов: особенностей бизнес-процессов, наличия «привилегированных» сотрудников и других обстоятельств, которые невозможно поменять за короткое время. Поэтому неплохо иметь инструмент, который позволяет без потери быстродействия, не беспокоя пользователя, копировать содержимое файлов и каталогов на компьютерах пользователей в централизованное место хранения.

И такой инструмент есть. Программа NTI Shadow for ReadyNAS – входит в комплект поставки сетевого хранилища и находится на прилагаемом CD в директории ...bin/shadow. Установка программы должна производиться на каждой рабочей станции, но происходит довольно просто.

Достаточно запустить файл setup.exe, принять лицензионное соглашение, указать имя пользователя, название организации, нужно ли производить установку для всех пользователей или только для текущего – вот и весь процесс установки.

Сразу за этим появляется основное окно, и программа готова к настройке резервного копирования. Так как невозможно охватить все ситуации, при которых может понадобиться теневое копирование, продемонстрируем на примере сохранения папки «Мои документы».

В основном окне программы есть кнопка **Создать задание резервирования**. Появится окно **Выбор файлов и папок**, где выбираем объекты для копирования. Дополнительно, нажав кнопку **Настройки**, можно вызвать окно задания дополнительных параметров, в котором можно задать фильтры по расширению для копируемых или игнорируемых файлов (см. рис. 11.10).

В следующем окне **Где и когда резервировать** указывается ресурс для хранения резервных копий (каталог на сетевом хранилище ReadyNAS) и задается расписание создания резервных копий. В данном примере указан параметр **Сохранить изменение папки/файла в месте назначения каждый раз при сохранении их на компьютере**. Преимуществом этого метода является то, что в данном случае резервирование происходит автоматически, как только файл закрывается пользователем (становится доступным другим приложениям), вне зависимости от того, является ли это просто изменением в старом файле или создан новый.

В качестве альтернативы можно выбрать сохранение файла через определенный интервал или еженедельный бэкап в заданное время (см. рис. 11.11).

Потом нужно указать количество предыдущих версий для сохранения. В данном примере указано число «3» (то есть сохраняются три последние версии файла). Альтернативой выбору фиксированного числа версий может быть сохранение абсолютно всех версий или совсем не сохранять предыдущие версии файла (см. рис. 11.12).

В заключительном окне **Обзор задания резервирования** будет предложено ознакомиться с параметрами вновь созданного задания.

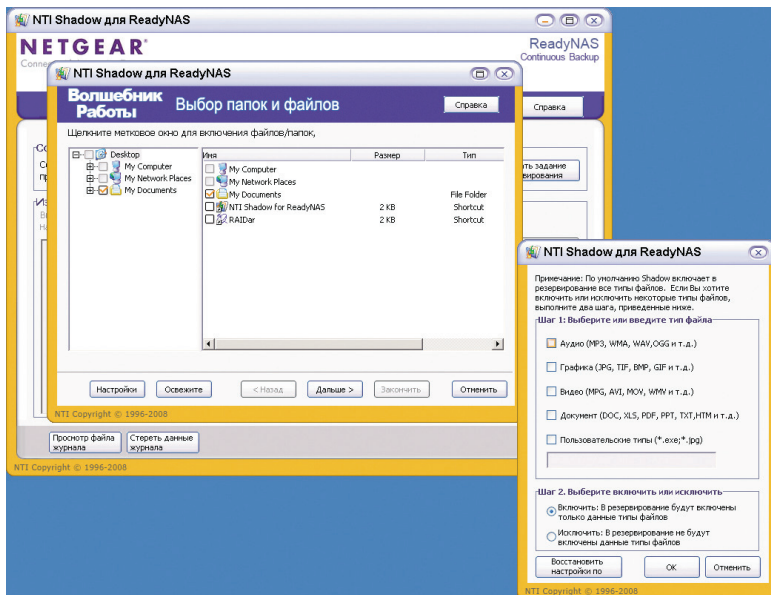


Рис. 11.10. Окно **Выбор файлов и папок** программы NTI Shadow for ReadyNAS с открытым окном настроек

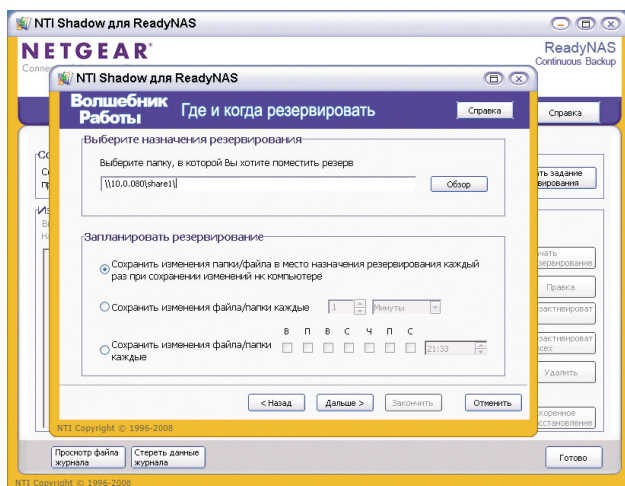


Рис. 11.11. Окно **Где и когда резервировать** программы NTI Shadow for ReadyNAS

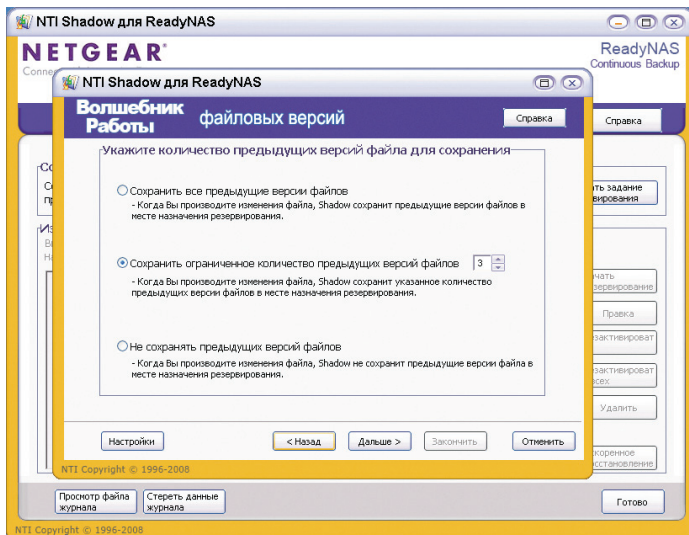


Рис. 11.12. Окно «файловых версий» программы NTI Shadow for ReadyNAS

В случае необходимости можно будет вернуться назад, воспользовавшись соответствующей кнопкой, чтобы изменить свой выбор. И в самом конце программа в небольшом окошке **Начать резервирование** предлагает либо немедленно запустить процесс резервного копирования, либо просто включить данное задание.

Задание создано, теперь проверим возможность восстановления. Нажимаем кнопку **Ускоренное восстановление**, перед нами появляется окно с аналогичным названием, в котором нам советуют использовать правую кнопку мыши для выбора восстановления либо в предыдущее расположение, либо в другое место. После нажатия кнопки **Открыть** мы попадаем в обычное окно Проводника Windows, в котором нам предлагают выбрать файлы для восстановления (см. рис. 11.13).

В случае выбора другого местоположения будет предложено выбрать, куда восстанавливать файлы, и появится окошко, в котором можно будет проследить процесс восстановления файлов. Как видно из примера, операции резервного копирования и восстановления при помощи «NTI Shadow for ReadyNAS» выполняются крайне просто.

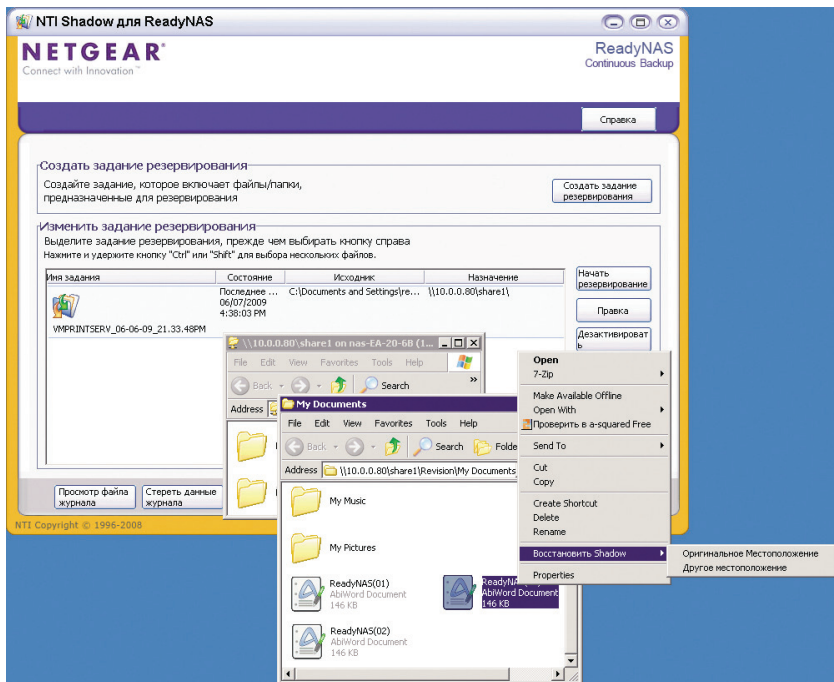


Рис. 11.13. Выбор файла для восстановления

## 11.4. Настройка резервного копирования содержимого хранилища

Система резервного копирования порой тоже нуждается... в резервном копировании. Тем более если для хранения копий используется NAS, доступен 24 часа в сутки и потому подвергается риску вместе с остальной инфраструктурой.

Задания такого рода используют самые простые схемы ротации, например два жестких диска, подключаемые по USB-интерфейсу, которые поочередно перемещаются за территорию предприятия. Наиболее простая схема ротации – раз в неделю делается Full Backup, потом Incremental, далее носители заменяются.

В случае возникновения форс-мажорной ситуации один USB-диск всегда будет находиться в безопасном месте. Из-за небольшого размера носителя не получится копировать абсолютно все содержимое

хранилища, приходится довольствоваться только файлами, содержащими критичную для бизнеса информацию.

Управление процессами резервного копирования содержимого хранилища ReadyNAS, как и все остальные операции, настраивается через веб-интерфейс по адресу `http://_адрес_хранилища_/admin/`.

Зайдя на страницу, переходим в раздел **Backup – Add a New Backup Job**. Здесь предлагается в несколько шагов создать задание по резервному копированию.

**STEP 1 – Select backup source (Шаг первый – Выбор ресурса для копирования).** Предлагается выбрать, что именно мы собираемся копировать. Причем можно выбрать не только каталоги на данном сетевом хранилище, но и расположенные удаленно, например на FTP-сервере (см. рис. 11.14).

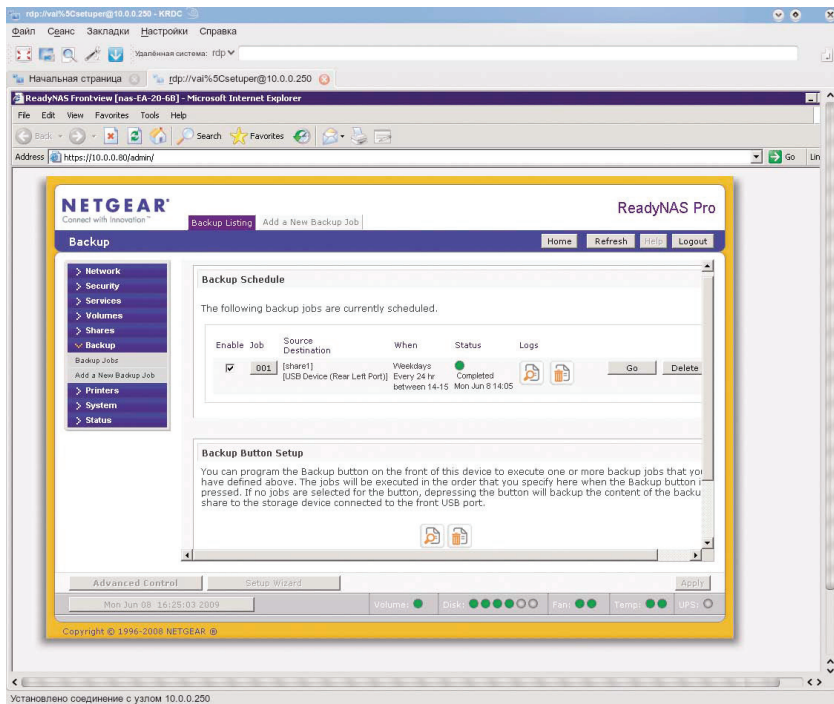


Рис. 11.14. Выбор объектов и конечного местоположения для резервного копирования

**STEP 2 – Select backup destination (Шаг второй – Выбор местоположения для резервной копии).** На этом этапе выбираем цель, куда будем осуществлять резервное копирование. В данном случае в качестве цели выбран USB-порт, к которому подключен жесткий диск.

**STEP 3 – Choose backup schedule (Шаг третий – Выбор расписания резервного копирования).** В основном идет привязка к суточному и недельному графику.

**STEP 4 – Choose backup options (Шаг четвертый – Выбор дополнительных параметров).** Далее следует ряд параметров:

- **Schedule full backup.** В ниспадающем меню предлагается выбрать **Every time** («каждый раз»). Альтернативой является выбор одного из следующих пунктов: **First time** («Первый раз»), **Every week** («Каждую неделю»), **Every 2 weeks** («Каждые 2 недели»), **Every 3 weeks** («Каждые 3 недели»), **Every 4 weeks** («Каждые 4 недели»).
- **On backup completion, send (errors only / full backup logs / status and errors) to the alert e-mail address.** Выбор типа уведомления по e-mail: выслать только ошибки, полный журнал событий резервного копирования или статус и ошибки.
- **Remove the contents of the backup destination before a full backup is performed...** Данная функция удаляет файлы старой резервной копии при выполнении Full Backup (полного копирования). По умолчанию данный флажок снят.
- **Remove deleted files on backup target (rsync only).** Данная функция актуальна только в случае копирования на другое сетевое хранилище ReadyNAS.
- **After backup is complete, change ownership of files in the backup destination to the share owner if the destination is a ReadyNAS share...** Сменить владельца после выполнения резервного копирования, если для сохранения резервной копии используется общий ресурс ReadyNAS. В нашем случае в качестве сменных носителей используются USB-диски с файловой системой FAT32, поэтому данная функция не имеет для нас никакого значения.

По окончании можно демонтировать USB-носитель и отправить его за пределы предприятия.

## 11.5. Заключение

В заключение мне остается напомнить, что рассмотренная схема скорее подходит для малого офиса, нежели для крупного предприятия. Лучше всего спроектировать и внедрить систему резервного копирования уровня предприятия раньше, чем ИТ-инфраструктура и обрабатываемые объемы информации перерастут существующую минисистему сохранения данных.

# Глава 12

## Сеть обмена данными и резервное копирование

*– У нас небольшая проблема с сервером....  
Оппа! Уже большая!*

(Из телефонных переговоров  
аутсорсинговой компании)

### 12.1. Страшная история со счастливым концом

Жила-была организация, не очень большая и не очень маленькая. И была у ней система резервного копирования. Аппаратное и программное обеспечение для этих нужд было выбрано с учетом ежедневных требований, а о большем и не помышляли. И вроде бы все было сделано неплохо, то есть по науке: централизованная топология, схема ротации резервных носителей с учетом требования бизнеса: «дед–отец–сын». Такая схема вполне подходит для организаций, где большая часть персонала работала по 40-часовой рабочей неделе.

Но даже в такой уютной обстановке случаются чрезвычайные происшествия. Однажды в субботнюю ночь администратор резервного копирования проснулся от телефонного звонка. Дело было в том, что менеджеры одного из дальних филиалов решили устроить рекламную акцию в формате «non-stop». То есть и в будни, и в выходные круглосуточно. И конечно, все было сделано, не особо раскрывая деталей, ни с кем не советуясь, не поставив в известность ИТ-службу. Вполне себе штатная ситуация для большинства российских компаний.

И в самый важный и счастливый миг эти менеджеры вдруг осознали, что система учета бонусов просто не работает. Никак. В панике

они позвонили ИТ-директору. Он, в свою очередь, перезвонил начальнику отдела администрирования, а тот – одному из системных администраторов, который сумел понять: SQL-сервер, на котором находилась база данных учета бонусов, парализован каким-то очень мощным ресурсоемким процессом. Анализ сетевых соединений ясно показал, что этот самый чудесный процесс принадлежит агенту резервного копирования, который запустился в штатное окно бэкапа и начал свою работу по сохранению данных компании. И надо только прервать бэкап, чтобы восстановить работу сервера СУБД в штатном режиме.

Казалось бы, все просто, но вполне себе неплохая программа резервного копирования имела маленькую особенность. Она попросту игнорировала любые попытки настройки приоритетов распределения ресурсов. То есть что бы администратор ни делал, программа-агент продолжала пользоваться всеми ресурсами в свое удовольствие и отправлять данные мощным потоком по локальной сети. Кстати, во всем остальном к работе данного ПО претензий не было. Пока работа шла по графику и по выходным не было такой интенсивной нагрузки, такая «бесцеремонность» со стороны backup-системы никого особо не беспокоила. И не было бы вовсе никаких неприятностей, если бы администратор резервного копирования знал о предстоящей акции и настроил запуск заданий после окончания этого волнующего события. Но в российских компаниях проблемы «айтишников» мало кого волнуют, вот и случилась беда.

Ну хорошо, причину выяснили, теперь же нужно что-то делать. Вроде бы нет ничего проще – удаленно подключиться к серверу резервного копирования и выключить задание. Дело сильно осложнялось тем, что данный сервер был не только сервером управления, но еще и медиа-сервером для хранения резервных копий. И просто так «зайти» на него на тот момент было нельзя из-за высокой нагрузки. Те же самые признаки выдавал и сервер SQL, на котором работала программа-агент резервного копирования. Самые известные средства удаленного управления: RDP-соединение и оснастка MMC «Удаленное управление компьютером» – не давали никаких результатов. И конечно, так как до этого все работало без сбоев, начальство ни в какую не соглашалось выделить деньги на KVM-switch. И сервер был самый дешевый, без системы удаленного управления на базе IPMI (например, iLO от HP).

И вроде бы не оставалось никакой надежды, но администратор резервного копирования был человек недоверчивый и хитрый. Он умудрился каким-то уголком подсознания предчувствовать данную ситуацию. На серверах внутри периметра, где устанавливались агенты для резервного копирования, он настраивал и службу сервера Telnet. Де-

далось это исключительно «на всякий случай», и этот случай настал. Но наш герой пошел дальше. На эти же серверы он прозорливо установив пакет утилиты Sysinternals от Марка Руссиновича. В итоге он сумел подключиться по telnet, командой pskill прервал работу агента резервного копирования. Вот такой неординарный вариант развития событий, можно сказать – подвиг, спасший репутацию компании.

Когда данные идут «сплошным потоком», вероятность развития таких событий не так уж и мала. Поэтому эта глава посвящена вопросу: «Как разграничить потоки данных и сохранить возможность удаленного управления оборудованием в любых ситуациях».

## 12.2. Как изначально предотвратить потерю управляемости

Чаще всего доступ к серверу теряется из-за того, что доступ для приложений происходит по тому же сетевому каналу, что и передача данных при резервном копировании и других «тяжелых» задачах (например, репликации или построении отчетов) (см. рис. 12.1).

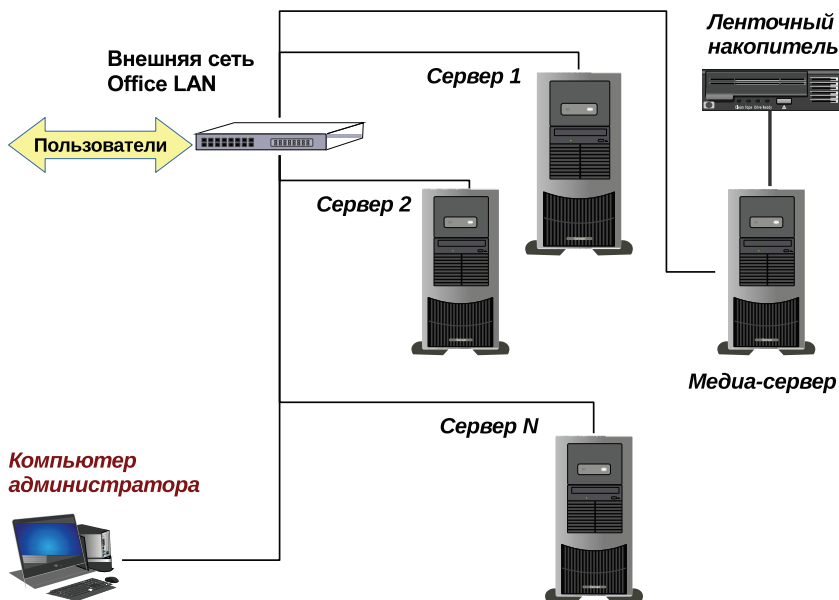


Рис. 12.1. Обычная структура организации сети

Обычно таким событиям предшествует желание сэкономить на всем, что только возможно, или запаздывающий переход от начального уровня организации ИТ-инфраструктуры до уровня Middle, а впоследствии – Enterprise (в главе 4 я писал о том, что Middle-уровень во многих случаях можно пропустить). Но не менее часто встречается ситуация, когда системный администратор, создававший сеть, просто не был знаком с другими возможными вариантами организации сетевой и серверной инфраструктуры. Ниже мы рассмотрим несколько способов, как распределить сетевую нагрузку. Основной постулат при этом: разделение на сеть доступа и сеть обмена данными.

## 12.3. Топология LAN-free и Server-free

В четвертой главе шла речь о различиях в схемах топологии резервного копирования. Напомню, что наиболее перспективным является централизованная топология, когда центральный сервер несет управляющую функцию, а передачей данных во время резервного копирования занимаются программы-агенты.

Но прогресс не стоит на месте, и централизованное решение получило дальнейшее развитие в виде топологии LAN-free.

### 12.3.1. Краткое описание топологии LAN-free

Основной особенностью LAN-free являются наличие в инфраструктуре сети передачи данных (SAN) и передача данных при резервном копировании непосредственно по каналам SAN.

Использование данного термина подчеркивает независимость передачи сети хранения данных от локальной сети и то, что этот подход к организации ИТ-инфраструктуры является наиболее безопасным и, к сожалению, гораздо более дорогим. Учтены многие факторы, такие как разделение нагрузки, динамическое распределение дисковых ресурсов, отказоустойчивость. Есть решения как от известных вендоров, таких как NetAPP, EMC, HP, IBM, так и от менее «раскрученных» брендов. Сети хранения данных имеют замечательные инструменты, такие как создание моментальных снимков (снапшотов) на уровне хранилищ, значительно упрощающих функции резервного копирования и репликации, – все это несомненные преимущества данной архитектуры. Общая схема организации инфраструктуры на базе сетей хранения данных показана на рис. 12.2.

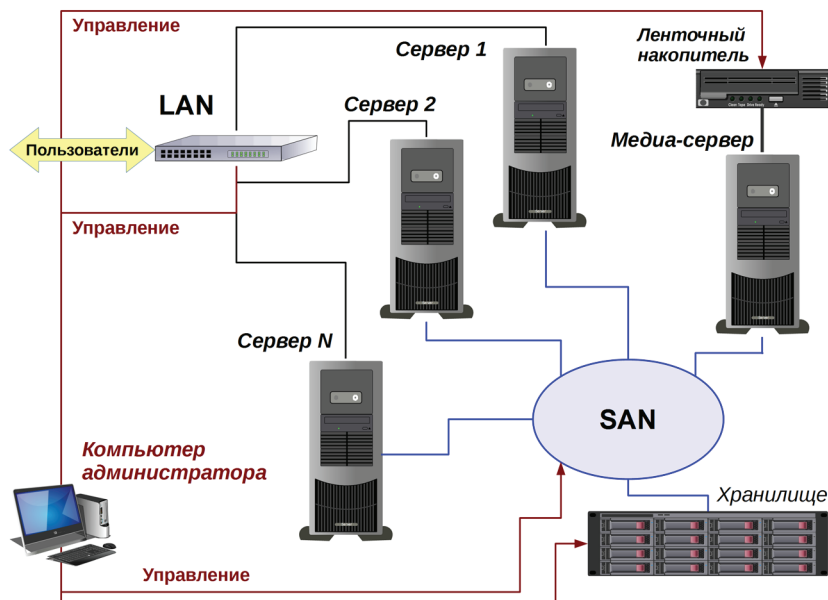


Рис. 12.2. Использование сети хранения данных (SAN) на базе Fibre Channel

### 12.3.2. Краткое описание топологии Server-free

Это фактически дальнейшее развитие топологии LAN-free. Название Server-free служит для обозначения независимости резервного копирования в данной топологии от самих серверов. Данный термин, способный внести сумятицу в понимание самого процесса, на самом деле только указывает на ключевую деталь: сбор данных с хранилища и запись их на устройство хранения резервных копий реализуются, минуя серверную часть инфраструктуры. С беспристрастной точки зрения это не совсем очевидно, серверы и соответствующее программное обеспечение все равно участвуют в процессе, хотя бы как управляющий элемент. Но в целом при использовании такой технологии можно существенно разгрузить как локальную сеть, так и основные рабочие ресурсы.

Справедливости ради стоит отметить, что разделение на LAN и SAN не решает абсолютно всех проблем. Например, некоторое программное обеспечение для резервного копирования виртуальных машин копирует по сети хранения данных только образы виртуальных

дисков, а для переноса дампов памяти виртуальных машин и файлов конфигурации использует локальную сеть. Но даже в этом случае подобное «разделение труда» может значительно облегчить выполнение задачи резервного копирования.

Основное ограничение использования топологии LAN-free и Server-free – стоимость решения. Следует признать, что оборудование и программное обеспечение для обеспечения данного процесса стоят дорого, даже очень дорого. Именно поэтому в свое время были созданы такие решения, как iSCSI, которые позволяют избежать больших затрат на закупку оборудования, ограничившись сетевыми коммутаторами. Ниже мы рассмотрим вариант с заменой классической SAN на базе Fibre Channel сетевым оборудованием для локальной сети.

## 12.4. Использование дополнительной сети передачи данных

Если в бюджете нет денег на организацию классической сети хранения данных на базе Fibre Channel, можно попытаться обойтись менее дорогими, но и менее надежными и производительными средствами. Например, сделать дополнительную локальную сеть для обмена данными между серверами и сетевыми хранилищами, а также для резервного копирования, минуя основную офисную сеть (см. рис. 12.3). Такая организация сетевой инфраструктуры неплохо подходит для средних организаций, при условии что они не будут интенсивно расти и развиваться, и отделу информационных технологий не придется через короткое время ломать голову над тем, как решить проблему с падающим быстродействием в условиях интенсивной работы бизнес-приложений.

Такая сеть хорошо работает в случае, когда для передачи данных достаточно использовать протоколы высокого уровня для предоставления общего доступа к ресурсам: CIFS (SMB), NFS, FTP и т. д. Если говорить о специфических протоколах, которые используются для передачи потока данных от программ-агентов к медиа-серверам резервного копирования, то здесь также ожидается прирост производительности за счет высвобождения полосы пропускания. Но самое главное – в этой ситуации уменьшается вероятность возникновения инцидента, когда из-за сплошного потока данных становится невозможна любая работа с сервером.



При организации внутренней сети очень важно понимать, что для максимальной разгрузки сетевого оборудования крайне важно использовать

именно дополнительную физическую сеть, а не отдельный VLAN на тех же самых коммутаторах. Потому что, помимо всего прочего, при передаче данных сплошным потоком возрастает нагрузка на коммутатор. В случае с VLAN к обычному функционалу добавляются операции определения принадлежности кадра Ethernet к тому или другому виртуальному сегменту локальной сети, а также анализ ACL (списков доступа) и выполнение дополнительных условий, таких как управление рассылкой широковещательных пакетов и т. д. В итоге с применением VLAN время задержки по вине сетевого оборудования может не только не снизиться, но даже возрасти. Поэтому лучше всего организовать отдельную сеть с максимально широким каналом (насколько позволяет бюджет).

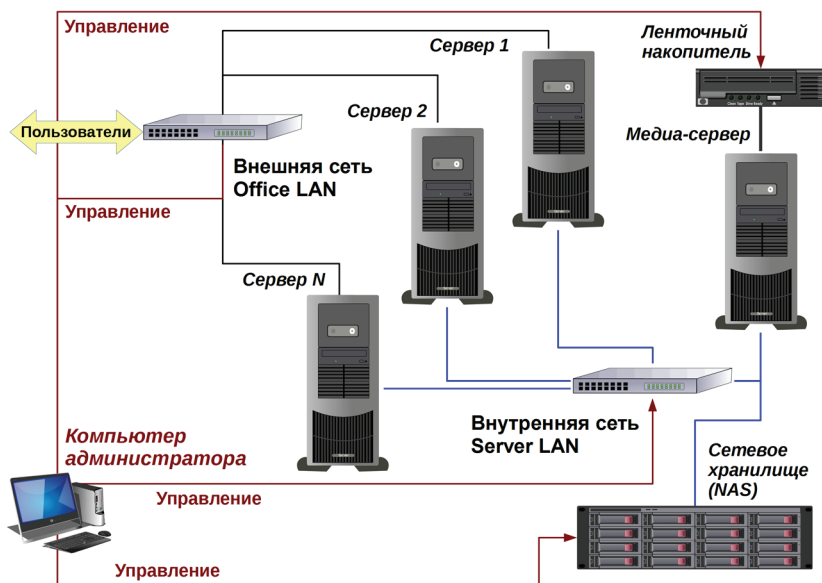


Рис. 12.3. Использование дополнительной сети

Гораздо хуже обстоит дело, когда на базе локальной сети приходится эмулировать работу сети хранения данных, например используя iSCSI.

Несмотря на некоторую экономию при первоначальной закупке, впоследствии по мере развития сетевой инфраструктуры такая попытка использования LAN в качестве суррогата сети хранения данных может привести к значительным затратам при масштабировании и обслуживании. Начиная от необходимости использовать дорогое сетевое оборудование, рассчитанное на полосу пропускания 10 Гб, включая специализированные сетевые карты, «заточенные» под ра-

боту с iSCSI, и заканчивая повышением нагрузки на сетевые ресурсы при обслуживании передачи трафика протоколов, таких как iSCSI, средствами конечных устройств (серверов).

Аналогичную ситуацию с iSCSI можно наблюдать и в системах виртуализации серверов. При использовании небольших виртуальных сред, нацеленных в основном на экономию ресурсов за счет консолидации нетребовательных к ресурсам серверов, таких как контроллеры домена, управляющие серверы и т. д., подключение хранилищ по NFS себя зарекомендовало довольно неплохо. Если функционала и производительности ресурсов становится недостаточно, лучше сразу начинать готовить проект внедрения классической схемы SAN на базе FC.

Еще один интересный нюанс. Протоколы семейства TCP/IP изначально создавались для передачи данных на большие расстояния при использовании каналов связи, порой не слишком хороших в плане ширины канала или устойчивости соединения. Спецификации этих протоколов в большинстве своем содержат возможности контроля ошибок и «правила поведения» при обнаружении таких проблем. В свою очередь, специальные протоколы передачи данных, например тот же Fibre Channel, изначально создавались для высоконадежных скоростных сетей. Поэтому решения на базе IP-сетей будут работать там, где «классические» сети SAN будут отказывать. Например, если по причине повышенной секретности необходимо перенести медиа-сервер на другую площадку с посредственным каналом связи, в этом случае лучше использовать сеть на базе TCP/IP с ее протоколами обмена данных прикладного уровня.

Подводя промежуточные итоги, стоит отметить, что даже при дефиците бюджета необходимо сделать все возможное для улучшения отказоустойчивости и оптимизации потоков передачи данных. И ввод в строй дополнительной сети «для служебного пользования» в этом случае является одной из ожидаемых мер.

Далее мы рассмотрим ситуацию для небольшой организации, которая располагает совсем небольшими средствами, и как в этом случае обеспечить управляемость серверным оборудованием, несмотря ни на что.

## 12.5. Организация дополнительной сети для управления серверами

Если вспомнить пример из практики, описанный в начале главы, то становится очевиден основной источник проблемы. В данном случае

плотный трафик мешал выполнить любые функции по удаленному управлению сервером, используя обычную локальную сеть. Наличие дополнительной сети для обмена данными решило бы проблему за счет перераспределения основного потока трафика.

Но для небольших организаций, в которых менее 10 серверов, включая вспомогательные, на раннем этапе не имеет смысла создавать дополнительную сеть для обмена данными. Во-первых, маленький бюджет не позволит сразу организовать зрелое решение на высоконадежном оборудовании. Во-вторых, при небольших объемах передачи данных мощное сетевое оборудование попросту может простаивать.

Использование дешевого оборудования от сомнительных вендоров чревато негативными «сюрпризами» в виде сбоев, плохой работы под нагрузкой и т. д. Кроме того, по мере развития инфраструктуры все равно потребуются модернизация, а в случае с дешевыми решениями может встать вопрос о полной замене имеющегося оборудования. Получится, что «скупой платит дважды».

В то же время есть возможность избежать подобных затруднений. Решение заключается в организации дополнительно мини-сети исключительно для управления оборудованием. В случае с организацией полноценной сети передачи данных рекомендовалось строить именно вторую физическую сеть и не использовать то же самое сетевое оборудование, даже с организацией отдельного VLAN. Но поскольку в данном случае нас интересуют лишь функции управления, при которых уровень загрузки сети минимален, то вполне можно обойтись и одним из имеющихся коммутаторов. Чтобы избежать дополнительного широковещательного трафика, лучше создать отдельный виртуальный сегмент, но при совсем малых объемах можно использовать тот же широковещательный Ethernet-домен. (При совсем малом количестве оборудования – не более трех-четырех серверов – системные администраторы просто добавляют в свой компьютер несколько сетевых карт и напрямую соединяют свой компьютер с дополнительными сетевыми интерфейсами серверов.) В итоге даже при полной нагрузке на основной сетевой интерфейс системный администратор может подключаться со своего компьютера и управлять нужными участками инфраструктуры. В эту же сеть рекомендуется подключить специализированные интерфейсы управления, такие как iLO, IPMI, IP KVM и т. д.

Данное решение требует совсем малых затрат. Как правило, имеются и не занятые порты в коммутаторе, и свободные сетевые интерфейсы на серверах. Иногда приходится устанавливать дополни-

тельные недорогие сетевые карты в серверы, в этом случае подойдут недорогие, но надежные адаптеры, например производства компаний Intel или 3Com.



Рис. 12.4. Специализированная мини-сеть для управления серверами

В итоге за совсем небольшие деньги можно избежать проблем с потерей управляемости сервера.

## 12.6. Заключение

Все описанные рекомендации не являются взаимоисключающими. В идеале должны быть реализованы сразу все решения для повышения отказоустойчивости и сохранения управления в любой ситуации: сеть хранения данных на базе Fibre Channel, дополнительная сеть для обмена данными между серверами и приложениями и отдельная сеть для управления серверным и другим оборудованием. В итоге ИТ-инфраструктура только выиграет от такого комплексного подхода к данному вопросу.

# Вопросы к третьей части для закрепления материала

1. В чем преимущества и недостатки ленточных накопителей и дисковых массивов?
2. Что означают аббревиатуры NAS и SAN?
3. Для чего нужен стандарт LTO?
4. Что означают термины LAN-free и Server-free?
5. Почему лучше отделять сеть управления и сеть передачи данных?

# Часть IV

---

«Что нам стоит  
дом построить,  
нарисуем –  
будем жить».

Проектирование  
отказоустойчивых  
систем

# Глава 13, которой быть не должно

.....

## Что мешает создать отказоустойчивую инфраструктуру?

*Спешка плоха уже тем, что отнимает очень много времени.*

Гилберт Честертон

### 13.1. Зачем об этом писать?

Число «13» у суеверных людей считается «плохим», «неудачным». И если верить и их представлениям, то эту главу правильнее всего было бы вообще исключить из книги.

Часто бывает, что в организации все кувырком. Диски отказывают, резервное копирование не завершается, серверы виснут, сеть пропадает... Кто виноват? Черная кошка, число 13 или что-то еще?

Прежде чем приступить к проектированию отказоустойчивых систем, необходимо понять и устранить факторы, которые с самого начала могут помешать в реализации этого плана. Часто можно наблюдать картину, когда бизнес и деньги тратит немалые на ИТ, и персонал работает неплохо подготовленный, а ситуация в инфраструктуре близка к критической.

### 13.2. Дуализм подходов к работе ИТ

Есть два различных подхода к обеспечению работы ИТ-инфраструктуры. Несмотря на противоречия, они часто существуют вместе и успешно дополняют друг друга. Мало того, само деление очень условное. В разные моменты жизни в различных ситуациях человек может выступать поочередно в той или иной роли.

### 13.2.1. «Архитекторы»

Эти люди живут и работают по принципу «или делать хорошо, или вообще ничего не делать». Перфекционизм, здоровый пессимизм и чувство меры, объединенные вместе, способны творить чудеса. Системы, создаваемые такими специалистами, оставляют чувство защищенности и уверенности в завтрашнем дне. Например:

- Аппаратное обеспечение подбирается с учетом возможного роста нагрузок. Это может быть необязательно известный бренд. Важно, чтобы было представление о том, как будет работать это «железо» через года полтора-два, когда придет время обновлять ПО и вводить в строй новые сервисы.
- Программное обеспечение закупается с оглядкой на поддержку производителя и «репутацию» разработчика.
- Одновременно с запуском основных сервисов вводится в эксплуатацию адекватная система резервного копирования. Это необязательно должна быть дорогая система уровня предприятия. Даже децентрализованная схема, работающая через тот же Comodo Backup, может спасти ситуацию в трудную минуту. Когда случается горе, уже не так важно, чем сделана резервная копия. Важно, чтобы она была свежей и рабочей.
- Антивирусная защита проектируется и устанавливается сразу, а не когда произойдет первое масштабное заражение.
- Одновременно устанавливается система мониторинга и оповещения о критичных ситуациях. Необязательно закупать дорогую систему. Бесплатный Zabbix вполне способен облегчить жизнь ИТ-службе.
- Система заявок внедряется сразу при создании ИТ-системы.
- Вся информация фиксируется, пишется подробная документация.

Искушенный читатель вспомнит о такой профессии, как «системный архитектор». Да, действительно, чаще всего в их число попадают те, кто является архитектором в душе. Но изначально сленговый термин «архитекторы» появился гораздо раньше, чем профессия «системный архитектор».

Этот тип ИТ-специалистов получил свое название от ассоциации со строительной архитектурой. Нужно не просто, чтобы строение было построено, а служило долгие годы и было надежным и безопасным.

«Архитекторов» вне зависимости от типов темперамента и других психологических качеств объединяет одна важная черта: основатель-

ность и некая неторопливость. И очень стойкое отвращение к работе «быстрее-быстрее». Еще одна забавная черта, которую иногда встречал, — их раздражает избыточный контроль. Характер у людей такой — думать о вечном. Даже если строится тестовая система сроком жизни в полчаса.

### 13.2.1. «Шаманы»

Если смысл подхода «архитекторов» заключается в том, чтобы не допустить сбоев, заранее убрав причины возникновения, то подход «шаманов» заключается в быстром их устранении. «Не важно, сколько проблем возникло за данный период, важно, что все они были устранены». Данный подход весьма популярен у определенного круга ИТ-специалистов.

Обозначение «шаманы» именно благодаря умению чудесным, порой трудно объяснимым способностями образом решать всевозможные проблемы, подчас прибегая к нетрадиционным средствам. Например, в случае проблем с доступом к определенному ресурсу в Интернете можно проанализировать список правил на файерволе, а можно посоветовать пользователю применить внешний прокси-сервер. Широкий кругозор вместе с высокой квалификацией позволяет специалистам этой категории поддерживать на плаву ИТ-инфраструктуру даже во время таких жестоких потрясений, как массовое заражение компьютерным вирусом или сбой основных систем и сервисов при неправильно установленном обновлении. Только он один знает, как подправить ключик в реестре, чтобы самодельная программа документооборота перестала отжирать непомерное количество ресурсов или что при подключении МФУ к принтсерверу нужно использовать не штатные драйверы, а более поздней модели. Шаман, да и только!

Большая нагрузка из-за постоянной востребованности, похоже, не мешает большинству «шаманов» насладиться жизнью. Иногда кажется, что они способны извлекать дополнительную энергию из осознания своей ключевой роли в решении проблем. На самом деле каждый «шаман» поистине уникален. Если «архитектор» в своей работе может опереться на типовые решения и предыдущий опыт коллег и построить свою «непотопляемую систему», то «шаман» должен уметь решать проблемы «по ходу пьесы». Помимо навыков владения поисковыми системами, он должен обладать широким кругозором и профессиональной интуицией.

Ни в коем случае нельзя путать «шамана» с простым низкоквалифицированным бездельником. В отличие от лентяев, просто не

желающих учиться и профессионально развиваться, «шаман» очень трепетно относится к процессу чтения технической литературы и изучению своей предметной области. Мало того, многие из них держат целые библиотеки печатных и электронных книг, а также имеют сайты и блоги в Интернете, где они обмениваются информацией и обучают новичков на добровольной основе.

Еще одна интересная деталь – многие специалисты данной группы не любят подробно документировать систему, которую обслуживают. Обвинять их в лени или недальновидности бессмысленно. Дело в том, что человек ориентируется на свои, подчас действительно уникальные навыки и способности и не представляет, как что-то с его точки зрения совсем очевидно, может быть, кому-то непонятно.

Люди с такими навыками бывают очень полезны. Всегда приятно иметь рядом квалифицированного и находчивого специалиста, умеющего решать максимум задач в кратчайшие сроки. Но есть одна маленькая деталь, которая может перечеркнуть всю пользу от его супернавыков. Казалось бы, все здорово, и ИТ-инфраструктура в надежных руках. Пока наш кудесник быстро все устраняет, особых проблем нет. Беда случается, когда он начинает что-то внедрять или тем более проектировать.

Одно из кажущихся преимуществ такого подхода – некоторая экономия средств на начальном этапе. Например, вместо внедрения системы резервного копирования уровня Enterprise можно попытаться использовать бесплатные средства. Проблемы будут ожидать позже, когда возрастет число объектов, подлежащих резервному копированию, и управлять этим процессом станет все труднее и труднее. Придется в спешном порядке переходить на новую систему резервирования уровня предприятия, не прерывая бизнес-процессов. А такой переход обойдется гораздо дороже, и осуществить его гораздо сложнее, чем сразу использовать корпоративное решение, приобретенное с учетом дальнейшего развития.

Хотим мы этого или нет, но большинство крупных компаний, прежде чем стать таковыми, были сначала малым, а потом средним бизнесом. А в секторе СМБ «шаман» чувствует себя как рыба в воде. Мало того, на начальном этапе бизнес действительно нуждается в «шаманах», способных экономить средства компании. Таким образом, с самого начала закладывается неправильное направление развития ИТ-процессов компании, приводящее в тупик, выход из которого – серьезная переделка уже существующей ИТ-инфраструктуры.

При этом все усилия будут сосредоточены не на повышении отказоустойчивости, а на сиюминутной экономии средств и быстрого устранения возможных сбоев.

### 13.2.3. Рекомендации

Это было бы бесполезной тратой времени – диагностировать проблему и не дать ни одного метода ее устранения.

Самое важное и первейшее – уметь поставить себя на место другого человека. «Шаману» бывает трудно отрешиться от своих супервозможностей и представить себе, что будет чувствовать человек, который вольно или невольно окажется на его месте. Поэтому всегда нелишне будет задать себе и другим участникам процесса вопросы следующего содержания: «Как будут решаться вопросы при его отсутствии на рабочем месте, например во время отпуска или больничного? Будет ли шанс у всей остальной команды разобраться с проблемой?»

Что можно посоветовать, если вы столкнулись с «шаманским наслідием»? Вначале, не откладывая, провести аудит и составить документацию. Бессмысленно конфликтовать с ее невольным разработчиком. Лучше войти с ним в контакт и постараться получить максимум информации, которая позволит в дальнейшем залатать все найденные прорехи.

У «архитекторов» при работе таких проблем, как правило, не возникает. Но и у них есть свое слабое место. Дело в том, что построение системы «на века» требует времени, а его, как правило, не хватает. В итоге проект рано или поздно приходится завершать по принципу «хватай мешки, вокзал отходит». Поэтому у любой надежной и хорошо спроектированной системы найдется слабое место, на ликвидацию которого как раз и не хватило времени в конце срока. Избежать подобных ошибок помогает нормально составленный график работ, а также своевременно поданная заявка на перенос сроков окончания проекта. Поэтому и здесь срабатывает принцип «поставь себя на место другого», в данном случае на место своего руководства или заказчика.

## 13.3. Проблемы перехода

Говоря о трудностях, возникающих при перерождении компании из малого и среднего в крупный бизнес, нельзя обойти стороной аспект личного профессионального роста ИТ-специалистов.

Для примера возьмем обычную торговую компанию среднего уровня. Допустим, им повезло, и в роли ведущего системного администратора работает хороший специалист, профессионал, любящий свое дело. Работает довольно долго и успел много сделать для развития ИТ данного предприятия. Он знает всю инфраструктуру до последнего болтика. И пока все хорошо, и все с уверенностью и надеждой смотрят в будущее. Прогресс все время идет вперед, и от нашего героя уже требуют других задач, да и технологии не стоят на месте, а тут еще маячит необходимость модернизации в связи с перерастанием компании из среднего уровня в Enterprise. И тут возникает ситуация, что у этого прекрасного профессионала нет времени ни изучать что-то новое, ни менять или переделывать. Все его жизненные ресурсы уходят на поддержание собственноручно созданной и стремительно устаревающей системы.

Что произошло? Дело в том, что когда наш герой создавал и развивал свою инфраструктуру, он постепенно развивался вместе с ней. Но когда его «ребенок» вырос и достиг потолка в развитии, появились новые технологии, новые направления в ИТ. Однако системный администратор обязан в первую очередь заботиться о своем питомце. Чтобы не остановились бизнес-процессы, чтобы уберечь его от гибели в результате аварии. И что делать в такой ситуации?

На данном этапе можно обратиться к услугам аутсорсинговой компании и принять помощь извне. Но услуги хороших специалистов стоят дорого, и цены у хорошей аутсорсинговой компании могут быть не по карману развивающемуся бизнесу.



На переходном этапе ни в коем случае нельзя делать выбор исключительно из-за цены. Сомнительные услуги посредственных специалистов могут повлечь не только бесполезную трату денег, но и заложить настоящую бомбу замедленного действия, которая вызовет существенные проблемы в ИТ-структуре в будущем.

Поэтому гораздо предпочтительнее пойти другим путем – растить собственные кадры. В создавшихся условиях придется подготовить достойного преемника, а лучше целый коллектив. Только в этом случае ведущий специалист может снять с себя груз ответственности и перейти к вопросам развития и модернизации. К сожалению, идеологические установки, царящие в обществе последнее время, и череда экономических кризисов подталкивают людей, в том числе ИТ-специалистов, к некоему «социал-дарвинизму», который отрицает взаимопомощь и подготовку младших кадров. А жаль. Потому что это сдерживает развитие и просто мешает плодотворно работать. В итоге

в силу смены обстоятельств рано или поздно даже суперспециалист может оказаться один на один с целым рядом проблем, которые невозможно решить в одиночку. И это может стать началом конца для всего бизнеса.

Учите других. Вовремя готовьте смену. Составляйте подробную документацию, и тогда непредвиденных случаев в работе станет меньше. И это даст шанс на дальнейшее развитие.

## 13.4. «Надкусывание»

Интересное явление, имеющее место в ситуациях, когда не хватает времени.

Допустим, необходимо решить 4 задачи примерно одной степени важности, и на решение каждой требуется 1 час времени. Но «на все про все» отведено примерно 2 часа времени.

Что пытается сделать обычный айтишник, не искушенный в ИТ-менеджменте? Во всех ситуациях, которые мне приходилось наблюдать, человек брался выполнять все четыре задачи одновременно, время от времени по очереди переключаясь между ними (см. рис. 13.1).

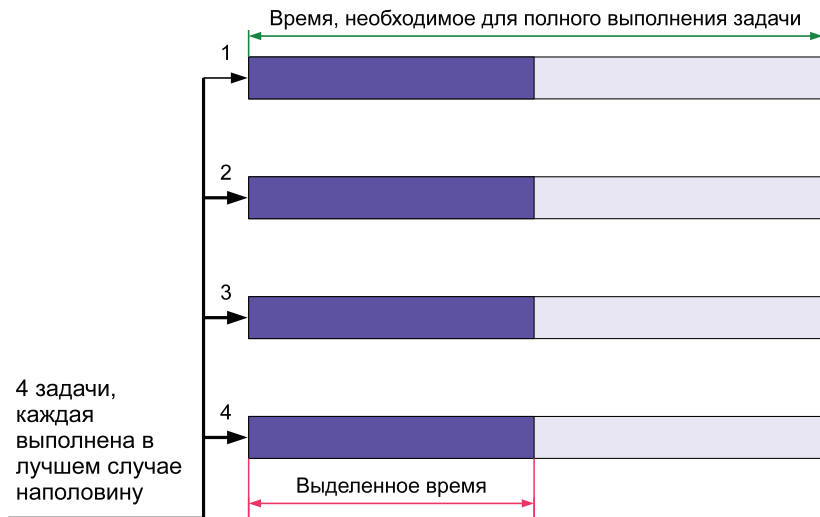


Рис. 13.1. Стандартный вариант «надкусывания»

Результат такого подхода вполне предсказуем. К концу срока все четыре задачи будут выполнены в лучшем случае наполовину. Сле-

дует также учесть, что на переключение между выполнением задач уходит время. Таким образом, даже 50% выполнения каждой задачи – это практически недостижимый результат. Придется отчитываться за четыре невыполненные задачи.

Этот ошибочный путь и получил название «надкусывание», когда исполнитель пытается решить ряд задач одновременно, в итоге проваливает для всех них все сроки исполнения.

Гораздо лучше сразу начать с исполнения только одной задачи, довести ее до конца и только тогда начать выполнять другую. В этом случае есть хороший шанс довести до конца решение двух задач из четырех (см. рис. 13.2). И придется отчитываться только за две невыполненные задачи из четырех.

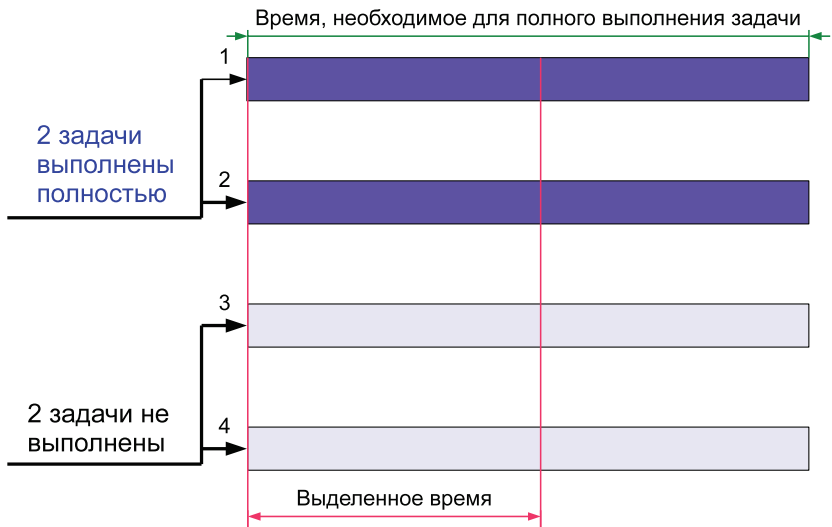


Рис. 13.2. Две законченные задачи лучше, чем «надкусанные» четыре

Разумеется, речь не идет о классических «окнах», когда, например, необходимо запустить резервное копирование в автоматическом режиме на длительный срок, а пока оно выполняется, можно заняться другими проблемами. Такой ситуации не возникает, когда четко удастся определить разные приоритеты для каждой из задач и с самого начала определено, чем можно пожертвовать.

## 13.5. «Дожигание»

Это явление также возникает в условиях сжатых сроков.

Допустим, на нормальное завершение некоторого проекта требуется 10 рабочих дней. Но работу нужно выполнить за 8 дней и к концу 8-го дня представить готовый результат. Перенос сроков не предусмотрен.

### **Что происходит в этом случае?**

Чаще всего исполнитель попытается судорожно выполнить задачу в установленный сжатый срок, без каких-либо ограничений. В итоге часть проекта все равно останется невыполненной, причем какой частью данного проекта придется пожертвовать – решается на самом последнем, «скоростном» этапе. К сожалению, чаще всего жертвуют такими вещами, как отказоустойчивость, безопасность и экономия ресурсов.

### **И что в итоге?**

Порожденный таким образом монстр начинает время от времени подбрасывать «сюрпризы» в виде отказов системы, утечек информации, а то и просто полного выхода из строя. В итоге время, потраченное на ликвидацию последствий, не с лихвой перекрывает «экономия» на этапе проектирования (см. рис. 13.3).

### **Что делать, чтобы избежать подобных ситуаций?**

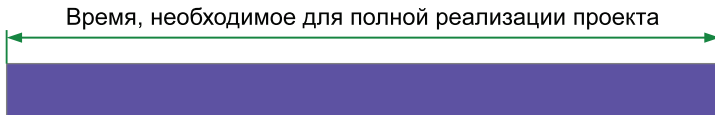
Разумеется, необходимо сосредоточить максимум усилий для получения реальных сроков исполнения задачи. Все остальные рекомендации, которые будут приведены ниже, не являются решением проблемы, а только средством для ее минимизации. Очень важное место заключается в честном и правдивом подходе. Руководитель, отвечающий за проект, обязательно должен быть проинформирован в устной и письменной форме о возможных последствиях такого подхода и, в свою очередь, сам должен донести эти сведения до руководства. Да, при таком подходе велик шанс вызвать недовольство, но когда в процессе эксплуатации всплывут недоделки или выстроенная система начнет рушиться от случайных факторов – будет гораздо хуже.

### **Но если не удалось отодвинуть сроки исполнения проекта?**

Можно попробовать реализовать следующий подход. Очень часто имеет место правило 80/20. 80% – это необходимый функционал, а также первичные меры по безопасности и отказоустойчивости, без них не обойтись. Этот объем работы, как правило, занимает не самую большую часть выполнения проекта. А 20% – это дополнительные

функции, так сказать желательные, но не обязательные вещи. Скептики приводят формулу  $80/20 - 20/80$ , то есть 80% необходимых работ занимают 20% времени, а на 20% дополнительных работ уходят оставшиеся 80% времени. В реальности дела обстоят не так жестоко, но смысл подобного подхода вполне очевиден.

### Необходимые условия



### Работа в режиме дефицита времени и «дожигания»

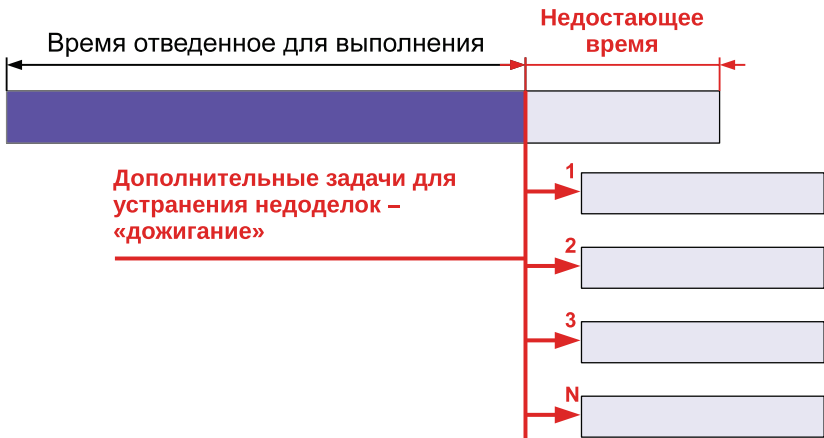


Рис. 13.3. Запланированный вариант и вариант с «дожиганием»

Этот принцип позволяет еще на начальном этапе выбрать, чем можно пожертвовать. Если какие-то функции не особенно важны и роль их не так очевидна, имеет смысл отложить их реализацию на более поздний срок, а может, и вовсе отказаться. Это позволит выиграть время для построения надежной отказоустойчивой системы, которую в дальнейшем можно будет развивать и модернизировать.

Приведу пример. При подготовке фермы терминальных серверов на «строительство» самой фермы, включая монтаж оборудования, настройку дисковой подсистемы, установку программного обеспечения, системным администратором было затрачено около двух недель.

И столько же времени было потрачено на создание возможности тонкой настройки совсем необязательных программ в автоматическом режиме. В результате сроки исполнения проекта были сорваны. Понятно, что в данном случае человек хотел сделать как лучше и оградить себя от дополнительной работы в дальнейшем. Но необходимость использования подобного программного обеспечения не очевидна, тем более не очевидна необходимость изменять настройки по умолчанию.

Еще один хороший способ снизить сроки исполнения проекта – распараллеливание процессов путем отработки решений сложных и необычных задач в тестовой среде. Возвращаясь к нашему примеру, вполне можно было завершить работу с основным функционалом, впоследствии подготовить отдельное решение по тонкой настройке. Даже если потребуется кратковременная остановка системы при внедрении на рабочей ферме – это может быть куда меньшее зло, чем срыв сроков исполнения.

Тестовая система – это очень хорошее подспорье при реализации проектов в сжатые сроки. Можно неделями искать в Интернете подтверждение или опровержение имеющейся информации, а можно проверить в экспериментальной среде в течение нескольких минут. Можно отработать все действия, заранее проанализировать все варианты и после просто автоматически перенести полученные наработки на «боевую» систему. Все рассуждения на тему: «Времени мало, нам некогда этим заниматься», – чаще всего не более чем ленивые отговорки.

Но опять же стоит напомнить: самый надежный способ избежать «дожигания» – установить реальные сроки. В противном случае все равно придется чем-то жертвовать.

#### **А что делать, если такая система досталась «в наследство»?**

Вначале необходимо тщательно изучить всю имеющуюся документацию по проекту. Хотя если создателем данного «чуда инженерной мысли» был «шаман», документации в необходимом объеме, скорее всего, не существует.

Далее в любом случае необходимо провести тщательный аудит с целью как устранения пробелов в документации, так и выявления узких и слабых мест системы.

Дальше идет выработка решений тестирования в экспериментальной среде.

При положительных результатах тестов можно начинать вносить коррективы.

Главное в реализации работ по устранению подобных недочетов – не скатиться к новой схеме «дожигания». В противном случае получится невообразимый бардак, единственным шагом устранения которого будет полная переделка «с нуля» существующей инфраструктуры.

## 13.6. Заключение

В 13-й главе мы познакомились с факторами, способными помешать внедрению отказоустойчивых решений. Как видим, даже номер «13» не помешал провести анализ и сделать выводы о том, какие нюансы нужно учесть, чтобы переход к новой инфраструктуре прошел «как по маслу».

# Глава 14

## Десять шагов к вершине. Процесс проектирования отказоустойчивых систем

*Когда кажется, что все уже работает, все объединено в систему, – вам еще осталось работы на четыре месяца.*

(Чарльз Портман, ICL)

Говоря о процессе проектирования, каждый понимает его по-своему. Кто-то имеет в виду блестящую идею, кто-то – исключительно подготовку документации по всем требованиям ГОСТа. Кто-то считает, что проектированием должны заниматься исключительно «специально обученные люди», а кто-то уверен, что любой системный администратор должен уметь самостоятельно создавать собственные решения и внедрять их в жизнь.

Поэтому предлагаемый в этой главе материал имеет целью обозначить общие моменты, которые являются некой основой для построения системы. И не важно, где и кем работает наш читатель. Прочтя эту главу, он все равно почерпнет что-то новое для себя.

Я сознательно не делал различий между ситуациями, когда над проектом работает команда специалистов из системного интегратора и когда над поставленной задачей трудится один человек, работающий во внутренней ИТ-службе. Разумеется, между этими двумя

полюсами масса различий в деталях, имеется огромное количество различных специфичных требований, но непременно есть и кое-что общее – кто бы не занимался построением отказоустойчивых систем, он должен делать это на совесть. И не может быть никакой разницы в качестве работ, не важно, создается ли данная система «для себя» или «на продажу» стороннему заказчику.

В то же время я старался по мере сил давать комментарии там, где необходимо указать на различия для той или иной области.

Не нужно рассматривать этот материал как некую универсальную инструкцию на все случаи жизни. Основная задача – составить правильное представление. В то же время «все течет, все изменяется», и те нюансы, которые были актуальны сейчас, в недалеком будущем могут оказаться архаизмами.

Весь материал условно разбит на 10 этапов (см. рис. 14.1).

- аудит системы и бизнес-процессов;
- составление технического задания;
- технический проект;
- пилотный проект;
- подготовка рабочей документации;
- внедрение системы;
- подготовка исполнительной документации;
- приемо-сдаточные испытания;
- передача объекта заказчику;
- опытная эксплуатация и техническое обслуживание объекта.

## 14.1. Аудит системы и бизнес-процессов

Наиболее часто встречающаяся ошибка – это попытка выполнить модернизацию ИТ-системы в отрыве от реальной жизни предприятия. Допустим, в связи с возрастанием роли интернет-продаж имеющийся веб-сервер компании перестал справляться с высокой нагрузкой. Требуется провести модернизацию имеющейся системы с учетом требований отказоустойчивости, то есть создать некий кластер высокой доступности и перенести на него сайты компании. Вот и все, казалось бы. Но почему никто не вспоминает про SQL-сервер, который обслуживает запросы? Систему управления базами данных ведь тоже нужно модернизировать и тоже перестроить в кластер. Идем далее. А как быть с почтовой системой? На ней лежит немаленькая роль при обработке заявок. Тоже – в кластер. А система безопасности? Сервер сертификатов? Серверы backoffice, сотрудники которого участвуют

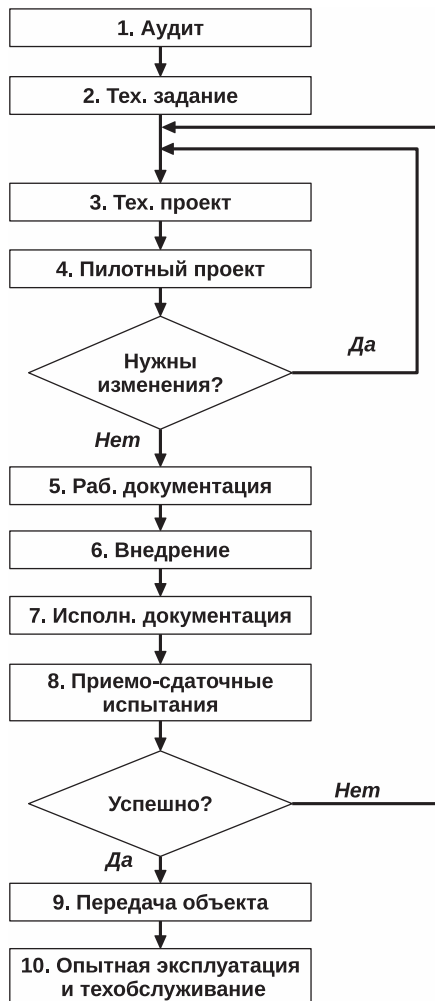


Рис. 14.1. Этапы проектирования отказоустойчивых ИТ-систем

в продажах через Интернет? И наконец, кто-нибудь подумал о системе резервного копирования, которая должна сохранять данные, все время возрастающие в объеме?

Гораздо проще подойти к проблеме с другой стороны и сразу задать вопрос: насколько существующая архитектура соответствует потреб-

ностям предприятия? И как нужно модернизировать систему, чтобы упрочить эту связь?

Для этого и нужен *аудит*. Для сбора информации о нуждах и проблемах заказчика. Заметьте, не конкретной системы, не ИТ-инфраструктуры в целом. Информации о нуждах заказчика, точнее его бизнеса, его организации труда.

Возможно, кто-то из читателей не согласится. Ну а кто же знает лучше свою компанию, как не «айтишники», которые в ней работают.

Увы, такое возможно далеко не всегда в силу некоторых причин.

**Люди всегда люди.** Часто человек склонен не замечать очевидных вещей. Например, что его замечательной и удобной системы резервного копирования скоро не будет хватать для нужд бизнеса. Или у вечно задержанного сисадмина, заваленного бесчисленными заявками от капризных пользователей, уже нет времени на чтение логов не кстати упавшего сервера.

**Дополнительная экспертная оценка.** Любой профессионал, каким бы классным ни был, не может быть специалистом абсолютно во всех областях. Например, системный администратор, прекрасно владеющий навыками развертывания веб-приложений, может не обладать нужным уровнем подготовки, когда дело доходит до организации системы хранения. Привлечение внешних аудиторов позволяет получить мнение не одного, а целой команды специалистов, каждый из которых имеет хорошую подготовку в определенной области.

**Недостаток ресурсов.** В данном случае речь идет прежде всего о высвобождении рабочего времени специалистов. Не секрет, что в большинстве организаций штатные специалисты загружены по полной (а чаще всего вынуждены работать сверхурочно для выполнения тех или иных работ). И высвободить хотя бы несколько часов на решение вопросов аудита бывает очень и очень трудно.

**Необходимость специальных исследований или испытаний.** Например, при аудите систем безопасности иногда осуществляют тестовую проверку устойчивости системы защиты к вторжениям извне, проще говоря, пытаются взломать систему в целях обнаружения слабых мест в системе безопасности. При проверке системы резервного копирования и/или disaster recovery выполняют попытку восстановления системы в тестовой среде (например, в арендованном на время ЦОД) и анализируют результаты, в первую очередь время, потраченное на восстановление. Разумеется, выполнить подобные задачи силами одного человека или даже целой команды системных

администраторов, и без того загруженных работой, зачастую не представляется возможным.

**Подробная документация**, которая составляется во время аудита также. В предыдущей главе я писал о трудностях при документировании системы, особенно при определенных подходах к работе ИТ-системы. Могу только добавить, что часто волей-неволей люди упускают из виду определенные моменты, которые так и не документируются. Например, согласно существующему распорядку, полное резервное копирование должно запускаться по выходным, но администратор баз данных попросил одну базу данных временно копировать в понедельник утром, потому что по выходным формируются отчеты. Расписание перенастроили, а в документацию не внесли («это ведь все временно»), а в итоге все так и осталось.

По итогам аудита формируется **экспертное заключение**, в котором указываются выявленные «узкие места» и даются рекомендации по их устранению. Помимо наличия полезной информации, такой отчет служит хорошим аргументом при разговоре с руководством компании о необходимости модернизации имеющейся системы, соответственно, о выделении на это денежных средств.

Бывают ситуации, когда аудит с привлечением сторонних экспертов невозможен по некоторым причинам. Например, нет денег. Или в городе (округе, регионе) нет специалистов нужной квалификации. Еще одна весомая причина – режим секретности на объекте. В подобных случаях приходится проводить аудит своими силами. Основное требование для таких случаев – полная беспристрастность и возможность поддерживать тот самый «взгляд со стороны», который присутствует у приглашенных сторонних специалистов. Например, назначить одного из специалистов ответственным за проведение аудита, временно освободив его от исполнения основных обязанностей.

На основании данных, полученных из экспертного заключения, формируются или обновляются другие документы, в частности это BCR, DRP и SLA, а также составляется техническое задание для очередного проекта модернизации.

## 14.2. Составление технического задания

Итак, допустим, аудит был проведен, получены результаты, на основании которых были подписаны соответствующие вышеуказанные документы.

Если текущее состояние дел не соответствует требованиям, предъявляемым бизнесом к ИТ-инфраструктуре, определяется область, подлежащая модернизации, и формируется задание на проектирование.

Техническое задание (сокращенно «ТЗ») – самый важный документ при проектировании. Именно в нем формируются требования, предъявляемые к новой системе, сроки, в которые она должна быть внедрена, и порядок приема в эксплуатацию. При составлении ТЗ требуется ответственный подход, поскольку по итогам работ бизнес получит не более того, что указано в данном документе. Например, если представители заказчика забыли указать пропускную способность сети передачи данных, то исполнитель может запросто подключить критически важные серверы в сеть Fast Ethernet со скоростью 100 Мб/с, даже если на самом деле требовался канал шириной в 10 Гб/с.

Основные пункты, которые должны присутствовать в техническом задании:

- основание для разработки проектной документации (указываются нормативные документы, на основании которых будут проводиться последующие работы и, самое главное, выделяться денежные средства);
- организация – владелец объекта (сведения об организации-заказчике);
- цели и задачи проекта (где, собственно, указывается, для чего этот самый проект был нужен);
- расположение объекта (место, где находится строящийся/модернизированный объект);
- численность сотрудников и режим работы (тоже важный пункт, особенно если работы проводятся в рамках требований сохранения государственной, военной или коммерческой тайны);
- исходные данные (информация об использовании обновляемой части ИТ-инфраструктуры до начала работ. В этот пункт следует включить данные, собранные во время аудита системы);
- требования к решениям (здесь следует описать то, что, в принципе, необходимо получить после реализации проекта. Характеристики, которыми должна обладать новая/обновленная система, и дополнительные условия, например требования к электробезопасности, охране окружающей среды и т. д.);
- **приложения к ТЗ.** Несмотря на то что приложения не являются «обязательной частью программы», их наличие позволяет наиболее полно раскрыть суть поставленной задачи перед

потенциальными исполнителями. Например, можно добавить план здания, информацию о численности персонала, схему вентиляции, электроснабжения и т. д.

Параметры, которые могут потребоваться при работе над проектом, лучше указать подробно в самом ТЗ. Например, если требуется построить отказоустойчивую систему хранения, нужно указать не только требуемый объем, пропускную способность сетевых интерфейсов и параметры портов Fibre Channel, но еще и уровень Disaster Recovery, предпочтительный метод обеспечения отказоустойчивости, предполагаемые «окна» для резервного копирования и/или репликации, средства управления и т. д. Нелишне будет также предусмотреть возможность модернизации, масштабирования и миграции, чтобы по прошествии некоторого времени не пришлось отказываться от данного решения и начинать все с начала.

Если, например, проектируется система резервного копирования, то в ТЗ необходимо указать размер «окна бэкапа», глубины хранения, необходимость выноса носителей off-site и другие параметры, описывающие системы резервного копирования. Что же касается таких глобальных параметров, как RPO и RTO и т. д., то они должны присутствовать во всех ТЗ для любых систем, претендующих на отказоустойчивость.

Крайне желательно указать в техническом задании не только оборудование, но и запасные комплектующие. Например, если создается дисковая система хранения, то весьма вероятно, что один или несколько дисков могут выйти из строя. Даже если жесткие диски на гарантии, на время обмена хранения система все равно должна работать без риска потерять данные. Если же речь идет о проектах с более высоким уровнем секретности, то, скорее всего, служба безопасности вообще не позволит выносить какие бы то ни было носители информации с территории предприятия. Поэтому всегда нужно иметь соответствующий запас сменных комплектующих. Аналогично можно сказать о модулях памяти, системах охлаждения и т. д. В ЗИПе должен присутствовать каждый заменяемый компонент хотя бы по одному на устройство.

На основании технического задания формируется бюджет. Это примерная сумма, которую бизнес готов потратить на модернизацию ИТ-инфраструктуры в рамках данного проекта.



При составлении технического задания полезно закладывать характеристики «на вырост». То есть выше, чем необходимо на данную минуту. Например, если для создания системы резервного копирования требуется

50 Тб дискового пространства, можно смело умножить эту величину на два – 100 Гб. Во-первых, от начала составления ТЗ до окончания проектных работ пройдет значительное время. А существующие бюрократические процедуры согласования способны увеличить любые, даже самые долгие сроки в несколько раз. Во-вторых, весьма вероятно, что следующий шанс для модернизации ИТ-инфраструктуры представится нескоро. Всегда лучше указать несколько более высокие требования, которые, скорее всего, и так урежут до минимально необходимых.

Сформировать довольно сложный и ответственный документ силами внутреннего ИТ-отдела может быть весьма непросто. Поэтому нет ничего зазорного в привлечении внешних независимых консультантов, чья задача – указать на явные промахи и помочь скорректировать требования так, чтобы они, с одной стороны, вписались в выделенный бюджет, с другой – удовлетворяли требования ИТ-инфраструктуры и бизнеса в целом.

Существует практика, когда проведением аудита, консультированием при составлении технического задания, проектированием и внедрением, а также последующим обслуживанием занимается одна организация. С одной стороны, такой подход имеет свои преимущества, поскольку позволяет исполнителю досконально изучить бизнес-процессы заказчика, его ИТ-инфраструктуру и выработать наиболее оптимальный подход к решению задачи. С другой стороны, у исполнителя возникает соблазн подогнать условие задачи под готовое решение. Например, исполнитель не работает с вендорами, поставляющими ленточные накопители, и поэтому технология хранения на лентах объявляется «крайне устаревшей», «прошлым веком», «технически небезопасной», и заказчику из всех сил навязываются дисковые массивы в качестве единственного варианта хранения резервных копий.



Разумеется, все, что сказано выше в кавычках о технологии хранения данных на лентах, является домыслом. Постоянное обновление линейки устройств ЛТО – уже достаточное подтверждение того, что ленточные накопители как метод хранения информации вовсе не собираются исчезать с рынка ИТ-технологий.

Чтобы исключить такие уловки со стороны исполнителя, очень полезно на этапе формирования технического задания привлечь независимого консультанта с целью критической оценки поступающих предложений.

Возможен вариант, когда аудит проводит одна компания, исполнителем является другая, а консультационные услуги по формированию ТЗ – третья, и каждая из них выбирается на конкурсной основе. Такой под-

ход позволяет не только заручиться независимой экспертизой, но и выбрать наиболее выгодный вариант с точки зрения цены и качества.

Готовых рецептов тут быть не может. Хотя на выбор подхода оказывают влияние внешние факторы, такие как действующее законодательство, рекомендации регуляторов, а также сложившиеся традиции. Поэтому применяемый подход для государственных и полностью коммерческих структур может сильно отличаться. В первом случае требуется соблюдать действующие нормативные акты, во втором – представители заказчика имеют гораздо больше свободы выбора.

После формирования технического задание выставляется на **тендер**. Участники тендера представляют на конкурсной основе свои решения, где по итогам конкурса определяется победитель, который в качестве награды получает соответствующий заказ.

Для государственных структур обязательно проведение **открытого** тендера, правила проведения которого жестко регламентируются действующим законодательством и контролирующими органами. Для коммерческих компаний чаще проводят **закрытый** тендер, в рамках которого основным регулятором выступает сам заказчик. Например, возможен отказ претенденту на основании того, что его служба техподдержки плохо себя зарекомендовала.

Если модернизация проводится без привлечения внешнего исполнителя, сугубо своими силами, то здесь также возможно проведение конкурса на лучшее решение. Например, может быть принято решение приобрести готовый NAS или сделать NAS на базе обычного сервера и специализированной операционной системы, допустим FreeNAS, Nas4Free, OpenMediaVault, и т. д. Каждая из сторон формирует свое предложение, а руководство компании решает, какое из них наиболее оптимально, исходя из совокупности различных параметров.

Техническое задание – самый важный документ во всей цепочке документации. Если отнестись к его составлению «спустя рукава», можно получить заведомо не работающее решение. Существуют прецеденты, когда спустя десять лет компания продолжала пожирать горькие плоды неправильной модернизации, основу которой положило ошибочное ТЗ.

## 14.3. Технический проект

После того как было составлено техническое задание с учетом всех вышеперечисленных нюансов, оно становится доступным для проектировщиков. Вне зависимости от того, кто занимается проектирова-

нием: компания – системный интегратор, частное лицо или сотрудники внутренней службы ИТ, – они обязаны соблюдать определенные правила. В то же время условия проведения подобных работ сильно отличаются и зависят от множества факторов: открытый или закрытый тендер, является ли заказчик государственной организацией или это частная компания, требуемый уровень секретности и т. д.

Что самое важное при проектировании отказоустойчивых систем? Нужно, чтобы проект был реальным. Если заявлен требуемый уровень Disaster recovery, то именно он и должен исполняться. Например, если бизнес-процессы заказчика требуют наличия резервного ЦОД, то нет смысла предлагать ему решить проблемы отказоустойчивости за счет «старого доброго бэкапа». В то же время бывает необходимо вовремя пресечь проявления гигантомании и не закладывать в проект ненужных позиций. Например, если по факту для сохранения промежуточных резервных копий подойдет обычный NAS, нет нужды предлагать сложную систему виртуальных ленточных библиотек только потому, что «это решение будущего». Очень часто проектировщики страдают излишним консерватизмом или новаторством, что в итоге не позволяет создавать эффективные решения.

Еще одно важное требование – критичная оценка со стороны заказчика. Например, когда проектируется система резервного копирования, бывает весьма полезно самому себе задать такие «неудобные вопросы», например:

- Что произойдет, когда процесс резервного копирования перестанет попадать в «окно бэкапа»?
- Имеет ли текущий канал запас пропускной способности и на сколько его хватит с учетом роста объема данных?

И многое, многое другое...

Несколько слов о стандартах оформления проектной документации. Надо отметить, что требования в различных случаях существенно отличаются. Например, если для госучреждений требуется жесткое следование ГОСТу, то в случае с коммерческими предприятиями вопросов к оформлению возникает гораздо меньше. Если проект создается в рамках внутренних работ ИТ-департамента, то, скорее всего, будет достаточно, чтобы представленные документы содержали подробную информацию о проекте. Поэтому, руководствуясь принципом Козьмы Пруtkова: «Невозможно объять необъятное», – я не буду подробно расписывать стандарты оформления документации. В конце концов, есть ГОСТ, и для получения самой точной и подробной информации лучше обратиться к первоисточнику.

Несмотря на различия в строгости требований для разных организаций, существует несколько общих принципов, которые должны соблюдаться в любом случае.

Как правило, технический проект содержит следующие разделы:

- «Пояснительная записка»;
- «Альбом со схемами»;
- «Спецификации оборудования»;
- «График работ»;
- «Смета проекта».

Остановимся по порядку на каждом из них.

**Пояснительная записка**, как следует из названия, содержит словесное описание проекта. В ней обязательно должны быть такие разделы, как цель проекта, сведения об организации. В некоторых случаях добавляют краткое описание предпроектного состояния, то есть в какой ситуации находится ИТ-инфраструктура на данный момент. Далее идет техническое описание проекта.

Основная ошибка, которую делают начинающие проектировщики, – слишком глубокое погружение в суть вопроса. Например, проводят описание теоретической части «начиная от рождения Вселенной». Разумеется, необходимо подкреплять практические выкладки теоретическими основами, но все же не стоит превращать документацию по проекту в учебник для вузов.

В пояснительной записке также могут присутствовать дополнительные небольшие схемы, рисунки, диаграммы, таблицы. Весь такой дополнительный материал лучше выносить в раздел «Приложения», который должен находиться в самом конце. В тексте принято указывать только ссылки на нужное приложение.

Обязательно оформляется *содержание* (оглавление), поэтому требуется правильная нумерация разделов и подразделов. Иначе возникает неразбериха, и неподготовленный человек может попросту запутаться в обилии терминов, обозначений, описаний процессов и т. д.

**Альбом со схемами** является не менее важной частью проекта, чем пояснительная записка. Существует довольно справедливый принцип построения документации: «Лучше 1 дополнительная схема, чем 10 страниц описания». Основное требование к схемам (помимо строгого оформления по ГОСТу, когда это необходимо) – их ясность и информативность. Например, наличие обязательных подписей к графическим элементам, нумерация объектов и т. д. В идеале необходимо стремиться к тому, чтобы любой проектный документ был в общих

чертах понятен человеку, чья профессия не связана с информационными технологиями.

**Спецификация оборудования.** Чаще всего представляет собой перечень необходимого оборудования, комплектующих и расходных материалов. Очень ответственный этап работы над проектом, ведь если что-то забыть, то придется покупать отдельно и оплачивать из собственных средств исполнителя. Учитывая тот факт, что на закупку и доставку некоторых компонентов может потребоваться некоторое время, то неправильное составление этого документа может привести к срывам сроков исполнения проекта и штрафным санкциям.

Частая ошибка при составлении спецификации – пренебрежительное отношение к дополнительным позициям. Например, забыли записать инструмент для обжимки кабелей. Если работы ведутся на удаленном объекте, это может стать проблемой. Поэтому необходимо указывать максимально полный список всего, что может понадобиться, включая запасной комплект прокладок, гаек, шайб и болтов на случай «вдруг потеряют» и влажные салфетки для очистки поверхностей. Помимо вопросов материального обеспечения, такой подход позволяет оценить реальную стоимость проекта с учетом накладных расходов.

**График работ.** Этот документ не менее важен, чем все предыдущие. Здесь указываются основные этапы работы над проектом и его внедрением и когда они должны быть завершены, а также кто является исполнителем. В свою очередь, сроки зависят от многих факторов: сколько человек будет задействовано, по какому распорядку, специалисты каких специальностей должны будут привлечены. В каждом случае показатели будут сильно различаться. Например, в одном случае необходимо работать исключительно в вечернее время и по выходным дням, чтобы не прерывать бизнес-процессов заказчика, в то время как в другом – строго с 10 до 17 пять дней в неделю, потому что это режимный объект и бюро пропусков работает по такому графику. В каких-то ситуациях потребуются круглосуточная работа из-за сжатых сроков, указанных в ТЗ. Часто возникает необходимость нанимать дополнительных специалистов или приглашать временных консультантов со стороны. Все это существенно влияет не только на сроки исполнения проекта, но и на его стоимость.

Основная ошибка при составлении графика работ – слишком оптимистичная оценка временных затрат. Существует хорошая практика: когда предстоит знакомая работа, то предполагаемый срок необходимо увеличить в 2 раза, если же предстоит неизвестная работа, то заявленный срок нужно умножить на число «пи» (3,1415926535...).

Иногда бывает, что получившиеся сроки, мягко говоря, далеки от пожеланий заказчика. В этом случае необходимо привлечь дополнительных специалистов. В противном случае можно нарваться на срыв сроков исполнения проекта и штрафные санкции. Скупой, как известно, платит дважды.

Еще одна ошибка при составлении графика работ – делать изначальную ставку на сверхурочные работы. То есть когда одни и те же специалисты вместо положенных по закону 8 часов работают по 10, по 12 и т. д. Никакими премиальными выплатами не получится компенсировать нехватку трудовых ресурсов. Дело в том, что уставшие люди работают медленнее и ошибаются в несколько раз чаще. В итоге можно запросто получить ситуацию, когда одна небольшая ошибка может потребовать переделки всего комплекса работ, а это снова спешка, снова сверхурочные работы в еще более жестком режиме, потом снова ошибки, снова переделка – и так по замкнутому кругу. В итоге подобные «экономичные» проекты так и остаются нереализованными до конца, несмотря на массу потраченных денег, сил и времени. Чтобы не попасть в такую ситуацию, необходимо перестать экономить на людях, соблюдать Трудовой кодекс и учитывать в графике работ реальные человеческие возможности.

**Смета проекта.** Составляется на основании вышеуказанных документов. В ней указываются статьи расходов и общая сумма, в которую исполнение проекта обойдется заказчику.

Основная ошибка при определении стоимости проекта – попытка «подогнать решение под ответ». Например, пытаться указать стоимость комплектующих с учетом еще не полученных скидок от вендора. Или пренебречь дублированием некоторых компонентов по принципу «авось ничего не случится». Или попытаться «сэкономить на людях», о чем я уже писал выше. Все эти «хитрости» в итоге проявляются либо во время приемно-сдаточных испытаний, либо в более близкие сроки – при реализации проекта.

## 14.4. Пилотный проект

Итак, мы будем считать, что у нас уже готов технический проект, авторы которого постарались учесть большинство положительных или отрицательных факторов.

Но что произойдет, если мы вот так, с бумажной документацией, «в глаза не видя» внедряемых систем, оборудования, программного обеспечения, придем к заказчику и скажем: «Все готово для конечного внед-

рения, нужно только купить вот это и это, и можно будет с ходу запустить все в работу»? Разумеется, ни один человек, ни одна команда не в состоянии учесть всех возможных нюансов. Возможно, нам повезет, и мы сможем провести внедрение системы, которая до этого существовала исключительно на бумаге. Но в любой лотерее выигрышей бывает гораздо меньше, чем участников, и если ориентироваться исключительно на удачу, рано или поздно наступит сокрушительное фиаско.

Будет куда лучше, если заранее подстраховаться и провести хотя бы минимальные тесты на проверочном стенде. Например, создается система резервного копирования. Что мешает развернуть в тестовой среде фрагмент некой абстрактной инфраструктуры, схожей с той, что имеет заказчик, установить на нее пробную версию ПО для резервного копирования и провести ряд операций по сохранению и восстановлению информации? Это позволит не только убедиться в правильности основных положений нашего технического проекта, но и провести ряд тестовых замеров. Например, проследить время создания резервной копии нужного объема при определенных характеристиках (пропускная способность сети, скорость работы дисковой подсистемы и т. д.). Конечно, далеко не всегда есть возможность создать точно такую же систему, как у заказчика, но никто не запрещает провести экстраполяцию полученных данных и сравнить результаты с требуемыми характеристиками из технического задания.

Аналогичным образом поступают, например, при создании системы виртуализации, создавая «уменьшенную копию» и проверяя на ней основной функционал будущей системы: развертывание гостевых систем, миграцию виртуальных машин, выполнение операций по обеспечению отказоустойчивости и много другого.

Вышеописанные действия называются *стадией пилотного проекта*, который должен обязательно присутствовать при создании любой отказоустойчивой системы, чтобы обезопасить заказчика от неприятных «сюрпризов», а исполнителя – от потери репутации и штрафных санкций.

Разумеется, на создание тестовой инфраструктуры придется затратить некоторые материальные ресурсы, а это значит, что кто-то должен за это заплатить. Есть несколько путей минимизации расходов на пилотный проект.

Самый простой способ сэкономить – не «затачивать» тестовую среду под конкретный проект, а создать некое универсальное решение для работы со множеством проектов. Например, если речь идет о проектировании отказоустойчивых систем с применением виртуа-

лизации, то инфраструктура для пилотных проектов должна обеспечить тестирование различных виртуальных сред на одном и том же оборудовании. К счастью, в настоящее время существует достаточное количество решений, в том числе и на базе бесплатного ПО, позволяющих развернуть тестовую среду для проверки любых систем, в том числе и резервного копирования.

Другой способ – договориться с поставщиками и заказчиком, чтобы разделить процесс приобретения оборудования для проекта на несколько этапов, при этом в состав первой партии включить все необходимое для организации пилотного проекта. Пока прибывает остальное приобретаемое оборудование и будут получены лицензии на все используемое программное обеспечение, у группы тестирующих появляется время на отработку основных решений. Иногда бывают случаи, что после такого этапа приходится корректировать список аппаратного обеспечения и ПО, заменяя одни позиции на другие, более соответствующие требованиям проекта.

Ну и наконец, третий способ – предложить заказчику приобрести оборудование тестового стенда, например для дальнейшего расширения уже имеющейся инфраструктуры или для обучения персонала, а также для отработки нестандартных ситуаций, например проведения учений по восстановлению данных. Так как это оборудование фактически уже выполнило свою основную задачу, то для дополнительной привлекательности можно предложить его заказчику со скидкой, в рассрочку и т. д.

Следует также отметить важный момент: если предполагается дальнейшее сотрудничество исполнителя с заказчиком в виде технической поддержки, проведения платных консультаций, обучения персонала и других услуг в рамках аутсорсинга, то наличие собственного стенда, частично имитирующего инфраструктуру заказчика, является важным конкурентным преимуществом.

Как видим, при должном уровне планирования и организации работы из стадии пилотного проекта можно извлечь выгоду при небольших затратах.

## 14.5. Подготовка рабочей документации

Внимательный читатель, наверное, уже догадался, для чего был нужен пилотный проект. По его итогам составляются план и методика проведения работ, а также другие документы под общим названием *рабочая документация*.

«А зачем нужна еще какая-то документация, если практически все уже описано в техническом проекте?» – спросит нетерпеливый читатель.

Рабочая документация отличается от проектной более тщательным, более детальным подходом. Например, если в техническом проекте указано: «...производится запись резервной копии на ленточный картридж объемом 1,5 Тб», то в рабочей документации будет указано: «...производится запись на ленточный картридж LTO5 производства компании HP, при этом встроенное шифрование не используется...». Как видим, вся информация значительно конкретизируется.

Мало этого, в рабочей документации, как правило, описываются процедуры подключения оборудования, развертывания программного обеспечения, настройки системы, в общем, вся информация, которая потребуется для полного внедрения спроектированной системы.

В первую очередь это необходимо для того, чтобы, получив оборудование и оказавшись с ним на площадке заказчика, иметь полный план, инструкцию и график работ. Тогда не придется, столкнувшись с той или иной конкретной задачей, срочно искать какие-то советы в Интернете или дополнительно заказывать недостающие позиции оборудования или ПО.

Но у рабочей документации есть и другая, очень важная роль. Наличие копии рабочей документации в большинстве случаев у исполнителя позволяет при необходимости повторить аналогичные работы в других проектах. Разумеется, иногда бывают специфичные требования, когда заказчик из соображений коммерческой или государственной тайны накладывает определенные ограничения на дальнейшее использование разработок, полученных в рамках данного проекта. Но чаще всего правильно составленная рабочая документация не только служит для пополнения базы знаний исполнителя, но и является некой гарантией от текучки кадров. Например, если конкуренты попытаются переманить ценных специалистов, то имеющаяся документальная база позволит ознакомить вновь принятых сотрудников с реализованными проектами и, при необходимости, провести их обучение.

Важной частью рабочей документации являются такие документы, как спецификация оборудования и программного обеспечения, учет трудозатрат, а также общая смета.

Внимательный читатель заметит, что данные документы уже упоминались на стадии технического проекта. Получается, одни и те же документы нужно составлять два раза? Если строго ориентироваться

на ГОСТ, то на стадии составления технического проекта вообще нет ни смет, ни спецификаций. Согласно госстандартам, все это появляется только в рабочей документации. Получается, что зря забежали вперед и зачем-то заранее составляли документы, которые все равно придется уточнять?

Дело в том, что на стадии технического проекта делаются *предварительные расчеты*, которые, в свою очередь, базируются на вышеописанных документах. Это позволяет еще в самом начале определить, стоит ли вообще начинать работы по данному проекту, насколько это выгодно и можно ли вообще его реализовать. Поэтому так важно максимально точно учесть все возможные затраты еще на этой стадии. А уже при составлении рабочей документации можно внести незначительные уточнения и скорректировать предоставленные ранее данные.



Согласно ГОСТ, сметы, спецификации и аналогичные документы входят именно в рабочую документацию. В составе документов технического проекта их быть не должно.

Основная ошибка, которую делают при составлении рабочей документации, – либо все описывается чересчур подробно, и документация разбухает до размеров, когда становится практически не читаемой, либо все описывается слишком коротко и вяло, и в итоге из рабочей документации невозможно что-либо вообще, а инженеры на площадке заказчика от безысходности начинают заниматься «народным творчеством», принимая решения на свой страх и риск. И то, и другое одинаково недопустимо. Рабочая документация должна включать только ту информацию по развертыванию и настройке системы, чтобы подготовленный инженер, пусть даже больной, не выпавшийся, но способный читать и шевелить руками, мог воспроизвести описанные действия.

## 14.6. Внедрение системы

После проведения соответствующей подготовительной стадии наступает время внедрения нашей спроектированной системы.

Основное правило при внедрении любой системы: читать и пытаться понять все, где есть буквы и цифры. Не только рабочую документацию по проекту или многостраничный материал по настройке функционала. Очень важно уделить внимание упаковочному листу (packing list) или инструкции по монтажу сервера в стойку. Это не шутка, лично я знаю случай, когда дисковое хранилище установили

«не той стороной». При монтаже ориентировались на тот факт, что красивая декоративная панель обычно находится спереди, а порты подключения – сзади. Но в данном случае все оказалось с точностью до наоборот. В итоге дорогостоящая и мощная система хранения работала как насос, закачивая раскаленный воздух из «горячего коридора» и транслируя его в основное помещение серверной.

Важный момент: если внедрение или модернизация системы проводится на уже работающей инфраструктуре, то все действия, которые могут повлиять на функционирование бизнес-процессов, необходимо согласовывать с представителями заказчика. Желательно, а чаще всего и обязательно такие согласования проводить в письменном виде. Например, перезагрузка серверов при установке агентов системы резервного копирования. Или тестовый запуск системы репликации с задействованием основного канала обмена данными, что может наилучшим образом сказаться на быстродействии.

При проведении внедренческих работ очень важно избегать стремления что-то улучшить или исправить по собственной инициативе и прямо на ходу, если все это не указано заранее в рабочей документации. Излишний перфекционизм и проведение экспериментов на «живой системе» могут превратить процесс внедрения или модернизации системы в бесконечный «сизифов труд», когда одно «улучшение» несет за собой целый ряд несанкционированных изменений, приводящих к мало предсказуемым результатам.

И очень важно не идти на поводу у заказчика, внося изменения и модификации в еще не завершенную систему «по ходу пьесы». При внедрении должны выполняться только те работы, которые описаны в рабочей документации. При необходимости внесения изменений (а это вполне допустимо и часто встречается) сначала этот этап возвращается на доработку в пилотную стадию, по итогам тестирования вносятся изменения в рабочую документацию, и только потом выполняется модернизация внедряемой системы.

Еще одно серьезное требование при осуществлении работ по внедрению спроектированной системы: обязательно записывать комментарии, делать пометки и т. д. Фактически по окончании работ на руках у команды по внедрению должно быть описание готовой системы. Во-первых, это необходимо для составления исполнительной документации (этот вопрос мы затронем чуть ниже). Во-вторых, пока что неизвестно ни одной системы, которая бы на этапе внедрения не претерпела даже самого маленького изменения, по сравнению с первоначальным представлением. Естественно, со временем такие нюансы

забываются, когда приходит время разбираться с какими-то инцидентами, то возникают разного рода «сюрпризы». К тому же подробное документирование защищает коллектив разработчиков от ситуации, когда какие-то важные данные содержатся исключительно в чьей-то голове, и если носитель этой головы заболевает или увольняется, то весь проект оказывается под угрозой срыва.

Итак, будем считать, что наша система уже внедрена, на руках у команды имеется некий носитель данных со всеми изменениями, дополнениями, исправлениями, а также точной информацией о действиях, выполненных во время проведения работ. Самое время приступить к написанию исполнительной документации.

## 14.7. Подготовка исполнительной документации

Для чего нужна исполнительная документация? И почему недостаточно документов из технического проекта, рабочей документации?

Дело в том, что исполнительная документация отличается от предыдущих версий своей... конкретикой. В нее уже не включаются ни методика развертывания, ни способы применения той или иной технологии. Зато с предельной точностью подробно описываются детали текущей реализации. Например, если в техническом проекте указано: «...передача данных между серверами Server-1 и Server 2 осуществляется по локальной сети в Гб/с...», то в рабочей документации это будет выглядеть как «...для организации сетевого взаимодействия используются сетевые карты Intel Pro 1000 MT и гигабитный коммутатор Cisco WS-C3750G-24TS-S1U...», а в исполнительной документации просто указывается: «сетевой порт NIC1 сервера Server-1 соединяется с портом № 1 коммутатора Switch-1, а сервер Server-2 соединяется с портом № 2 коммутатора Switch-1».

Если рабочая документация делается для того, чтобы проработать процесс внедрения и при необходимости воспроизвести его по новой (хотя бы в голове вновь принятого инженера), то исполнительная документация пишется, чтобы как можно быстрее разобраться в новом «хозяйстве», оставшемся после внедрения. При этом гораздо предпочтительнее, если данные размещаются в виде таблиц, схем и других способов наглядного представления.

Иногда в исполнительную документацию включают различные приложения, такие как инструкции для дежурного администратора,

поэтапный регламент отключения оборудования и другие аналогичные документы, упрощающие организацию работы с системой.

Конечная цель исполнительной документации – чтобы, получив набор папок с соответствующей информацией, специалисты заказчика смогли сами разобраться с внедренной или модернизированной инфраструктурой или предоставить необходимые данные приглашенному консультанту.

## 14.8. Приемо-сдаточные испытания

Говоря об этом важном и, наверное, самом волнующем этапе создания или модернизации отказоустойчивой системы, необходимо выделить несколько основных моментов.

Как всегда, не стоит забывать, что любые испытания должны быть регламентированы. В первую очередь нужно помнить о том, что все характеристики новой системы должны быть описаны в техническом задании. В том числе порядок контроля и приемки системы.

Для ответственных объектов, таких как центры обработки данных различных государственных структур, подобные испытания могут быть дополнительно описаны в соответствующих нормативных актах, обязательных к исполнению.

Согласно вышеописанным документам, формируется *план испытаний*, в котором достаточно ясно описываются проводимые тесты и ожидаемые результаты.

Соответственно, по итогам проведения составляются и подписываются акты испытаний.

Помимо плана испытания, могут потребоваться и другие документы, такие как *акты освидетельствования скрытых работ*, акт о выявленных дефектах оборудования и т. д. Разумеется, состав и порядок оформления документов во многом диктуются спецификой проекта и требованиями внешних регуляторов.

И наконец, самый важный документ – **план устранения замечаний**. Не стоит забывать о том, что идеальных проектов, как и идеальных людей, в природе не существует. И так или иначе испытания выявят ряд недочетов, которые необходимо устранить. Поэтому для подтверждения своего согласия с выявленными претензиями и согласования сроков, необходимых для исправления ситуации, создается вышеуказанный документ, обязательный к исполнению.

При проведении подобных испытаний и заказчик, и исполнитель в равной степени могут совершить два вида ошибок: либо выбрать для

проверки очень сложные тесты, редко встречающиеся в обычной практике, либо максимально упростить задачу, при этом рискуя так и остаться под воздействием факторов неизвестности. В первом случае возникает еще один дополнительный нюанс. При тестировании системы отказоустойчивости легко «перегнуть палку» и привести систему в абсолютно нерабочее состояние, при том что сценарий такого рода испытаний может быть абсолютно нереалистичным. Поэтому при обсуждении и подготовке набора приемо-сдаточных испытаний необходимо в первую очередь руководствоваться здравым смыслом и трезвым расчетом.

Иногда для анализа и подтверждения результатов испытаний приглашают представителя сторонней организации. Об этом пойдет речь в следующем разделе.

## 14.9. Передача объекта заказчику

Наверное, это самый неопределенный из 10 этапов.

Как говорится: «Обо всем об этом можно одновременно говорить очень много и совсем ничего не сказать». Практически все зависит от местной специфики, взаимоотношений исполнителя с заказчиком и просто случайных факторов. Если в случае с внутренними проектами достаточно продемонстрировать начальству результаты работ, то в случае с госструктурами (в особенности если имеет место военная приемка) процесс сдачи объекта представляет собой сложную многоэтапную процедуру.

Существует практика, когда на этапе приемки привлекают внешний источник экспертизы, например специально уполномоченную организацию или просто независимого консультанта. Такой подход, несмотря на кажущуюся сложность, выгоден обеим сторонам. Для заказчика это возможность принять во внимание мнение третьей стороны, не зависящей от исполнителя. А для исполнителя это шанс избежать придирок заказчика. Предполагается, что если некая авторитетная организация соизволила поставить свою визу на документах, значит, внезапно возникшие замечания заказчика не являются достаточно обоснованными.

## 14.10. Опытная эксплуатация и техническое обслуживание объекта

Важная часть, про которую почему-то все забывают. Основная задача грамотного проектировщика – не мечтать о том, как будет здорово,

когда его система будет внедрена и заработает, а подумать, как будет плохо, когда эта система начнет сбоить или вовсе перестанет работать. Тут как раз тот случай, когда один хорошо осведомленный пессимист ценится дороже целой армии оптимистов.

Именно поэтому вопрос дальнейшей поддержки поднимается практически при каждом мало-мальски серьезном проекте.

С одной стороны, это страхует заказчика от представителей «обезьяньего бизнеса», того самого, когда «схватил и на дерево». Логика тут проста: «Если исполнитель всеми силами уваливает от дальнейшей поддержки своей же системы – что-то здесь, должно быть, не так...»

С другой стороны, требование дальнейшей поддержки при грамотном заключении контракта дает исполнителю возможность дополнительного заработка, а также укрепляет связи с заказчиком. Особенно это важно при сотрудничестве с коммерческими предприятиями, которые имеют большую свободу при выборе партнеров. Наличие в штате компании-партнера людей, с которыми уже есть опыт взаимодействия, усиливает стремление сотрудничать именно с этой организацией.

Важной частью такого этапа является опытная эксплуатация, когда исполнитель по договоренности с заказчиком осуществляет пристальный контроль за только что внедренной системой и моментально исправляет все выявленные недочеты. Иногда опытную эксплуатацию назначают до официальной сдачи объекта, и тогда она становится частью приемо-сдаточных испытаний только что созданной или модернизированной ИТ-системы.

Разумеется, наличие определенной базы знаний, хорошо написанной документации, а также тестовой среды, позволяющей воспроизводить аналогичные инциденты в лабораторных условиях, может значительно облегчить техническую поддержку собственных творений.

## 14.11. Заключение

На протяжении всего этого многоэтапного повествования я старался акцентировать внимание на простых принципах народной мудрости: «Сначала подумай, потом делай», «Доверяй, но проверяй», «Семь раз отмерь, один раз отрежь» – и других замечательных высказываниях. Стоит еще раз повторить, что сам факт необходимости создания отказоустойчивых систем есть демонстрация высшей степени превосходства здравого пессимизма над восторженным оптимизмом. Чтобы минимизировать риски, избежать финансовых и любых других по-

теперь, нужно задумываться над каждым шагом и постоянно отвечать на пресловутый вопрос: «А что будет, если...?» И вот только когда уже даны исчерпывающие ответы на самые неожиданные вопросы, можно, что называется, «с надеждой посмотреть в будущее».

В то же время надо понимать, что любые элементы безопасности и отказоустойчивости стоят денег, порой – немалых. Следует не забывать золотое «правило 100 долларов»: «Первые 90% безопасности стоят 10 долларов, остальные 10% – 90 долларов». Именно поэтому так важно самым тщательным образом анализировать риски, составлять исчерпывающую документацию и проверять основные функции системы на тестовых стендах. Соблюдение этих простых принципов позволяет избавить заказчика от неприятных «сюрпризов», а исполнителя – от потери репутации и штрафных санкций.

# Глава 15

## «Возьми с собой ключи от моих дверей». О безопасности резервного копирования

– У нас дыра в безопасности!

– Ну, хоть что-то у нас в безопасности...

(Из жизни администраторов ИБ)

### 15.1. Почему вопросы защиты информации будут всегда актуальны

«А зачем нам заботиться о безопасности?» – спросит недоверчивый читатель: «Мы же не ЦРУ, не Пентагон, не Генеральный штаб... Ну, узнают злоумышленники бюджет нашей компании, но это и так публикуется в ежегодном отчете...»

Даже если предположить, что некая организация не хранит ни коммерческих, ни служебных, ни государственных тайн, у нее нет никаких торговых секретов, все данные о ней доступны из открытых источников, все равно стоит позаботиться о защите. Взлом чьей-то системы сам по себе является хорошим компрометирующим фактом. Если кто-то просто выкрадет резервную копию, а потом опубликует это в качестве «трофеев», да еще и сопроводив уничижительным комментарием, – хорошего тут мало. Потеря доверия со стороны парт-

неров, трудности при получении кредита – это только малая часть неприятностей, которые могут обрушиться на несчастную жертву кражи информации.

Самое дорогое и самое легко похищаемое – это репутация компании.

Резервное копирование – это одна из самых уязвимых точек в системе безопасности. Если сопоставить классическую сетевую атаку хакера с банальной кражей носителя Full backup – это все равно, что сравнить проникновение осторожного ночного вора с наглым вывозом по фальшивым накладным всего имущества со склада на грузовике через распахнутые ворота.

Во второй главе я писал о том, как резервное копирование является основой для построения отказоустойчивых систем практически на каждом уровне. И поэтому забота о безопасности резервного копирования – это первейшая задача при проектировании любой системы в принципе.

## 15.2. Избыток так же плох, как и недостаток

О том, что «безопасности мало не бывает», говорят не только новички, но и, казалось бы, достаточно опытные «зубры» в сфере защиты информации, к авторитету которых прислушиваются многие ИТ-специалисты и начальство. В чем же тут может быть подвох? Ниже приводится описание нескольких важных правил, которые необходимо учесть при построении любой системы защиты данных.

**Меры повышения уровня безопасности стоят недешево.** Например, сертифицированные системы шифрования данных стоят достаточно дорого. Разумеется, существуют и бесплатные продукты, но их использование в России либо запрещено, либо сфера применения очень ограничена. Примерно то же самое можно сказать про многие другие элементы системы безопасности: фаерволы, системы защиты от вторжений, средства контроля и наблюдения за действиями пользователей.

При этом надо понимать, что даже неплохие бесплатные средства, например средства фильтрации трафика, поставляемые с сетевым оборудованием, или средства контроля, встроенные в операционную систему, все равно нуждаются в настройке, документировании и постоянном обслуживании либо, как минимум, наблюдении. На

все это расходуется бесценное время ИТ-специалистов, которые в этот момент могли бы решать другие, не менее важные для бизнеса задачи.

**Ограничения, вводимые в целях безопасности, снижают производительность.** Чем выше уровень безопасности, тем больше ограничений – тем больше неудобств. Не существует вариантов, когда при вводе в эксплуатацию соответствующих систем не пришлось бы чем-то жертвовать. Пример: для доступа к ИТ-инфраструктуре под управлением Active Directory была организована двухфакторная аутентификация. Теперь пользователи вынуждены, помимо запоминая «волшебного слова» (в данном случае PIN-кода), еще и носить с собой смарт-карту. Соответственно, рассеянные люди, забывающие карту дома, в сеть попасть не могут, что может являться весьма раздражающим неудобством. Соответственно, при достаточно обширном сочетании подобных ограничений можно добиться значительного замедления в функционировании бизнес-процессов и даже полной их остановки.

**Чем сложнее система, тем выше риск ее отказа.** Более сложные системы состоят из большего количества элементов, что, в свою очередь, повышает вероятность выхода из строя.

Например, используется смешанная топология резервного копирования, при которой сервер СУБД выгружает рабочие базы данных в специальные файлы, защищенные паролем, а те, в свою очередь, забираются и записываются на ленты системой резервного копирования в зашифрованном виде со специальным криптоключом. Любой мелкий аппаратный сбой (например, некритичный перегрев процессора) во время одной из этих операций может вызвать ошибки при шифровании, что, в свою очередь, превратит резервную копию в нечитаемую. При этом в журналах как сервера БД, так и в системе резервного копирования подобные сбои могут не отражаться. Другая опасность заключается в том, что можно просто утратить один из ключей и получить бесполезный набор нулей и единиц вместо актуальной резервной копии. Особенно такие проблемы характерны для архивного копирования, когда копии могут храниться по нескольку лет без попытки использования.

Становится очевидно, что уровень системы безопасности должен соответствовать требуемому значению. Если принятые меры не защищают от возможных угроз, с большой вероятностью данные будут подвержены утечке. Если степень защиты превосходит уровень угроз,

то возрастают риски, связанные с повышением стоимости обслуживания и вероятностью сбоев самой системы защиты данных.

Существует простое правило «ста долларов» при обеспечении безопасности: первые 90% безопасности стоят 10 долларов, остальные 10% – оставшиеся 90 долларов.

## 15.3. Разделение ролей при работе с резервными копиями

Для упрощения разграничений прав доступа в современных системах резервного копирования (и не только) используют ролевую модель. Проще говоря, все права и доступные функции группируются в некоторый заранее определенный набор – «роль», которая соотносится с тем или иным пользователем или группой.

Обычно выделяют две роли: «администратор системы резервного копирования» и «оператор резервного копирования». Различаются они в первую очередь возможностью управления самой backup-системой. Например, администратор может создавать политики с заданиями на резервное копирование, а оператор сам создавать не может, но может изменять расписание запуска, наблюдать за ходом их выполнения, при необходимости останавливать и запускать заново.

Это делается для сохранности самой системы резервного копирования при делегировании полномочий третьим лицам. Например, при организации защиты филиальной структуры местные сотрудники сами заменяют носители для резервных копий, но они не смогут внести искажения в политики резервного копирования. Происходит своего рода страховка от неквалифицированного вмешательства или заведомого вредительства.

В некоторых случаях вводится еще третья главенствующая роль – «архитектор системы резервного копирования». Чаще всего это сотрудник, в чьи обязанности входит решение глобальных вопросов, связанных с backup-системой, например работа с базой данных, где хранится информация о резервных копиях, решение задач, связанных с криптозащитой носителей (об этом мы поговорим ниже), и т. д. Обычных рутинных задач, связанных с регулярным резервным копированием, вроде запуска-остановки политик и учета резервных копий он не касается. Введение такой дополнительной «настройки» оправдано в крупных компаниях с распределенной многоуровневой системой резервного копирования.

На практике трехзвенное разграничение прав встречается крайне редко. Но почти всегда присутствуют две роли, о которых шла речь выше: администратор и оператор.

Если ИТ-службе предъявляются высокие требования к минимизации простоев и/или сохранности данных, существует практика использования двух независимых систем резервного копирования. Например, одна система использует блочный бэкап и применяется при снятии образов системы для Disaster Recovery, а вторая использует файловый доступ и применяется для организации архивного хранения. Этот довольно популярный способ разделения функций таит в себе дополнительную угрозу безопасности. Необходимо, чтобы за обе системы отвечала одна и та же группа лиц, а полномочия по снятию информации не размазывались тонким слоем по всему коллективу «айтишников».

## 15.4. Классификация и структура методов обеспечения безопасности при резервном копировании

Существует довольно много методов, которые используются для защиты информации от несанкционированного доступа через резервное копирование. Чтобы не допускать излишней избыточности и в то же время не оставить брешь в защите, необходимо использовать некоторую классификацию.

Все используемые методы можно разделить на два основных направления:

- ограничение возможности копирования данных;
- ограничение возможности использования резервных копий.

## 15.5. Ограничение возможности копирования данных

Наиболее идеальная утечка данных в подобном случае — сделать свою собственную резервную копию на удаленный ресурс или на собственный локальный носитель, чтобы потом тихо вынести его за пределы периметра. Поэтому так важно защитить свою систему от взлома еще на этапе копирования.

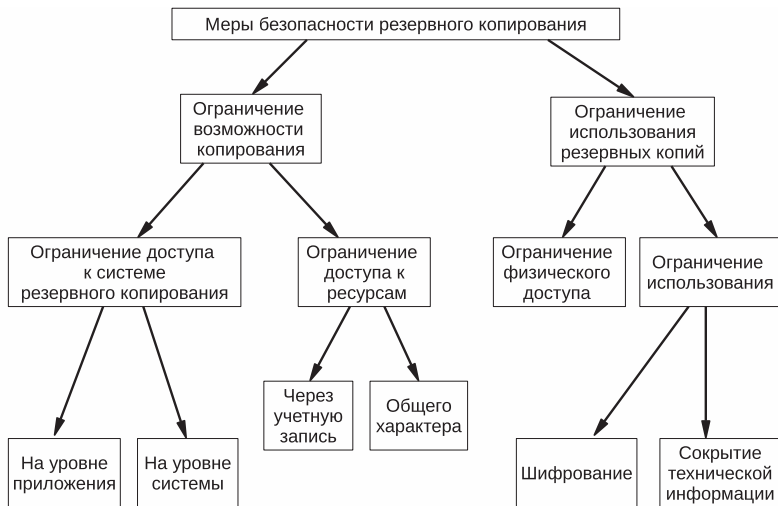


Рис. 15.1. Классификация мер по обеспечению безопасности при резервном копировании

Принимаемые меры безопасности можно разделить на два направления:

- доступ к системе резервного копирования;
- доступ к ресурсам, которые могут быть скопированы.

### 15.5.1. Ограничение доступа к системе резервного копирования

Данный вопрос разделяется на две независимые задачи: защита на уровне приложения, то есть самой системы резервного копирования, и защита на более «низком уровне» – например, ограничение доступа к серверу резервного копирования, отсутствие возможности запуска резервного копирования сетевых ресурсов от недопустимой учетной записи.

Прекрасным примером для данного случая может послужить защита паролем управляющей консоли системы резервного копирования. Когда нет возможности напрямую подглядеть в консоли, что и куда копируется и когда будет готова очередная копия, то получить информацию о месте размещения только что созданной резервной копии, а также о ее содержимом становится весьма непростым делом для посторонних людей.

Помимо пароля на консоль программы, стоит позаботиться об ограничении доступа к самому серверу резервного копирования. Желательно, чтобы к нему невозможно было подключиться не только рядовым пользователям, но и другим ИТ-специалистам компании, не обладающим полномочиями администратора резервного копирования. Что касается роли оператора, то будет весьма неплохо, если он сможет запускать консоль управления со своего рабочего компьютера, не подключаясь к серверу средствами удаленного управления: RDP, VNC, SSH, Telnet и т. д. И, разумеется, не стоит открывать общие ресурсы на любом сервере, предназначенном для системы резервного копирования, будь то управляющие или медиа-сервер. Исключение составляет децентрализованная топология, когда на каждом сервере сети размещается отдельная копия ПО, которая записывает резервные копии на сервер-хранилище. Даже здесь можно ввести необходимые ограничения, например «урезав» права учетной записи оператора, чтобы разрешить только управление консолью управления backup-системы.

### **15.5.2. Ограничение доступа к ресурсам, которые могут быть скопированы**

В любой ситуации всегда есть обходные пути. Например, чтобы получить доступ к данным, необязательно использовать консоль централизованной системы резервного копирования. О методах закрытия таких путей и пойдет речь ниже.

Контроль за ресурсами можно разделить на два направления: ограничение доступности через учетную запись и ограничения общего характера.

### **15.5.3. Ограничение доступа через учетную запись**

Несмотря на то что этот пункт в построении системы кажется очевидным и понятным, все равно его лучше проиллюстрировать на примере.

Допустим, система резервного копирования использует служебную учетную запись пользователя backup-админ для доступа к тем или иным ресурсам. Получив пароль этой записи, можно попытаться подключиться напрямую к требуемым ресурсам. Например, если речь идет о платформе Windows, то можно попробовать открыть себе доступ по протоколам CIFS/SMB, использовать соединение RPC, чтобы разрешить себе доступ через средства управления операционной системой. В волшебном мире UNIX для этих целей подходит вез-

десущий SSH с его возможностью не только управлять системой, но и копировать файлы по протоколам SCP/SFTP.

Еще одна возможность для злоумышленника – постараться перенести нужную информацию с защищенного на публично доступный ресурс. Например, скопировать файлы с закрытого каталога в папку Public, к которой имеют доступ все пользователи. Или сделать дампы базы данных MySQL в папку tmp с последующим копированием в нужное место.

Разумеется, подобные «лазейки» должны быть закрыты, а реквизиты для доступа должны храниться в строгом секрете. И чего уж точно не надо делать, так это использовать учетную запись системы резервного копирования для вспомогательных целей, например для рассылки сообщений системой мониторинга, подключения систем диагностики и т. д. У каждой службы должен быть свой собственный аккаунт с узкими специальными полномочиями.

Особое внимание к безопасности необходимо во время так называемых переходных периодов, когда ИТ-инфраструктура модернизирует систему резервного копирования или меняет ее на более удовлетворяющую. Чтобы абстрагироваться от ошибок из-за нехватки прав доступа, при тестировании новой системы учетной записи предоставляют максимальные права. Впоследствии, когда новая backup-система внедрена и отлажена, про расширенные права забывают, и все остается работать до первого серьезного инцидента. Учитывая уровень доступа, который может получить злоумышленник, этот инцидент может стать критическим для жизнедеятельности организации.

Иногда во время таких переходов работают сразу две системы резервного копирования: старая и вновь внедряемая. Новая система успешно проходит испытания, а устаревшая... так и продолжает работать. Если данных немного и есть возможность складывать копии без переполнения ресурсов (например, на дисковое хранилище с хорошо настроенной системой ротации), то такой «бэкап» может работать довольно долго, радуя своим существованием охотников за чужими данными.

#### **15.5.4. Ограничение доступа общего характера**

В данном разделе речь пойдет о дополнительных мерах, способных усложнить жизнь злоумышленника. К сожалению, в одной главе нельзя затронуть все аспекты обеспечения безопасности, поэтому мы подробно рассмотрим несколько ключевых моментов.

Для подобных мер защиты хорошо подходит система разграничений на уровне сети. Если пользовательская сеть разделена на не-

сколько виртуальных сетей (VLAN), это позволяет ограничить доступ пользователей к тем или иным объектам. Просто разделение сети на подсеть для рабочих станций и серверную значительно усложняет несанкционированный доступ к данным. Дальнейшее дробление еще больше повышает уровень безопасности. Например, можно запретить доступ к серверам баз данных с рабочих станций сотрудников, в служебные обязанности которых не входит работа с подобной информацией. Сотрудникам отдела рекламы совсем не обязательно иметь возможность подключаться к серверу СУБД для бухгалтерии, так же как и бухгалтерам не нужно иметь возможности просматривать свежие эскизы рекламных проектов. Если у каждой группы пользователей есть определенный сегмент, куда они должны получить доступ по служебной необходимости, а вся остальная ИТ-инфраструктура закрыта для проникновения, это можно считать хорошим достижением.

Разделение по протоколам также дает хороший результат. Простое использование файервола значительно снижает вероятность несанкционированного доступа. Большинство систем резервного копирования использует свой собственный протокол для обмена инструкциями и получения данных от своих программ-агентов. Можно просто закрыть соответствующие TCP-порты, оставив доступ только для серверов резервного копирования.

Разумеется, использовать централизованную систему управления сетевым доступом гораздо предпочтительнее, нежели каждый раз настраивать файервол на отдельном сервере. Такое управление возможно, например при использовании сетевых коммутаторов 3-го уровня с поддержкой access-lists – ACL.

Но не стоит забывать о том, что все хорошо в меру. При любых обстоятельствах уровень защиты должен соответствовать уровню угроз, но не более того.

## 15.6. Ограничения возможности использования резервных копий

Данное направление также делится на два основных направления:

- ограничение физического доступа к резервным копиям хранения в надежном месте, опечатывание;
- ограничение возможности применения резервных копий: шифрование, маркировка, сокрытие процедур создания резервных копий (структура данных, график, схема ротаций).

Не будем забегать вперед и рассмотрим все по порядку.

### **15.6.1. Ограничение физического доступа к резервным копиям**

На первый взгляд, это самый бесхитростный пункт в обеспечении системы безопасности резервного копирования. Здравый смысл подсказывает, что ценные предметы нужно хранить в запертом сейфе или несгораемом шкафу в охраняемом помещении с надежной дверью и решетками на окнах. Также не мешают сигнализация и система видеонаблюдения.

Разумеется, кассеты с записью бизнес-информации – это не золотые слитки, но порой стоимость их содержимого может стоить целое состояние. В то же время нет смысла держать резервные копии «за семью печатями», так как данные могут понадобиться в любой момент. То, что я писал выше по поводу избыточности средств защиты, как нельзя лучше применимо к ограничению физического доступа. Нет смысла покупать дорогостоящий сейф там, где можно обойтись несгораемым шкафом. Не стоит покупать сверхсекретный биометрический замок там, где достаточно обычных ключей. Затраты на хранение накопителей диктуют здравый смысл и финансовые возможности.

Но что делать, когда резервное копирование производится на дисковое хранилище? При таком способе хранения резервных копий возникает масса трудностей с обеспечением безопасности. Невозможно каждый раз после создания копии демонтировать хранилище и прятать в сейф. В этом случае в качестве ограничительной меры подойдет надежный серверный шкаф с запирающейся дверью. Очень желательно вынести дисковую систему хранения бэкапов в отдельное помещение или разместить за изолирующей перегородкой.

Но основной способ получения данных с дискового хранилища – через сеть, поэтому нужно обеспечить ограничение доступа по сети. Хорошая идея: защитить общие ресурсы сложным паролем, разместить хранилище за файерволом и настроить список доступа на сетевом оборудовании. Сочетание перечисленных мер дает хороший эффект и предотвращает от большинства вторжений.

Существуют и другие меры ограничения доступа. Например, использование топологии LAN-Free в сети передачи данных (SAN) на базе оптоволоконных соединений значительно безопаснее, нежели использование обычной локальной сети.

Разумеется, ничего не дается бесплатно. Применение файервола снижает производительность локальной сети, а топология LAN-Free, помимо покупки соответствующего программного обеспечения, требует организации самой SAN. Но все же следует понимать важность принимаемых мер и не пытаться выдать за экономию жадность и безответственность.

### **15.6.2. Ограничение возможности применения резервных копий**

Параллельно с ограничением доступа к резервным копиям необходимо предусмотреть меры по защите от использования посторонними лицами. Проще говоря, даже если злоумышленник сможет получить себе экземпляр копии данных, он не должен знать, как это использовать.

Такую защиту предоставляет шифрование данных. Большинство систем резервного копирования имеет свою встроенную криптографическую систему, основанную на известных алгоритмах. Такие резервные копии, создаваемые как на съемных носителях, так и на дисковых подсистемах, прочитать без ключа будет далеко не просто.

Аппаратная поддержка систем шифрования облегчает процедуру создания резервных копий без потери производительности. Например, оборудование, поддерживающее стандарт LTO, начиная с версии LTO4 предоставляет возможность использовать собственный чип, встраиваемый в ленточный картридж и упрощающий процедуру кодирования данных. Обратной стороной шифрования является невозможность сжатия данных.

Для защиты бэкапов, размещенных на дисковых подсистемах, можно порекомендовать, помимо использования встроенных возможностей ПО резервного копирования, использовать VPN-соединение и шифрование дисковых разделов. Это дает дополнительную защиту от атак вроде «Man in the Middle» («человек в середине»), когда посторонний участник процесса подключается к каналу, подменяя ключи шифрования.

Но, помимо шифрования, существует немало других способов усложнить жизнь злоумышленников. Хорошую службу может оказать особая маркировка копий, непонятная для злоумышленника. Например, ленточные носители подписываются не как «База бухгалтерии за июнь», а «Копия 6 источника 1». В идеале для маркировки можно использовать абсолютно нейтральные значения, например се-

рийный номер носителя. В этом случае гораздо сложнее определить, что это за копия и какие данные можно восстановить с ее помощью. Особенно это полезно, когда злоумышленники охотятся за чем-то конкретным, например за данными о контрагентах. В этом случае становится непонятно, какой съемный носитель следует похитить или какой архивный файл переписать с дискового хранилища.

Допустим, наш злоумышленник разобрался в маркировке, сумел похитить локальный носитель или переписать архивный файл с хранилища, получил ключ при помощи социальной инженерии или взломал при помощи специального оборудования и ПО. Далее, чтобы получить данные, ему необходимо воссоздать некую модель, пусть даже уменьшенную, той инфраструктуры, на которой были созданы эти данные. Допустим, данная операция затевалась с целью получения списка клиентов компании. В руках недоброжелателей оказалась база данных. Значит, необходимо воссоздать аналогичную систему, чтобы просмотреть требуемую информацию. Для этого как минимум потребуются знание, на каком ПО резервного копирования была создана данная копия (таких backup-систем сейчас довольно много) и какое приложение использовалось для работы с данными. Также может понадобиться знание паролей администратора СУБД для подключения базы. Чем меньше злоумышленник знает о восстанавливаемой системе, тем большие трудности ему предстоит преодолеть. Соответственно, вся информация о том, какое программное обеспечение используется, в каких форматах хранятся данные, какова процедура восстановления, — все это представляет такую же тайну, как ключ шифрования или пароль администратора. Чем меньше данных мы сообщаем «наружу», тем защищеннее является система.

## 15.7. Заключение

Мы разобрали самые простые и наиболее популярные способы обеспечения при резервном копировании. Разумеется, невозможно в рамках одной публикации описать все многообразие методов, используемых для защиты и нападения. Но, применяя предложенный материал, а также здравый смысл, смекалку и элементарные правила осторожности, читатель сможет разработать свою методику обеспечения безопасности.

# Глава 16

## Жизнь после смерти, или Как восстановить систему после мировой катастрофы

*Все системные администраторы делятся на тех, кто делает бэкапы и кто еще не делает. Соответственно, те, кто делает, подразделяются на тех, кто пробовал восстанавливать данные, и тех, кто еще не пробовал.*

(Народная мудрость)

### 16.1. Предисловие

Существует множество потенциальных угроз для жизни и благополучия ИТ-инфраструктуры. И это не только катастрофы глобального масштаба, такие как пожары, наводнения, извержения вулкана. Немало неприятностей доставляют аварии инженерных систем, в первую очередь систем кондиционирования. В отличие от сбоев электропитания, от которых ИТ-оборудование достаточно хорошо защищено, перегрев из-за сбоев кондиционеров может привести буквально к неисправимым повреждениям аппаратного и программного обеспечения, не говоря об информации, записанной на носители. И конечно, немало проблем приносят ошибки и вредоносные действия людей, как злоумышленников, так и вполне обычных пользователей. Не стоит забывать и о таких обстоятельствах, как банальная кража или изъятие компьютерной техники по другим причинам.

В обыденной ситуации дело не всегда доходит даже до восстановления из резервной копии. Например, системный администратор мо-

жет попытаться восстановить удаленный файл, используя утилиты для работы с файловой системой. Или «починить» жесткий диск, воспользовавшись программой для перенаправления сбойных секторов. А вот при глобальном сбое, повлекшем за собой выход из строя всей ИТ-инфраструктуры, на такую роскошь рассчитывать не приходится.

Некоторые читатели, уже имеющие опыт в сопровождении ИТ-систем, зададутся вполне резонным вопросом: «А какие тут могут быть сложности? Достаточно купить хорошую программу резервного копирования, чтобы иметь возможность при необходимости восстанавливать данные из резервной копии».

Но не следует забывать, что при катастрофических авариях или, например, изъятии компьютерной техники быстро восстановить из резервной копии нереально по нескольким причинам.

Причина первая – некуда восстанавливать. Не забываем, что ИТ-инфраструктура разрушена или попросту недоступна.

Причина вторая – отсутствует источник для восстановления. Не забываем, что если отсутствует вся инфраструктура, то система резервного копирования (и восстановления) отсутствует вместе с ней.

Единственный способ преодолеть этот замкнутый круг – подготовиться заранее к самому худшему.

## 16.2. Несколько слов о резервном копировании

Из предыдущего раздела становится ясно, что простое восстановление из штатной резервной копии и восстановление системы с нуля – это не одно и то же. Разумеется, наличие «бэкапов» всей бизнес-информации является важным условием для защиты ИТ-инфраструктуры от форс-мажорных обстоятельств. Но резервное копирование играет скрытую и очень важную роль в жизни предприятия. Это и восстановление случайно удаленных данных, исправление ошибок в работе пользователей, и гарантия целостности информации в случае хакерских атак, и даже контроль лояльности пользователей. Если у организации нет такого механизма защиты и контроля, у нее нет не только прошлого, но и будущего.

**Резервное копирование должно быть всегда.**

Пытаться работать без системы резервного копирования (или с системой, не отвечающей требованиям) – это как ездить по оживленной трассе без ремней и подушки безопасности, оставив дома запасное ко-

лесо. Нет смысла обсуждать систему восстановления после аварии, если нет защиты на каждый день.

## 16.3. Создание Disaster recovery plan

### 16.3.1. Для чего нужно писать Disaster Recovery Plan?

Ни для кого не секрет, что планировать свои действия заранее и тем более писать планы бывает весьма утомительно. Особенно когда дело касается неких глобальных процессов, таких как восстановление крупной ИТ-системы.

Во многих российских предприятиях и организациях так и поступают – не пишут никаких планов восстановления. Предполагается, что в коллективе работают хорошие специалисты, знающие свою систему до тонкостей, и уж им точно не составит труда с легкостью воспроизвести такую же систему. Обычно эти постулаты порождаются самонимением сотрудников, а уже потом доводятся до начальства. Бывает, что подобную точку зрения навязывает само руководство, не желая тратить дополнительных ресурсов на «непродуктивные задачи». «Настоящему профессионалу не нужны никакие планы, он всегда знает, что нужно делать!» – такой лозунг можно развесить в большинстве серверных на территории стран бывшего СССР.

Подобная иллюзия длится до первого серьезного сбоя. А потом начинаются вопросы, например:

- А кто-нибудь помнит пароль от почтового реля?
- От Exchange?
- Нет, от внешнего, там sendmail...
- А что, у нас и такое есть?
- Есть, точнее было, просто про это мало кто знает...

Или:

– Кто-нибудь знает, как позвонить нашему сетевому администратору? Мобильник не доступен, у кого-нибудь есть его городской номер?

В итоге процедура восстановления, которую планировали завершить за несколько часов, затягивается на долгие дни, а иногда и недели.

Все остальные, кто наслышан о подобных проблемах или уже имел горький опыт, пишут план восстановления.

### 16.3.2. Что такое Disaster Recovery Plan

Disaster Recovery Plan (сокращенно DRP) – это группа документов, детально регламентирующая процесс восстановления системы с «нуля».

Несколько характеристик, отличающих Disaster Recovery Plan от простых «заметок на полях», которые делает любой системный администратор «на всякий случай, чтобы не забыть».

Disaster Recovery Plan представляет из себя многоуровневый набор документов с блочной организацией. Каждый из документов представляет собой законченный набор данных, достаточный для выполнения той или иной задачи. Блочная структура подразумевает, что любой документ может быть заменен или переделан без влияния (или с минимальным влиянием) на другие документы. Проще говоря, чтобы переделать инструкцию по восстановлению прокси-сервера, не придется переписывать весь пакет документов.

### **16.3.3. Описание бизнес-процессов компании и их связь с ИТ**

Самый верхний уровень DRP – набор документов, кратко описывающих работу компании и ее бизнес-процессы. Далее идет описание связи бизнес-процессов и информационных технологий. Это едва ли не самая важная часть документов. Без понимания, какая служба для чего работает, крайне трудно понять, в каком порядке, с какими приоритетами и в каком объеме необходимо восстанавливать ту или иную систему.

Например: программный комплекс 1С используется только для бухгалтерии или в этой системе работают и другие подразделения, например служба поставки и логистики, розничная продажа и т. д. Разумеется, если система «одна на всех», то приоритет ее восстановления повышается.

Далее идет подробная технологическая документация ИТ-процессов и взаимодействия различных сервисов. Если взять на вооружение указанный выше пример электронной почты, то необходима схема с описанием: куда первоначально поступает почтовое сообщение, где проверяется на спам и наличие вредоносных программ, куда пересылается дальше и как в итоге попадает в программу – почтовый клиент пользователя. Существуют ли какие-то системные политики, которые могут влиять на этот процесс, какие фаерволы и другие устройства встречаются на пути почтового потока, какие есть правила и ограничения и т. д. В итоге у прочитавшего этот документ должна сформироваться полная картина работы почтовой системы. И так по всем направлениям, начиная от системы управленческого учета и до АТС и системы контроля физического доступа в помещение.

Разумеется, гораздо проще и удобнее описывать данный процесс соответствующими терминами, как, например, RPO, RTO и другими

«страшными словами», о которых подробно рассказывается во второй главе.

## 16.4. Стратегия восстановления

После детального описания всех бизнес-процессов следует принять непростое решение: на основании чего и как строить стратегию восстановления. Все варианты, принимаемые на данном этапе, можно разделить на два направления.

### 16.4.1. С использованием виртуализации

Развитие подобной стратегии предусматривает практически полный переход на системы виртуализации серверов и десктопов. При этом для восстановления соответствующих виртуальных машин достаточно просто развернуть данную машину из бэкапа и при необходимости актуализировать данные. Например, для восстановления SQL-сервера понадобится «поднять из бэкапа виртуалку», на которой уже установлена операционная система, сервер СУБД, необходимые утилиты управления, при необходимости отдельно восстановить базы данных из последней архивной копии для гарантии актуальности и целостности информации. (Подробнее о вопросах резервного копирования виртуальных машин мы говорили в главе 12.)

При таком подходе восстановление аппаратной платформы сводится к созданию новой системы виртуализации, на которой развертываются заранее сохраненные копии виртуальных машин.

При этом сведена к минимуму необходимость заботиться о соответствии аппаратного набора устройств, который был до аварии. Есть, конечно, некоторые ограничения, например не стоит менять процессоры Intel на AMD, ARM или Power-PC. Что касается драйверов для чипсетов материнских плат, или сетевых карт, или любого другого оборудования, скорее всего, точное соответствие не понадобится. Основное требование к новому оборудованию – совместимость с используемой системой виртуализации. Чаще всего этого бывает достаточно.

### 16.4.2. С использованием физического резервирования

Если стратегия с использованием систем виртуализации позволяет абстрагироваться от аппаратного обеспечения, то при ориентации исключительно на физическое оборудование первую скрипку играет

именно вопрос совместимости с аппаратным обеспечением, которое использовалось до аварии.

Есть несколько приемов, позволяющих избежать проблем с аппаратной совместимостью. Во-первых, крайне желательно использовать оборудование одной платформы, например Intel x86 совместимое, и от одного крупного вендора, например сервера и хранилища только от IBM, или только от HP, или только от HUAWEI и т. д. В этом случае есть хорошая вероятность, что крупный производитель не снимет данную линейку с продаж и совместимое оборудование и комплектующие можно будет купить.

Некоторые программы блочного копирования предоставляют возможность при восстановлении системы из образа установить драйверы от нового оборудования.

В некоторых случаях в качестве образов для восстановления системы используют виртуальные машины. Проще говоря, вместо того чтобы сохранять образы разделов дисковых томов физических серверов, сразу производят миграцию физического сервера в виртуальную машину и сохраняют уже копию получившейся гостевой системы. Такой подход позволяет избежать процесса подбора совместимого оборудования и подстановки драйверов.

Разумеется, если ИТ-инфраструктура с самого начала основана на системе виртуализации, то в таких сложных мерах нет необходимости.

## 16.5. Тактика восстановления

После полного описания всех систем, задействованных в ИТ-бизнес-процессах, разрабатывается тактика восстановления.

Проще говоря: что восстанавливаем в первую очередь, что на втором этапе, а какие сервисы могут и обождать. На этом этапе обязательно указываются контрольные сроки и ответственные сотрудники для каждого участка. Также определяются меры, необходимые в случае, когда ИТ-служба не может обойтись собственными силами. Например, для первоначальной настройки мини-АТС потребуются пригласить стороннего специалиста.

На этом же этапе формируются требования к аппаратным ресурсам. Если у предприятия имеется резервный ЦОД (центр обработки данных), то определиться, какие сервисы дублируются изначально методом «горячего резервирования», какие подлежат перенесению или быстрому восстановлению с нуля. Как правило, руководство предприятий стремится сэкономить, и мощности резервных ЦОД форми-

руются по принципу «только критически важное». Это означает, что сразу и быстро без серьезных потерь можно восстановить только жизненно необходимые службы, чтобы продемонстрировать, что организация не умерла и по-прежнему готова выполнить взятые на себя обязательства. Другие, менее критичные элементы ИТ-бизнес-процессов планируется восстановить позднее, например после закупки дополнительного оборудования, перевода в облачные сервисы и т. д.

Разумеется, если у предприятия не существует резервного ЦОД, то все вопросы придется начинать с поиска помещения, закупки оборудования, проведения соответствующих монтажных работ, и только после этого может идти речь о восстановлении инфраструктуры.

В любом случае, имеется резервный ЦОД или нет, важную роль играет определение стоимости восстановления на каждом этапе. Для того чтобы быстро и без потерь восстановить ИТ-инфраструктуру, потребуются значительные финансовые вложения.

Денежные затраты нужны не только на аппаратные и инженерные ресурсы: закупка оборудования, ввод дополнительных электрических мощностей, модернизация системы кондиционирования, кабельные работы и многое другое, – но на оплату сверхурочных работ ИТ-подразделения, оплату приглашенного персонала (например, монтажников) и узкоспециализированных специалистов (например, тот же профессионал по настройке офисной АТС).



Руководство большинства компаний стран бывшего СССР склонно предполагать: «Наши ребята никуда не денутся, выйдут ночью, в выходные, поднажмут и все восстановят», – что в корне неверно. Очень часто после серьезных форс-мажоров, таких как пожар, наводнение, попытка рейдерского захвата, ИТ-специалисты предпочитают сменить место работы, полагая (порой весьма обоснованно), что нынешний работодатель вряд ли сможет снова вернуть былую славу и величие. Разумеется, такой исход является для начальства «как снег на голову». Сложно начинать восстановление системы с поиска квалифицированных работников, притом что далеко не многие готовы пойти работать на «пепелище». Наличие справедливой оплаты труда, в том числе и выплата сверхурочных, помогает удержать персонал от спонтанных увольнений, а также демонстрирует стабильность компании и серьезность намерений. Для партнеров и акционеров это может послужить в качестве доброго знака, показывающего, что предприятие «держится на плаву» и погибать не собирается.

После создания описания бизнес-процессов и создания общего плана восстановления необходимо создать частные планы восстановления для каждой службы и сервиса. Например, возвращая к жизни систему документооборота, может потребоваться восстановить SQL-сервер, потом сам сервер приложений и, наконец, веб-сервер для кли-

ентского доступа. В какой последовательности восстанавливать каждый элемент и какие нюансы необходимо учесть – обо всем об этом и призван рассказать частный план восстановления.

## 16.6. Рабочие документы DRP

После создания частных планов восстановления разрабатываются исполнительные листы. Исполнительные листы – это, по сути, детальная инструкция по реанимации необходимых составляющих сервиса. Например, если восстанавливается SQL-сервер для системы документооборота, то необходимо указать, где хранятся резервные копии, как они промаркированы, какая программа резервного копирования используется и т. д. Данное руководство должно быть достаточно подробным и написано понятным языком, чтобы им смог воспользоваться любой человек, имеющий даже самый небольшой опыт работы с подобными системами.

В дополнение к исполнительным листам разрабатываются контрольные тесты, необходимые для проверки правильности реанимации данного участка. Например, набор SQL-запросов, на которые должен отвечать сервер SQL, отсутствие сообщений об ошибках в журналах сервера приложений, доступность веб-сервера по протоколам HTTP и HTTPS и т. д. В случаях, когда при работе того или иного сервиса используется несколько составляющих компонентов, очень важно выполнять восстановление и проверку в порядке очередности, чтобы избежать ситуации, когда что-то не работает, а что именно – понять не удастся, и приходится заново восстанавливать по одному и смотреть, что же происходит на каждом этапе.

В некоторых случаях в дополнение к тестам разрабатывают специальные акты приема в эксплуатацию после восстановления. Обычно таким строгим подходом отличаются компании, сильно зависимые от ИТ-инфраструктуры, например банки или облачные провайдеры. В этом случае владелец сервиса, подписывая акт, подтверждает свое согласие с результатами восстановления и готовностью сервиса к работе. Подобные документы могут служить в качестве защиты ИТ-подразделения от претензий в ходе дальнейшей эксплуатации, например когда отдел разработки ПО пытается свалить вину за плохо работающее приложение на последствия сбоя после восстановления.

**База знаний** – еще одна крайне важная часть Disaster Recovery Plan. Это своего рода хранилище всевозможной информации, которая может потребоваться при восстановлении системы с нуля. Например: перечень и характеристики оборудования, какие роли исполняет тот или

иной сервер или виртуальная машина, схемы кроссировки и подключения сетевого оборудования и т. д. Нелишним будет описание дисковых подсистем хранения данных: какое хранилище данных используется, какие RAID-массивы созданы, какая модель RAID-контроллера установлена, версия прошивки и другие конструктивные особенности.

Здесь же нужно разместить контакты сторонних служб и сервисов, например службы техподдержки провайдера, телефоны сторонних консультантов и, конечно же, телефоны службы безопасности, руководства компании. Также не помешает заранее разузнать и добавить в базу знаний контактную информацию компаний, специализирующихся на ремонте сложного оборудования, восстановлении данных и других деликатных задачах.

Отдельно и в защищенном месте стоит разместить местоположение ключей шифрования, логины и пароли от всевозможнейших служб и сервисов и другую информацию секретного свойства.

Разумеется, следует избегать ситуаций вроде: «Ключ от срочной медицинской комнаты хранится в срочной медицинской комнате». Поэтому бессмысленно размещать базу знаний и другие документы DRP на серверах в составе ИТ-инфраструктуры, которая подлежит восстановлению. Лучше, если это будет защищенный ноутбук, хранящийся отдельно от здания основной ИТ-инфраструктуры. Помимо этого, копию информации имеет смысл разместить на компакт-дисках, например CD-RW или DVD-RW, также обязательно сохранить в бумажном виде на случай, если под рукой не окажется никаких рабочих компьютерных средств.

Крайне не рекомендуется хранить информацию на SSD-дисках и флэш-накопителях из-за риска потери информации при длительном хранении. Сочетание трех носителей: бумажного, CD/DVD-RW и жесткого диска на ноутбуке – в принципе, достаточно для сохранения документов из состава Disaster Recovery Plan.

## 16.7. Подготовка персонала

Можно подготовить блестящий план, но без людей, способных его исполнить, он так и останется пачкой ненужных документов.

### 16.7.1. Подготовка специальной группы

Необходимо иметь некую группу быстрого реагирования из ИТ-специалистов различного профиля, способных быстро воссоздать разрушенную ИТ-структуру хотя бы в минимальном объеме.

**Примерный состав группы восстановления:**

- специалист по системам хранения и восстановления данных;
- специалист по серверному оборудованию;
- специалист по бизнес-процессам и схемам организации использования данных;
- специалист по базам данных – DBA;
- специалист по сетевым технологиям;
- специалист по закупкам и логистике;
- специалист по технической поддержке пользователей.

Эти специалисты должны быть всегда доступны по мобильному телефону и/или другим средствам связи. Это должны быть лояльные сотрудники, материально и морально заинтересованные в скорейшем восстановлении ИТ-структуры. Также крайне желательно, чтобы они имели положительную мотивацию, например получали специальную доплату или другой компенсационный пакет.

**16.7.2. Формирование кадрового резерва**

После подготовки группы восстановления следует подумать еще и о том, что будет, если кто-нибудь из этих замечательных, хорошо обученных специалистов вдруг заболеет, уволится или просто окажется недоступен, например будучи в отпуске. Необходимо, чтобы кто-то взял на себя его функции. В этом и заключается суть кадрового резерва – освоение специалистами смежных областей. Подготовка кадрового резерва можно осуществлять по двум направлениям: горизонтальному и вертикальному.

**Горизонтальное** – когда специалисты из этой же группы осваивают зону ответственности друг друга. Например, системный администратор серверов овладевает навыками сетевого администрирования, администратор БД осваивает функции системного администратора и т. д. Преимуществом такого подхода является расширение кругозора специалистов, что дает возможность видеть проблему с нескольких точек зрения и учитывать специфику работы коллег при подготовке и осуществлении своего участка системы восстановления.

**Вертикальное** – когда младшие сотрудники перенимают опыт старших коллег. Например, оператор системного копирования учится выполнять работу администратора резервного копирования. Плюсом такого подхода является профессиональный рост специалистов младшего звена, что может пригодиться, в том числе и в плане карьерного роста.

## 16.8. Тестирование восстановления

Любой замечательный план нуждается в проверке на практике. И Disaster Recovery Plan – отнюдь не исключение.

Ситуация отягощается тем обстоятельством, что далеко не каждый готов уничтожить работающую систему, чтобы проверить, как эффективно она может быть восстановлена. Для того чтобы, с одной стороны, следовать принципу «не навреди», а с другой – хотя бы частично убедиться в правильности и эффективности стратегии и тактики восстановления, используют различные методики, о которых речь пойдет ниже.

### 16.8.1. Условное, или «бумажное», тестирование

Такой вид проверки скорее напоминает деловую игру, нежели учения, приближенные к боевым условиям. В назначенный день и час объявляется, что система потерпела полный крах и необходимо провести полное восстановление. Участники группы извлекают из хранилища документацию (мы же помним, что она должна храниться минимум в 3 различных местах и видах: бумажная копия, компакт-диски и защищенный ноутбук) и начинают «работу». Задача каждого члена группы – подробно описать свои действия по решению поставленной задачи и указать примерное время, которое ему понадобится.

Несмотря на «игрушечное» впечатление, которое производит данный вид проверки, огромное количество подразделений не проходит даже этого этапа. Ошибки бывают самые разные: от отсутствия информации о местонахождении документации до непонимания, как восстановить систему из резервной копии. Бумажное тестирование позволяет выявить подобные промахи и вовремя устранить, пока не случилось горе.

### 16.8.2. Проверка на тестовом стенде

Это самый популярный вид проверки, несмотря на то что для его проведения необходимо выделить определенные физические ресурсы. Восстановление системы производится фрагментарно, обеспечения полной работоспособности в условиях штатной нагрузки при таком подходе не требуется, поэтому для проведения подобного тестирования подойдет недорогое или устаревшее оборудование. Главное условие – отсутствие сбоев оборудования и требуемое количество свободного пространства для тестирования восстановления данных. При правильном подходе таким образом можно проверить возможность

восстановления практически всех компонентов инфраструктуры, при этом не затрагивая работы основных бизнес-процессов.

Одно из главных требований к проведению подобного рода проверок – тщательное документирование результатов и полученных характеристик, например: время восстановления, объем данных, причина, по которой не удалось восстановить, и т. д. В дальнейшем методом экстраполяции это поможет получить примерные величины для учета реальных временных и материальных затрат.

### 16.6.3. Учения на production-системе

Как я уже писал выше, мало кто рискнет выполнять полное тестирование аварийного восстановления на работающей ИТ-инфраструктуре. Чаще всего в таких случаях речь идет о проверке на резервном ЦОД с кратковременным переключением после восстановления. При этом основная система остается без изменений.

Во время переключения обычно временно блокируют возможность работы с тестовой ИТ-инфраструктурой в обычном режиме. Иначе велика вероятность того, что во время переключения в информационную составляющую тестовой среды будут внесены изменения, которые впоследствии пропадут при переходе обратно на старую работающую систему.

Разумеется, о проводимых учениях должны быть предупреждены не только все участники процесса, но и рядовые пользователи. Это также необходимо, чтобы свести на нет или значительно минимизировать возможные расхождения, например если оператор базы данных все-таки сумеет добавить запись в базу данных во время переключения на ЦОД2.

В принципе, после прохождения всех трех проверок можно считать, что ИТ-инфраструктура и коллектив специалистов, отвечающий за ее работу, готов выполнить Disaster Recovery. «В принципе» – потому, что на практике часто встречаются сюрпризы, которых нельзя учесть при самом тщательном тестировании.

Чтобы максимально перекрыть возможные недочеты, проводят не только прямое тестирование «как написано», но и с некоторыми расхождениями. Например, восстановление при отсутствии администратора резервного копирования. Или восстановление при узком канале связи и многое другое, что не вписывается в обычные представления об идеальном процессе.

Чем больше проведено учений и чем больше учтено нюансов – тем больше вероятность благополучного результата в реальных условиях.

Суворовский принцип «Тяжело в учении – легко в бою» здесь подходит как нельзя кстати.

Только не стоит забывать о разумных мерах предосторожности, особенно при проведении учений в условиях, приближенных к боевым. В этом случае не исключено проникающее влияние ошибок, допущенных при тестировании, на работу основной системы. И в любом случае дополнительная резервная копия никогда не повредит.

## 16.7. Заключение

Создание плана восстановления – непростая, но крайне важная задача. Без соответствующей документации восстановить систему, даже имея полный набор резервных копий, резервный ЦОД и еще множество хороших вещей, бывает очень непросто. Разумеется, ни в коем случае нельзя забывать о бизнес-процессах и нуждах предприятия. Поэтому конечное слово при реализации подобной стратегии на местах всегда остается за исполнителем.

Лучше всего справляется с неприятностями тот, кто сумел заранее подготовиться.

# Вопросы к четвертой части для закрепления материала

1. Что означают термины «надкусывание» и «дожигание»?
2. Перечислите основные этапы проектирования ИТ-систем.
3. Опишите основные методы обеспечения безопасности при резервном копировании.
4. Для чего используется разделение ролей при резервном копировании?
5. Для чего нужно писать Disaster Recovery plan?
6. Какие основные части Disaster Recovery plan вы знаете?
7. Почему нужно проводить учения по восстановлению?

Часть **V**



**Суровый  
практический  
ПОДХОД**

# Глава 17

---

## «То, что нас не убивает, делает нас сильнее». Ошибки при создании отказоустойчивых систем

*У польского короля Яна Собеского была сабля с надписью: «Берегись неверных друзей, а от врагов я тебя защищу».*

### 17.1. Предисловие

Часто бывает, что одного знания «как надо» недостаточно, чтобы уберечь себя от беды. Всегда лучше учиться на чужих ошибках, чем на своих собственных. И в этом случае живые примеры «как делать не надо» бывают просто незаменимы. В этой главе читатель познакомится с реальными ситуациями, происходившими с разными людьми и в разное время. Некоторые детали были намеренно искажены для сохранения инкогнито участников данного процесса.

## 17.2. Наиболее часто встречающиеся ошибки

### 17.2.1. Игнорирование проблемы роста объемов данных при закупке оборудования

Такую ошибку обычно допускают на стадии проектирования. Стремясь сэкономить некоторую сумму при закупке, параметры оборудования выбирают, что называется, «впритык», только-только чтобы удовлетворить сиюминутные нужды. Выше я уже писал о необходимости предусмотреть возможность масштабирования. Но иногда стремление сэкономить приводит к тому, что в расчет берутся только сиюминутные текущие нужды, без учета будущих изменений.

Приведу пример: при создании системы предполагалось, что для еженедельного полного копирования хватит съемного картриджа ЛТО3, а для промежуточной копии – ЛТО2. Поэтому был закуплен накопитель ЛТО3 с соответствующим набором кассет. Но за полгода прирост данных составил порядка 20% от заявленного объема. В итоге емкость существующих накопителей перестала удовлетворять. Возник выбор: приобрести еще один комплект носителей или новый накопитель, поддерживающий формат ЛТО4, и компенсировать недостаток объема дополнительной покупкой картриджей ЛТО4. В итоге было принято решение пойти по первому пути как более экономичному на тот момент. Но еще через полгода объем данных возрос еще на 30%, и возникли проблемы с пропускной способностью, процесс полного резервного копирования перестал укладываться в установленные временные рамки. Так как накопитель был напрямую подключен к файл-серверу, на котором и концентрировалась большая часть пользовательских данных, то узким местом в схеме оказалась... скорость записи накопителя ЛТО3. И тогда все равно пришлось приобрести накопитель ЛТО4 с полным набором картриджей ЛТО4. Соответственно, купленные ранее картриджи ЛТО2 можно было использовать только для чтения, а картриджи ЛТО3 – только для промежуточного бэкапа.

Разумеется, можно было избежать ненужных трат, если сразу проектировать систему и закупать оборудование с расчетом на перспективу. Оптимальный вариант: приобретать оборудование и расходные материалы с расчетом на удвоение объема данных в течение периода от одного года до двух лет.

### 17.2.2. Отсутствие регулярной проверки резервных копий

Эта ошибка наиболее часто встречается у начинающих системных администраторов. Суть проблемы заключается в том, что даже при полной автоматизации процесса резервного копирования необходим постоянный контроль резервных копий. Не секрет, что любой процесс не гарантирован от случайных сбоев. Например, может оказаться нерабочим ЛТО-картридж. Или в связи с плохой работой накопителя лента не будет качественно записана. В идеале необходимо провести полное восстановление содержимого носителя на идентичную тестовую инфраструктуру и сравнение по контрольным суммам, работоспособности сервисов и т. д. Но далеко не всегда бизнес готов поддерживать такую сложную и дорогостоящую процедуру. Чаще всего приходится ограничиваться автоматической проверкой контрольных сумм (такая возможность заложена во многих программах резервного копирования) при создании копии и восстановлении нескольких файлов в другое местоположение с целью проверки целостности архива.

Но тем не менее такие тесты должны производиться регулярно. Например, при реализации месячной схемы «дед–отец–сын» проверка должны подвергаться все «деды» (ежемесячные резервные копии).

Также полезно время от времени тестировать резервные копии, находящиеся на хранении. Дело в том, что носители могут утрачивать свою работоспособность. Ведь магнитная лента подвержена размагничиванию, файловая система на дисковом хранилище может содержать ошибки. Регулярные проверки позволят вовремя определить проблемы и своевременно их устранить. Не забываем, что архив резервных копий – это своего рода история компании, а к истории нужно относиться бережно.

### 17.2.3. Отсутствие контроля за копируемым контентом

Еще одна частая ошибка. Данные имеют тенденцию постоянно меняться и, как правило, увеличиваться в объеме. Постепенное наращивание объемов обрабатываемой информации характерно для развития большинства компаний. Через какое-то время обнаруживается, что существующая система резервного копирования не справляется с возложенными на нее задачами. Гигабайты файлов и баз данных

перестают помещаться на носителях. Особенно это характерно для дисковых хранилищ. В отличие от ленточных накопителей и библиотек, куда без труда можно вставить чистый картридж и продолжить запись, с дисковым RAID-массивом такой номер так легко провести не получится.

Помимо нехватки объема, остро встает и нехватка времени для выполнения заданий, которые не успевают выполняться в отведенный период («окно бэкапа»). Приходится задумываться о приобретении нового дорогостоящего оборудования для кардинальной перестройки всей системы резервного копирования.

Но есть и хорошая новость: данную проблему можно обойти или хотя бы отстрочить. Выход прост: нужно время от времени подвергать тщательному анализу данные, подлежащие резервированию. При этом рекомендуется запастись приличной долей скепсиса.

Например, безжалостно исключить из резервного копирования все мультимедийные файлы развлекательного характера, личные фотографии и все остальное, что так дорого сердцу пользователя, но не задействовано в бизнес-процессах. Все это довольно часто можно обнаружить на общих ресурсах и в личных каталогах пользователей.

Файлы, к которым не обращались в течение длительного периода, имеет смысл перенести на отдельный съемный носитель и отправить в архив для хранения. Этот же рецепт подходит для оптимизации почтовых хранилищ, а также баз данных.

Эти простые меры позволят не только значительно разгрузить систему резервного копирования, но и навести порядок на серверных ресурсах, повысив быстродействие.

#### **17.2.4. Слишком долгий период между полным резервным копированием**

Суть ошибки такова. В целях экономии выполняется полное резервное копирование один раз в течение довольно длительного периода – допустим, раз в месяц. А для поддержания актуальности сохраняемых данных служит множество инкрементных или дифференциальных копий. Обычно такую странную схему создают из экономии средств на носители или для сокращения времени использования окна бэкапа. Все бы ничего, но при восстановлении приходится задействовать большое число инкрементных копий, что само по себе может значительно увеличить время «реанимации», а следовательно, удлинится пауза в работе бизнес-процессов.

Некоторые системы резервного копирования, например Acronis True Image, при повреждении одной из инкрементных копий перестают «видеть» и все последующие, созданные уже после поврежденной. Но самое страшное в этом случае – это утрата единственной полной копии за требуемый период. Случись это с еженедельным бэкапом – бизнес потеряет информацию за неделю, кроме того, возможно, удастся восстановить какие-то данные из инкрементных копий. Но в приведенном случае теряется вся работа за очень долгий период. Поэтому пользы от такого резервного копирования практически никакой.

### **17.2.5. Слишком много полных копий**

Это другая крайность, по сравнению с предыдущей ситуацией. Допустим, у администратора хватает и места на диске или ленте, и временного промежутка, чтобы выполнять полное резервное копирование. И он с удовольствием этот факт использует. Все бы ничего, но объем данных со временем обычно увеличивается.

Растет и бизнес, появляются новые серверы в инфраструктуре, уплотняется рабочий график, и все это неизменно ведет к тому, что для выполнения постоянного полного резервного копирования не хватает ни места, ни временного отрезка в «окне бэкапа». Я уже упоминал выше о необходимости постоянного мониторинга объемов информации, подлежащей сохранению. В данном случае объем исходных данных может быть не очень большим, но забивать носители для хранения резервных копий множеством мало отличающихся друг от друга версий вряд ли оправдано. Гораздо разумнее организовать инкрементное или дифференциальное резервное копирование, а еще лучше сделать это сразу, чтобы не изменять уже настроенную и работающую систему «на ходу».

### **17.2.6. Много копий на одном носителе**

Эта ошибка характерна для небольших компаний, где объем копируемых данных весьма незначительный. Возникает соблазн вместо ротации нескольких сравнительно дорогих носителей все записывать на меньшее число кассет. Проблема заключается в том, что при утрате одного, ставшего уже драгоценным носителя (например, при размагничивании, механическом повреждении, ошибках файловой системы на дисковом томе и т. д.) теряются резервные копии за весьма длительный период.

## 17.3. Реальные примеры того, как нельзя делать резервное копирование

В основе этого повествования, местами грустного, местами забавного, лежат реальные события и факты. Мне приходилось лично наблюдать ошибки системных администраторов, как начинающих, так и уже имеющих опыт, и эти ошибки приводили порой к довольно тяжелым последствиям.



Все описанные случаи происходили на самом деле. Никаких имен не сообщается, и некоторые мелкие детали сознательно изменены в целях анонимности. Но в целом эти прецеденты могут послужить хорошим уроком или просто в качестве средства проведения досуга.

### 17.3.1. «Надежное железо», или Как сохранить резервную копию на том же диске

Мне довелось работать в одной организации, которая, несмотря на то что работала в прибыльном рыночном сегменте, отличалась крайней небрежностью в области информационных технологий. Обязанности системного администратора исполнял тихий вежливый молодой человек, который почему-то считал, что главная обязанность ИТ-специалиста – с утра до вечера помогать бухгалтерам в их нелегком труде. Для других дел у него попросту не находилось ни времени, ни сил, ни желания.

Мои функции носили скорее консультативный характер, а хороший консультант, прежде чем дать рекомендацию, должен выяснить все обстоятельства. Но на невинный вопрос: «А как выполняется резервное копирование?» – я получил просто удивительнейший ответ:

– А у нас «железо» очень надежное, нам резервное копирование не нужно!

Этим «надежным железом» оказались старенький сервер Supermicro, который одновременно играл роль почтового и файл-сервера, а также контроллера домена Active Directory.

Сказать, что я был слегка шокирован, – это не сказать ничего. Но «законы Мерфи», как и другие законы окружающего мира, например физики или экономики, опираются на реальное положение вещей. И результат такого пренебрежительного отношения к резервному копированию не заставил себя ждать – через пару дней сервер перестал загружаться. Причина проста – авария жесткого диска. Как раз

такой случай, когда никто не виноват, но восстанавливать работу как-то нужно.

Надо сказать, что в качестве средства «высокой надежности» использовался RAID-5, который включал в себя один-единственный массив, содержащий все жесткие диски сервера. Средствами Windows он делился на два логических диска: «C:» и «D:» И этот самый RAID-массив после сбоя диска никак не переходил в состояние «Degraded», то есть в режим обеспечения доступности за счет оставшихся дисков и контрольных сумм. Не помогли ни перезагрузка, ни временное выключение с выдергиванием кабелей питания. Все диски видит, а в «Degraded» не переходит, выдает сбой – и хоть плачь.

Срочно был созван военный совет из начальника ИТ-отдела, системного администратора и других специалистов. И первый же вопрос, который прозвучал на повестке дня:

– *А за какое число у нас сделан бэкап?*

Молодой человек побелел, потом покраснел и дрожащим голосом тихо сказал:

– *Я же как-то говорил Вам, что у нас для резервных копий места нет, помните?*

Руководитель ИТ-отдела был на редкость выдержанный и воспитанный человек. Тем более он совсем недавно заступил на эту должность. Поэтому он не стал устраивать скандала. Вместо этого он тихо и вкрадчиво произнес:

– *Что, совсем ничего не сохраняется?*

– *Копия контроллера AD*, – ответил сисадмин.

На лице шефа от ИТ отразилась мысль вроде: «Ладно, как-нибудь отобьемся, хотя бы новый домен делать не придется...»

– *А где это лежит?*

– *На диске D*, – радостно ответил системный администратор.

– *На каком компьютере?* – недоверчиво спросил начальник.

– *На том же сервере!* – бодро отрапортовал молодой сисадмин.

«Занавес!» – как говорят драматурги.

Сервер мы все же восстановили. Для этого почему-то пришлось «походить» по меню консоли RAID-контроллера, а потом сохранить параметры и выйти. И он перешел в режим «Degraded», далее быстренько сняли резервную копию двумя способами подряд: через встроенную утилиту ntbackup и простым копированием. На самом деле на складе лежала в запасе пара исправных рабочих станций с довольно большими по объему дисками, поэтому вопрос «Куда сделать бэкап?» в принципе не стоял. В общем, все хорошо, что хорошо кончается.

Какие выводы можно сделать из этого случая?

Источником всех бед является неправильное понимание роли ИТ-службы в современном бизнесе. Эта служба призвана не «помогать пользователям», а обеспечивать бесперебойное функционирование информационных систем.

Никогда нельзя замалчивать проблемы. Принцип «подальше от начальства, поближе к кухне» в сфере информационных технологий работает с точностью до наоборот. Поэтому если нет возможности организовать нормальное резервное копирование в силу материальных причин: нет соответствующего оборудования, программного обеспечения и т. д., – нужно, что называется, «звонить во все колокола», пока проблема не будет окончательно снята. При этом устные высказывания вроде «когда-то что-то сказал», как правило, не работают. Все заявки на покупку оборудования, выделения времени для работ нужно подавать исключительно в письменном виде.

В данном примере таких проблем не наблюдалось вовсе. Руководство компании, в принципе, было не против выделить деньги на резервные копии. Как выяснилось впоследствии, некоторые сотрудники даже сами на свои деньги покупали съемные носители и сохраняли на них какие-то рабочие данные, оставляя тем самым широкую брешь в системе безопасности. Единственное, что останавливало, – нежелание самого системного администратора приложить некоторые усилия к решению данной проблемы и отсутствие контроля «сверху».

Ну и, конечно, нельзя допускать анекдотичной ситуации из серии «ключ от срочной медицинской комнаты хранится в срочной медицинской комнате». Резервные копии должны быть размещены таким образом, чтобы при аварии к ним можно было без особых трудов получить доступ. И уж точно ни в коем случае не размещать копии на тех же ресурсах и оборудовании, данные которых они же и содержат.



Бывает, что человек, отвечающий за распределение финансов, прямо отказывается закупать что-то необходимое или применяет тактику молчания, попросту тянет время, иногда мотивируя свой ответ чем-то вроде: «Вот когда что-то случится, тогда и будем что-то решать». Чтобы этого избежать, необходимо обязательно просить завизировать соответствующий документ, например просить написать: «Согласен» или «Отказать» – и поставить подпись.

Но что делать, если такой босс оставляет бумагу у себя со словами: «Я посмотрю...» – и все остается на том же месте? Значит, через короткий промежуток времени нужно снова принести такие же документы на подпись, а потом еще и еще раз, пока не будет получен результат. Иногда полезно об-

ратиться к более вышестоящему руководству вплоть до учредителей компании. Готовых рецептов для таких ситуаций быть не может, поэтому нужно действовать с учетом специфики предприятия. Но всегда ксерокопии всех докладных, служебных записок, счетов и коммерческих предложений нужно оставлять у себя на случай несправедливых обвинений.

### **17.3.2. «Спешка важна при ловле блох».**

#### **Как не сделать бэкап, когда все копируется на один сервер**

Эта история произошла в довольно крупной компании, имеющей офисы в нескольких местах. И что удивительно, хватало и времени, средств, и ресурсов, и квалификации специалистов.

Резервное копирование было построено следующим образом: в качестве основного ПО резервного копирования использовалась неплохая программа CA ARCserve. Очень хорошее и надежное средство, которое на тот момент имело ряд преимуществ, том числе в плане цены и быстродействия. Она использует централизованную топологию и через соответствующие программы-агенты собирает данные с защищаемых ресурсов, чтобы впоследствии записать на ленточные носители или в файлы-контейнеры.

В качестве основной системы хранения использовалась ленточная библиотека фирмы Quantum с шестью накопителями на магнитной ленте стандарта LTO2. А подключалась эта чудо-система к серверу на MS Windows 2003. Роль сервера исполняла ничем не выдающаяся «железка» HP Proliant DL360 G4, старенькая, но вполне рабочая.

Основная проблема заключалась в следующем. Для сбора данных со всей весьма немаленькой инфраструктуры было создано одноединственное задание резервного копирования, и все данные собирались на один-единственный сервер с ленточной библиотекой. Надо ли говорить, что в связи с довольно богатой историей компании в ее составе работали не только самые современные, но и весьма устаревшие «динозавры». Свои данные системе резервного копирования они отдавали долго и нехотя, подключенные по большей части через FastEthernet 100 Mbps, процесс создания резервных копий превращался в долгий и мучительный процесс. Full Backup начинался в пятницу в 18.00 сразу после окончания рабочего дня и заканчивался в понедельник около 10 часов утра. При этом если по какой-то причине происходил сбой, то всю неделю компания жила со старым бэкапом, в надежде, что черные силы пройдут стороной. Руководство компании ценило производительность работы сотруд-

ников гораздо выше безопасности работы ИТ-инфраструктуры и не разрешало нагружать систему таким «бесполезным занятием», как резервная копия.

Часто беда приходит со стороны, но и человек – сам кузнец своего счастья. Пользователи стали жаловаться на замедления в работе электронной почты, на доставку писем с большим опозданием. Старенький сервер с MS Exchange 2003 еле-еле копошился в раздутой почтовой базе, в которой каждый пользователь сохранял не только письма, но и объемные файлы-вложения. Молодой руководитель ИТ-отдела принял замечательное решение: выполнить дефрагментацию почтовой базы данных. Это вполне рабочая процедура, позволяющая хоть как-то ускорить процесс обработки.

И так как работа бизнеса – это святое, он ни в какую не соглашался провести эту операцию в другое время, кроме выходных. Как раз в то самое время, когда должен выполняться бэкап.

Уговоры, протесты и даже мольба системного администратора не возымели на молодого шефа никакого воздействия. Он отдает команду, под свою личную ответственность, выполнять дефрагментацию, не дожидаясь создания резервной копии. Зная, что этот процесс может занять долгие часы, он так торопился, что отказался даже отсоединить и сделать средствами копирования дубликат базы на другом сервере и работать с ним. В итоге все работы велись с той же самой базой на том же самом сервере. Результат нетрудно предугадать – по не установленной до конца причине происходит сбой, после которого почтовая база повреждена, процесс резервного копирования не был доведен до конца, и резервная копия еще не была создана.

Но это еще не все. Два предыдущих бэкапа также потерпели фиаско. В итоге организация восстановила данные из совсем старой копии месячной давности. Актуальная переписка была потеряна. Что-то удалось восстановить из кэша почтовых программ-клиентов. Большая часть переписки за последний месяц была утрачена.

Подводя итог этой печальной истории, можно отметить несколько ключевых моментов.

Не нужно допускать такой абсолютной централизации. Если есть проблемы с попаданием в «окно бэкапа», гораздо лучше разбить один большой процесс на несколько маленьких, для диверсификации резервного копирования и возможности перезапустить оборвавшийся процесс впоследствии.

При росте инфраструктуры нужно применять шаги по распараллеливанию потоков резервного копирования на разные приемники.

Вместо одной большой библиотеки лучше иметь несколько маленьких устройств хранения, подключенных к разным серверам, чтобы ускорить процесс создания копий.

И никогда не проводите сервисные работы над той же самой базой. Все мало-мальски серьезные модификации всегда выполняйте с копией, которую при успешном результате можно подключить на место основного экземпляра.

### **17.3.3. Чем заканчиваются попытки заменить резервное копирование синхронизацией**

Проводя аудит в одной компании, я столкнулся с одним весьма занятным изобретением. Назвать это системой резервного копирования просто язык не поворачивается.

Итак, к файл-серверу был подключен по USB небольшой дисковый массив RAID-0 для домашнего использования из двух дисков. Он же был объявлен в качестве публичного сетевого ресурса. Далее на каждом сервере в назначенный час запускалась команда gobosoru, которая синхронизировала копию данных на вышеописанной сетевой папке. При этом если файл был удален на исходном ресурсе, то он также удалялся из копии.

Судя по всему, таким образом решался вопрос роста объема резервной копии, которая просто не могла быть больше, чем совокупный размер исходных файлов.

Что сразу бросалось в глаза, так это общая ненадежность конструкции. Во-первых, USB-соединение время от времени отваливалось. Если сисадмин замечал это, он шел в серверную, вручную выдергивал и заново подключал USB-кабель. Если такая проблема происходила в неурочный час, например по выходным или ночью, то запущенная синхронизация не выполнялась.

Если пользователь по ошибке удалял файл и сразу сообщал об этом, то шанс на восстановление из вчерашней копии еще оставался. Если системного администратора не было на месте или пользователь обнаруживал «недостачу» файла слишком поздно, ничем помочь уже было нельзя.

Далее стоит привести слова администратора баз данных, работавшего с 1С:

«Пока ничего не сбоило, все было нормально. Но однажды случилась неприятность, и, чтобы восстановить работу, нашей бухгалтерии понадобилась копия за прошлые даты. Конечно, взять было не отку-

да. Мы просили любую копию, хотя бы с начала года – но все тщетно. В итоге заново проводили вручную весь период».

Вывод из этой печальной истории напрашивается сам собой. Нельзя путать репликацию с системой резервного копирования. И уж точно нельзя отказываться от архивного копирования. Копии данных за прошлые периоды могут понадобиться по множеству причин: проверка со стороны СБ, ошибка пользователя, скрытый сбой, последствия которого были обнаружены через некоторое время, и многое другое. Иначе можно остаться «у разбитого корыта», точнее с одной-единственной резервной копией, содержащей ошибочные данные.

Само собой разумеется, нельзя держать резервные копии на таком ненадежном устройстве. Какой смысл вообще делать резервную копию на массив RAID-0, который является более уязвимым, чем одиночный жесткий диск. И уж точно не надо подключать устройства для хранения копий по постоянно «отваливающемуся» соединению.

#### 17.3.4. Праздничная гирлянда снимотов

История эта случилась у одного из крупных системных интеграторов.

Ко мне обратился сетевой инженер с просьбой помочь ему вернуть к жизни виртуальную машину, отвечающую за статистику.

Система виртуализации VMware ESXi сама по себе довольно надежна. Но, открыв консоль, я с изумлением обнаружил, что данная виртуалка имеет как минимум пятнадцать моментальных снимков, следующих один за другим.

Первая мысль была о том, что это напакостила система резервного копирования. В главе 12 мы как раз говорили о том, что при копировании виртуальных машин используются снимоты. Иногда программа системного копирования «забывает» удалить за собой снимок, и может накопиться несколько снимотов подряд. Тогда ощутимо снижается общее быстродействие, и системный администратор вынужден вручную наводить порядок в системе, удаляя лишние снимки.

Но, как оказалось, всю эту «вереницу» создал... сам сетевой инженер. Человек искренне считал, что моментальный снимок – это и есть резервная копия. При этом его нисколько не смущал тот факт, что все это «хозяйство» хранится на том же томе, что и основная виртуальная машина.

Разумеется, когда снимков накопилось изрядное количество, виртуальная система уже не смогла поддерживать эту длинную «гирлянду», что вызвало сбой с последующим разрушением файловой системы на виртуальном диске.

В итоге все закончилось не так уж плохо. Откат на несколько снимков назад (разумеется, с потерей данных за соответствующий период) позволил загрузить виртуальную машину. Для удаления снимков я выполнил экспорт виртуальной машины в контейнер (OVF) и развернул под другим именем.

Так как приобретать нормальную систему резервного копирования никто не планировал, я просто научил инженера вместо создания моментальных снимков выполнять экспорт данных в файл и сохранять его на другом физическом компьютере. Хотя без нормального бэкапа все равно плохо.

### 17.3.5. Ночной кошмар без резервной копии

Работал в одной компании в крупном ИТ-подразделении один молодой системный администратор. И так уж случилось, что он был вынужден по личным обстоятельствам часто отпрашиваться с работы. Причина была очень уважительной, и его отпускали безоговорочно.

Но его мучил комплекс вины. Ведь пока он отсутствует, его коллеги вынуждены выполнять его работу. И он решил сделать им приятный сюрприз.

В составе инфраструктуры работал сервер, на котором был запущен MS SQL. На дисковом массиве сервера была оставлено неразмеченное пространство «на всякий случай». Но время шло, базы данных разрастались, и места на логическом диске стало не хватать. Его можно было расширить, используя неразмеченное пространство, но все «руки не доходили». И каждый раз кто-то постоянно жаловался на нехватку места.

Однажды вечером наш герой задержался на работе. Дождавшись, когда все ушли, он приступил к расширению диска. При помощи программы Acronis True Image он сделал снимок системы. Если бы он подготовился заранее и почитал про различные методы решения данной задачи – он мог бы просто использовать утилиту diskpart и закончить работу с массивом. Но молодой сисадмин был полон уверенности в достаточности своих знаний и решил действовать «в лоб». Поэтому он удалил все разделы, зачем-то создал новые и хотел уже было восстановить обратно данные из образа, но... сделал ошибку, снова запустив копирование, и перезаписал уже существующий образ. Сказались стресс и усталость, в общем, такие случаи хоть и происходят нечасто, но могут случиться с каждым из нас.

Он бросился к полкам с лентами, на которых хранились резервные копии. Система резервного копирования была полностью в зоне его

ответственности, но он в силу личных обстоятельств за ней не следил. Как выяснилось позже, резервные копии не выполнялись уже весьма продолжительное время.

Остаток ночи он провел за тщетными попытками исправить ситуацию. Это была последняя и самая роковая ошибка. В панике он пытался применить то один способ восстановления, то другой и тем самым уничтожил последние шансы восстановить разделы на диске и затертый файл образа диска.

Какие-то данные потом восстановили частично из совсем старых резервных копий, какие-то – из таблиц других баз данных с других серверов.

Какие же ошибки совершил системный администратор?

Первый и самый страшный просчет – спонтанное выполнение несогласованных работ. Всех неприятностей можно было бы избежать, если заранее проработать теоретическую и практическую часть работ, выполнить тестирование на стенде и только потом приступать к активным действиям на промышленном оборудовании.

При осуществлении таких критических действий, помимо штатного резервного копирования, обязательно нужно иметь как минимум две копии, два образа системы. Желательно сделанные разными программами и обязательно размещенные на разных ресурсах. Ведь сбой может случиться не только на «подопытном» сервере, но и на оборудовании, где хранится образ системы.

Ни в коем случае нельзя утаивать проблемы. При любом раскладе нужно иметь мужество заявить о проблеме. Умение признавать ошибки, порой самые тяжелые – отличительная черта профессионала. Если бы наш герой не пытался в панике что-то исправлять, а позвонил своим коллегам, пусть даже среди ночи, или оставил все как есть до утра и честно рассказал обо всех злоключениях, это увеличило бы шансы на более успешный исход событий.

И последнее. Что это за безобразие – не следить за резервным копированием?! Если не получается уследить самому – нужно поручить эту миссию другому сотруднику. Обязательный контроль, включая тестовые проверки, – это то, что должно быть всегда.

### **17.3.6. Зато у вас антивирус надежный!**

Эта забавная история произошла несколько лет назад.

Проводя штатное тестирование резервных копий, я обнаружил, что большая часть ночных заданий full-backup не выполняется. При

этом задания, в которых фигурируют небольшие объемы данных, по большей части отрабатывают нормально.

После некоторых изысканий было установлено, что виной всех бед является антивирусный монитор от одного из известных российских производителей. Когда программа резервного копирования запускает мощный процесс передачи данных, антивирус не в состоянии пропустить через себя такие объемы информации и просто отключает процесс копирования. Никакие «танцы с бубном» не принесли результата, и, влекомый искрой последней надежды, я обратился в техническую поддержку производителя антивирусного продукта.

Ответ сотрудника техподдержки был просто убийственным:

– *А зачем вам резервная копия? У вас же антивирус очень хороший!*

Я не счел нужным продолжать этот разговор.

Резервное копирование я все-таки настроил. Для этого пришлось отключить антивирусный монитор на серверах и использовать только сканер. Обидно, компания заплатила немалые деньги за это программное обеспечение. Выручил бесплатный Comodo Antivirus, чья лицензия в те старые добрые времена позволяла использовать его в том числе и на серверах.

## 17.4. Заключение

Как уже говорилось выше, все случаи, описанные в моей книге, имеют под собой реальную подоплеку. Хочется верить, что эти «крупницы бесценного опыта» помогут ИТ-специалистам избежать ошибок в организации резервного копирования.

# Глава 18

## Инструменты снятия образов системы для критичных случаев

*Если пользователю не говорить, что pop-Windows-системы – это сложно, он про это никогда и не подумает. В конце концов, ему все равно, какую систему не знать...*

### 18.1. Предисловие

Ранее читатель уже встречался с понятиями «блочный бэкап» и «система снятия образа». По сути, это означает одно и то же – возможность производить резервное копирование специальными средствами, минуя файловую систему. При таком подходе можно сделать слепок системы «как есть» в специальный файл, который называется «образ». И снятие информации происходит гораздо быстрее, чем при копировании каждого файла.

Такой инструмент идеально подходит, например, для ситуации, когда существует риск повредить информацию. Допустим, повреждена файловая система, и ее использование может погубить все данные. Или программное обеспечение компьютера заражено вирусом, и требуется срочно сделать копию, пока вирус не разрушил всех данных. Еще одна ситуация, при которой применение подобных систем практически не имеет альтернативы, – когда жесткий диск может выйти из строя в кратчайшие сроки и срочно нужна полная резервная копия.

Снятие образов диска – это один из излюбленных приемов службы безопасности, чтобы в спокойной обстановке исследовать файлы на выбранном компьютере.

Также системы снятия образов – незаменимый инструмент при внесении серьезных изменений в работу серверного и другого оборудования. Например, при обновлении программного обеспечения на критичных серверах очень важно иметь возможность откатить систему назад в случае неудачи. Конечно, штатное резервное копирование и встроенные средства являются неплохим решением, когда нет ничего другого. Но возможность снять полный слепок и потом вернуть все, как есть, – с этим не сравнится ни то, ни другое.

Системы снятия образов подразделяются на два типа: on-line, позволяющие сделать образ работы прямо во время работы операционной системы и другого ПО, и off-line – когда компьютер работает под управлением другой операционной системы, чаще всего на базе LiveCD- или LiveUSB-дистрибутива.

Среди систем on-line в качестве примера можно привести коммерческий продукт Acronis Backup & Restore и бесплатный Comodo Backup.

Off-line позволяют выполнить создание образа в наиболее безопасных условиях, когда работа основной системы не может повлиять на результат сохраняемых данных. Среди других выделяются дистрибутивы на базе продуктов open-source, такие как Clonezilla Backup, PING, и Redo Backup & Restore. Последний мне показался наиболее удобным для быстрого применения, а также для освоения даже начинающими специалистами.

## 18.2. Краткое описание дистрибутива Redo Backup and Recovery

Рассматриваемый нами Redo Backup and Recovery в сухом остатке, по сути, представляет собой набор сценариев на Perl.

Интерфейс приложения построен на библиотеке GTK2+. Основой для загрузочного Live CD послужил Ubuntu Linux с быстрым и симпатичным оконным менеджером OpenBox. Собственно, сам сценарий может работать практически на любом Linux-дистрибутиве, если в системе есть необходимые программы, включая драйверы для доступа к дискам. Поэтому при достаточном желании и уровне подготовки можно создать такую систему блочного копирования и даже организовать на ее основе децентрализованную схему резервного копирования для малого и среднего бизнеса. Тем более что исходники находятся в свободном доступе.

Распространяется это чудо программистской мысли под лицензией GPLv3.



Все перечисленные выше Live CD-дистрибутивы Clonezilla Backup, PING, и Redo Backup & Restore являются графической оболочкой для утилиты командной строки PartClone. В принципе, можно использовать эту программу напрямую, но в готовом дистрибутиве решены многие другие задачи, например наличие в системе необходимого набора драйверов, удобная загрузка и т. д.

Несмотря на кажущуюся простоту, находка обладает массой достоинств, среди которых:

- удобный интерфейс,
- не требует установки, запускается с Live CD или USB-носителя;
- поддерживает множество файловых систем как в Windows, так и в Linux;
- имеет встроенный DHCP-клиент. После подключения к сети самостоятельно получает сетевые настройки;
- располагает набором дополнительных средств для различных задач: от поиска информации в Интернете до манипуляции с дисковыми разделами.

При этом сам ISO-образ имеет относительно небольшой размер – всего 250 Мб.

## 18.3. Принципы работы

Все очень просто. Нужно дождаться загрузки с Live CD и нажать предусмотренные несколько кнопок. Система использует технологию блочного копирования, позволяющего «сгребать» данные с дисковых разделов большими кусками, минуя файловую систему.

Далее мы рассмотрим конкретный пример: сначала выполним резервную копию, а потом восстановимся из нее в новом окружении.

### *О тестовом стенде*

Все работы выполнялись на тестовой виртуальной машине. В качестве операционной системы используется ознакомительная версия Windows Server 2012 R2. В качестве хранилища создаваемых образов использовался компьютер под управлением Linux и Samba для придания ему функций файл-сервера.

### 18.3.1. Процесс снятия образа

Перед началом загрузки появляется окно с предложением:

- выполнить загрузку;

- загрузиться в Safe Mode;
- проверить диск;
- выполнить тест памяти: Memtest.

Такая последовательность приводится не случайно. Если вы не сумели загрузиться в основном режиме, то стоит попытаться сделать это в Safe Mode, а если и здесь потерпели фиаско, то имеет смысл проверить носитель на ошибки или протестировать модули оперативной памяти. На моей практике были инциденты, когда в плохой работе сервера и постоянных перезагрузках были виновны именно сбойные планки памяти.

Далее пользователя встречает окно **Welcome. Select an option**, о котором, собственно, и шла речь в начале главы: **Backup** или **Restore**. Такая гениальная простота интерфейса будет востребована и среди опытных сотрудников, и среди молодых новичков в подразделениях техподдержки (см. рис. 18.1).



Рис. 18.1. Так выглядит первичное окно после загрузки Redo Backup and Recovery

Нажимаем **Backup** и попадаем в следующее окно **Step 1: Select Source Drive** (см. рис. 18.2). Обычно перед загрузкой происходит

непродолжительная операция сканирования дисковой подсистемы. Стоит отметить, что Redo Backup and Recovery – продукт, в принципе, «всеядный» и хорошо работает с большим количеством устройств с PATA-, SATA-, SCSI- и SAS-интерфейсами. Пользователю предлагают выбрать диск – **Click on the box below to select the source drive that you would like to create a backup image from.** В нашем случае мы видим только один жесткий диск (виртуальный) на 25 Гб (второй у нас – CD/DVD-накопитель), поэтому смело нажимаем экранную кнопку **Next** (Далее) и переходим к следующему этапу.

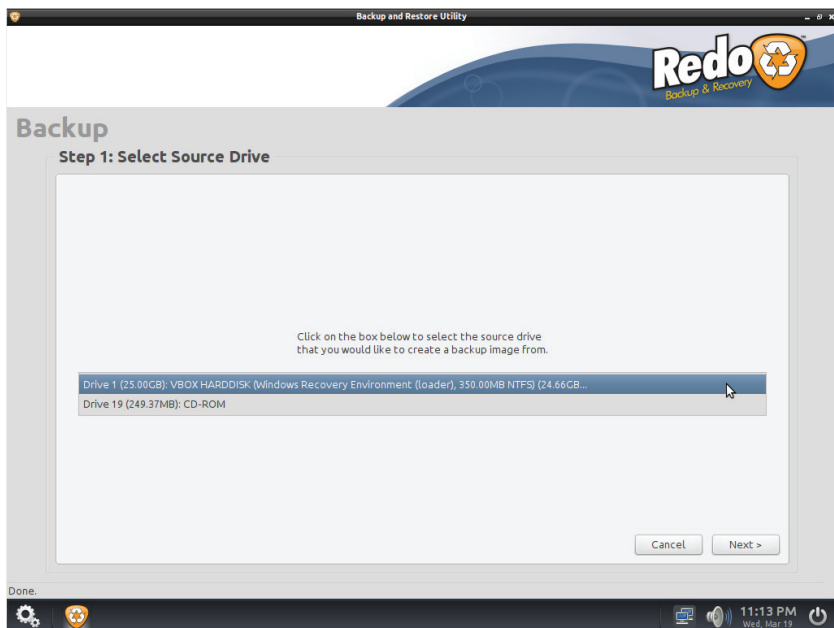


Рис. 18.2. Выбор жестких дисков для копирования

А дальше нам предлагают выбрать дисковый раздел, или, как еще говорят, «партицию». Появится новое окно **Step 2: Select Partitions to Save** с инструкцией на экране: **Select which parts of the drive to create a backup of. Leave all part selected, if you are unsure** (Выберите, какой части диска необходимо выполнить резервную копию). Если не уверены – оставьте выделенными «все части» (см. рис. 18.3).

Я воспользовался мудрым советом и оставил все разделы выделенными, после чего нажал **Next**.

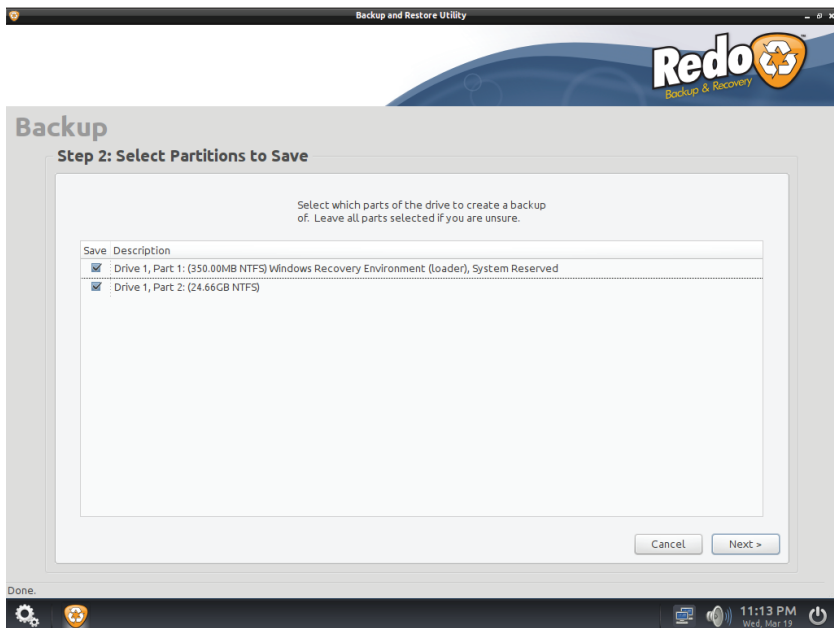


Рис. 18.3. Выбор разделов жесткого диска для копирования

Потом появится окно **Step 3: Select destination drive**. Здесь возможен выбор:

- **Connected directly on my computer** (Соединенный напрямую с моим компьютером);
- **Shared over a network** (Общедоступный по сети).

Выбираем второй пункт, так как в большинстве случаев (во избежание последствий форс-мажорных ситуаций) образы лучше хранить на сетевых ресурсах подальше от компьютера-источника.

Тут же на экране появились новые элементы для ввода сетевых параметров:

- **Server or share location** (Адрес сервера или сетевого ресурса);
- **Username (optional)** (Имя пользователя, если необходимо);
- **Password (optional)** (Пароль, если необходимо);
- **Domain (optional)** (Домен, если необходимо).

IP-адрес моего «хранилища резервных копий» и общую папку на нем я ввел вручную. Еще нужно было указать имя пользователя и пароль для доступа к ресурсу. Поле с именем домена я оставил пустым, так как целевой компьютер входит только в рабочую группу (см. рис. 18.4).

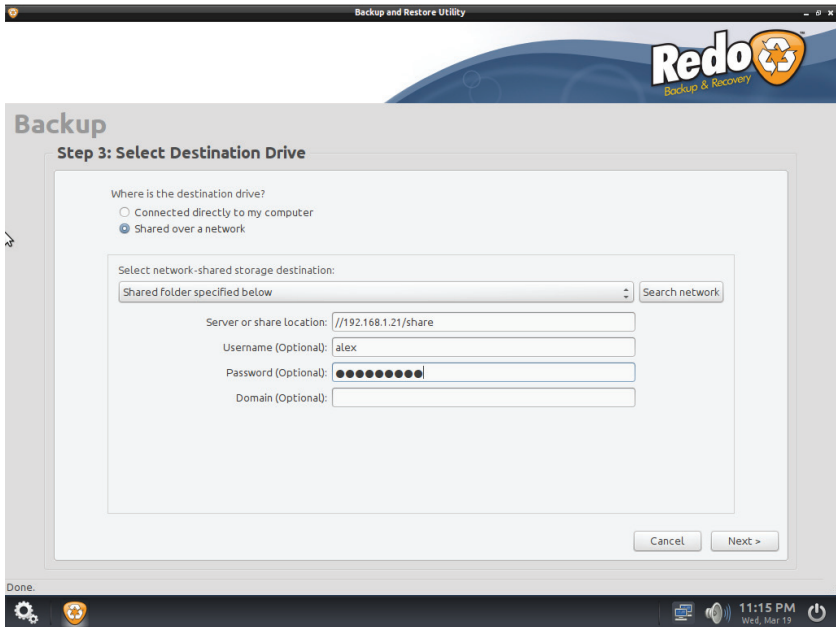


Рис. 18.4. Выбор целевого устройства для копирования

Сервер для складирования резервных копий лучше не включать ни в какие домены Active Directory и прочие глобальные системы авторизации. Дело в том, что копии серверов авторизации, например контроллеров домена или центров сертификации, тоже надо где-то хранить. И при сбое оных можно запросто получить ситуацию из законов Мерфи: «Ключ от срочной медицинской комнаты хранится в срочной медицинской комнате». То есть, чтобы восстановить сервер, нужно получить доступ к резервной копии, а чтобы получить доступ к резервной копии, нужно восстановить сервер. Поэтому лучше использовать в таких случаях локальную систему непосредственно на таком хранилище.

Если в общей папке есть еще подкаталоги, их также придется указать в окне **Step 4: Select Destination Folder**. Надо отметить, что очень нужная в таких случаях экранная кнопочка «Browse» верно отображает список каталогов на сетевом ресурсе.

Последнее окно **Step 5: Name your backup** предлагает ввести имя резервной копии.

Создаваемый при помощи Redo Backup and Recovery образ дисковой подсистемы или раздела жесткого диска не сохраняется в виде одного-единственного файла. В целевой директории будет находиться несколько файлов, каждый из которых важен для сохранности копии (см. рис. 18.5).

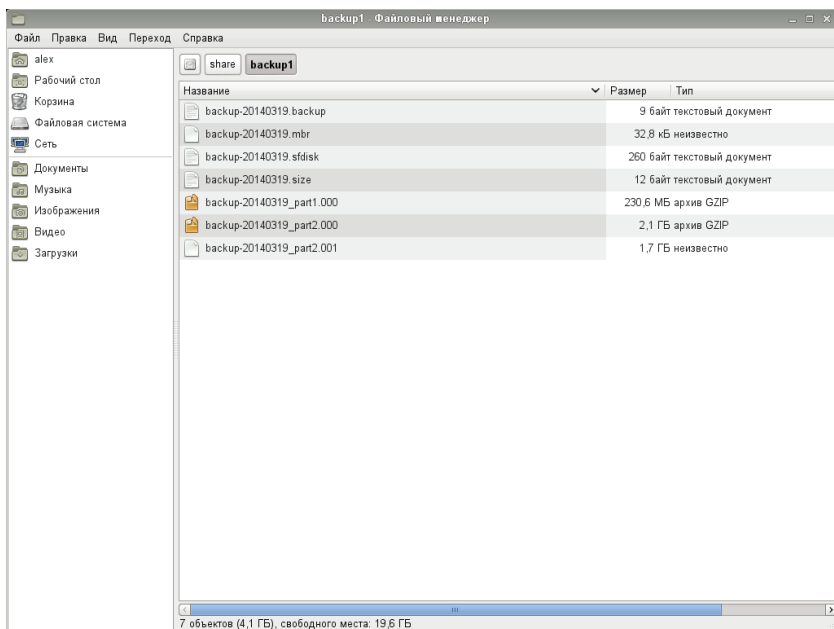


Рис. 18.5. Вид каталога на жестком диске с набором файлов образа

После нажимаем **Next** (Далее). Все, после этого началось копирование образа диска на сетевой ресурс.

После удачного выполнения программа поздравила меня фразой **Backup Saved in 9.3 minutes**. Вот так легко и оперативно можно снять образ с дискового раздела или всей подсистемы целиком.

### 18.3.2. Восстановление системы

Чтобы имитировать действия по восстановлению виртуальной системы, я выполнил клон тестовой виртуальной машины и удалил, а после заново создал на ней виртуальный жесткий диск. Таким образом у нас смоделирована ситуация, когда умер винчестер или даже RAID-

контроллер, купили что-то на замену и теперь восстанавливаемся из резервной копии.

Поэтому дождемся загрузки с нашего замечательного Live CD и смело нажмем на кнопку **Restore** (Восстановить) (см. выше рис. 18.1).

Надо отметить, что Redo Backup and Recovery устроен максимально просто и понятно для пользователей. Поэтому при восстановлении используются те же окна, что и при резервном копировании.

Идеология работы здесь предельно понятна. Сначала выбираем тип носителя: локальный диск или сеть, потом месторасположение образа. Далее отмечаем диск, который будем восстанавливать, и какой раздел (partition) на нем. Отвечаем положительно на вопрос о том, что все данные будут стерты и начинается процесс восстановления. Надо ли говорить, что клонированная виртуальная машина после восстановления заработала, как ни в чем не бывало. В общем, Redo Backup and Recovery – продукт достаточно качественный и удобный.

## 18.4. Меню Settings

Выше мы рассмотрели работу основных функций в режиме «по умолчанию». Но всегда может возникнуть ситуация, когда потребуются заставить работать в другом режиме, например использовать загрузку со съемного USB-накопителя.

Стоит отметить, что за реализацию подобных возможностей отвечает целый комплекс отдельных программ, доступных через меню **Setting**. Итак, нажимаем мышкой на значок с двумя шестеренками, имитирующий привычную кнопку **Пуск** из MS Windows, и посмотрим, что нам могут предложить.

**Create Bootable USB.** Хорошая удобная программа для создания загрузочной флэшки. Дело в том, что далеко не каждый компьютер снабжен персональным CD/DVD-накопителем. А вот USB-порты встречаются практически везде. Поэтому загрузочная флэшка давно стала ходовым инструментом у системных администраторов. Чтобы сделать такую флэшку, достаточно выбрать данный пункт. Появится окно программы с заголовком **Make Startup Disk**. Выбираете устройство чтения дисков CD/DVD, подключенный носитель, и программа выполнит перенос загрузчика и ПО на USB-накопитель.

**Customize Appearance.** Программа для настройки внешнего вида интерфейса. Можно выбрать несколько тем оформления, форму кнопочек, различные шрифты и т. д. и т. п. Программа, на мой взгляд,

не имеет прикладного значения и предназначена исключительно для эстетического наслаждения.

**Hardware Lister.** Программа для получения информации об имеющемся оборудовании. В основном применяется для анализа материнской платы и дополнительных устройств вроде RAID-контроллеров.

**Monitor Settings.** Программа для настройки основных параметров монитора.

**Windows Setting.** Запускает стандартный Openbox Configuration Manager. Так же, как и Customize Appearance, служит скорее для улучшения внешнего вида, нежели для упрощения работы.

## 18.5. Дополнительные возможности

Помимо основных функций: снятия образа диска и восстановления из такой копии, – Redo Backup and Recovery обладает довольно приличным количеством дополнительного программного обеспечения, которое окажется весьма полезным для системных администраторов при решении повседневных задач. Ниже опишем несколько наиболее интересных, с моей точки зрения, программ, входящих в данный дистрибутив.

### 18.5.1. Программа Gparted

**Gnome Partition Editor (GParted) Live** – программа для управления разделами жесткого диска, способна производить операции по созданию, удалению, изменению размеров, перемещению, проверке и копированию файловых систем и разделов жесткого диска.

При этом данный инструмент позволяет выполнять такие действия, как изменение размеров разделов, их копирование и перемещение, без потери имеющихся на них данных! Этот замечательный инструмент может работать с разделами, созданными в операционных системах Windows и Mac OS X, а также в различных дистрибутивах Linux.

Поддерживаемые файловые системы: ext2, ext3, ext4, FAT16, FAT32, HFS, HFS+, JFS, NTFS, ReiserFS, Reiser4, XFS. Например, с помощью GParted можно проделать следующие манипуляции:

- создать таблицу разделов (например, MS-DOS или GPT);
- создать, переместить, удалить раздел;
- изменить размеры и метку тома;
- изменить раздел и проверить его на наличие ошибок;
- включить и отключить соответствующие признаки (например, загрузочный или скрытый раздел);

- выровнять разделы;
  - можно восстановить данные с потерянных разделов.
- GParted работает со следующими аппаратными средствами:
- жесткие диски: SATA, IDE, SAS или SCSI;
  - устройства флэш-памяти, такие как карты памяти и твердотельные накопители (SSD);
  - некоторые RAID-контроллеры, а также программные RAID на базе Linux (SoftRAID).

Вот такая полезная программка идет в нагрузку к стандартным функциям резервного копирования и восстановления данного Live CD.

### 18.5.2. Другие средства

Помимо Gparted, на диске присутствует еще довольно обширный список инструментов, включенных в образ с целью сделать работу системного администратора максимально комфортной. Среди них:

- **Gnome Disk Utility** – богатая возможностями программа с GUI-интерфейсом, написанная на C++/GTK (GTK2/GTK3). Служит для манипуляций с дисковыми устройствами и другими носителями данных, основой приложения является небезызвестная утилита Udisks;
- **Drive Reset Utility** – программа для зачистки жесткого диска. Полезна для случаев, когда нужно удалить данные без дальнейшего восстановления;
- **PhotoRec** – программа с текстовым интерфейсом для восстановления удаленных данных. Первоначально создавалась для работы со сменными носителями (флэшками для смартфонов и фотоаппаратов), но позволяет восстанавливать всевозможные файлы с различных файловых систем: FAT, NTFS, exFAT, ext2/ext3/ext4, HFS+;
- **Logical Volumes Management** – программа для работы с томами LVM;
- **Disk Usage Analyzer (Baobab)** – утилита для анализа содержимого разделов жесткого диска. Позволяет понять, чем забит жесткий диск, нужна, чтобы удалить ненужную информацию до резервного копирования.

Кроме утилит, для непосредственной работы с дисками в дистрибутив входят файловый менеджер **PCManFM**, просмотрщик файлов **GpicView**, текстовый редактор **LeafPad**, интернет-браузер **Chromium** и, конечно же, программы текстового терминала **LXTerminal** для поддержки работы с командной строкой.

## 18.6. Некоторые ограничения

Отсутствует возможность настройки сети вручную. Точнее, нет удобного окна настроек с графическим интерфейсом, который предоставлял бы пользователю необходимые функции по настройке сетевого протокола и управлению сетевыми устройствами. Можно задействовать командную строку, но для этого требуются знания UNIX-like-систем на уровне администратора.

Отсутствует возможность добавить драйверы оборудования при загрузке системы с Live CD. К сожалению, пока что в вопросах поддержки оборудования приходится полагаться на «встроенные возможности» данной сборки Linux.

## 18.7. Заключение

В этой главе шла речь о продуктах для снятия образа диска. К услугам современных системных администраторов – множество качественных коммерческих и бесплатных продуктов. Поэтому чтобы лишний раз подстраховаться и спасти данные – достаточно просто этого захотеть.

# Глава 19

## Система резервного копирования для малого предприятия

*Пользователь не знает, чего он хочет, пока не увидит то, что он получил.*

(Э. Йодан)

### 19.1. Предисловие

Малый бизнес является непростой сферой для применения информационных технологий. С одной стороны, он пока не имеет значительных финансовых средств для внедрения дорогостоящих решений, с другой – он так же остро нуждается в информационной поддержке. Следует помнить, что малые предприятия часто находятся в ситуации, когда потеря одного-единственного крупного контракта может уничтожить хозяйственный субъект. Поэтому серьезный сбой в ИТ-системе может привести к смерти всего бизнеса.

Как разрешить это противоречие, с одной стороны, создав надежную систему, с другой – затратив минимум средств? Это настоящее искусство. Зачастую самые интересные и эффективные решения в области защиты данных выходили именно с «нижних этажей» бизнеса.

В данной главе описывается система, которая была спроектирована и внедрена несколько лет тому назад. Тем не менее рассматриваемый пример прекрасно подходит в качестве иллюстрации аналогичных мини-систем. Читатели, предпочитающие использовать готовые

примеры из книг в качестве шаблона, могут без особых проблем применить описанные в данной главе продукты, но с более поздними версиями.

## 19.2. Выбор оборудования

Наиболее целесообразно пойти по пути сочетания двух аппаратных решений: копирования образов системы на файл-сервер.

### 19.2.1. Использование медиа-сервера в качестве хранилища для резервных копий

Защититься от большинства внутренних факторов можно, установив специальный файловый сервер для создания и хранения резервных копий. Таким образом, большая часть данных копируется по сети в период наименьшей загрузки, чаще всего ночью на жесткие диски сервера, специально выделенного для хранения резервных копий.

Преимущества: данная система легко организуется и требует относительно небольших затрат. Достаточно компьютера с большим объемом дискового пространства и операционной системы. Данный комплект вполне можно использовать в качестве платформы при организации файл-сервера для хранения резервных копий информации с других серверов и рабочих станций.

Ограничения: эта технология мало подходит для защиты в случае форс-мажора, так как информация хранится на территории предприятия. Также она может оказаться бесполезной в случае заражения компьютерным вирусом и/или вредоносного воздействия других лиц, так как Backup-сервер, находящийся во включенном состоянии и доступный по сети, также может подвергнуться как вирусной, так и любой другой атаке.

### 19.2.2. Запись данных на ленточный картридж

Эта технология предполагает наличие некоего устройства, способного писать на сменные носители данных. В совокупности с физической транспортировкой носителя, содержащего свежую копию данных, за пределы предприятия (в дальнейшем этот прием будем называть off-site) данное решение способно обеспечить сохранность данных в большинстве случаев.

Преимущества: защищает как от внутренних, так и от большинства внешних факторов.

Ограничения: требует дополнительного оборудования и программного обеспечения для резервного копирования. Замедляется процесс восстановления данных в связи с необходимостью обратной транспортировки носителя на территорию предприятия. Также требует наличия дополнительных навыков у лица, ответственного за сохранение и восстановление данных.

### 19.2.3. Сочетание обоих методов

Как я уже сообщал выше, наиболее рациональным представляется метод, сочетающий в себе наличие внутреннего медиа-сервера с дисковым хранилищем и копирование данных на сменный носитель.

В этом случае появляется возможность значительно расширить возможности нашей backup-системы, используя технологию, когда данные в ночное время предварительно копируются на медиа-сервер, а уже оттуда выгружаются на ленту. Это позволит разгрузить компьютерную сеть и более эффективно использовать время, отведенное для резервного копирования. Так как при этом медиа-сервер обеспечивает процесс копирования данных на ленту, необходимо выбирать аппаратные и программные средства, способные реализовать данную функцию.

## 19.3. Создание системы резервного копирования

Учитывая величину оборота компании, можно вычислить примерную стоимость одного бизнес-дня и, соответственно, важность коммерческих данных. В наших условиях целесообразно остановиться на следующем варианте.

В качестве метода резервного копирования используем сочетание Backup-сервера и накопителя на съемных носителях (ленточного накопителя).

Схема ротации реализует сценарий, по которому выполняется еженедельный Full backup предварительно на Backup-сервер с последующей выгрузкой на ленту и транспортировкой в безопасное место (off-site) ленточного носителя данных (картриджа) (см. рис. 19.1). В будние дни выполняется Incremental Backup, позволяющий восстанавливать данные из актуальной копии. В подобных системах Backup-сервер также называют «медиа-сервером», стремясь подчеркнуть его промежуточное значение перед окончательным копированием на ленту.



Рис. 19.1. Процесс резервного копирования с off-site

### 19.3.1. Планирование процесса восстановления

Создание резервной копии – это, безусловно, очень важная часть в системе защиты и восстановления данных. Но не менее важно знать, как будут восстанавливаться данные из резервной копии.

Поэтому при выборе системы резервного копирования следует опираться на стандартизированное решение, поддерживаемое большинством производителей оборудования и программного обеспечения. Например, таким решением являются ленточные накопители (стримеры), поддерживающие стандарт LTO (Linear Tape Open), разработанный такими известными участниками компьютерного рынка, как IBM, HP и Seagate.

Примем за основу следующую схему восстановления:

- 1) мелкие, но срочные задачи по восстановлению данных, такие как реанимация испорченного по ошибке файла, выполняются с медиа-сервера;
- 2) в случае возникновения более серьезных задач, например когда нужно восстановить данные после серьезной аварии (после уничтожения большого количества файлов компьютерным вирусом и т. д.), а также в случае необходимости получения доступа к информации за прошлый период будем восстанавливать со съемного носителя, хранящегося в безопасном месте.

В нашем случае выбран децентрализованный метод резервного копирования. Данные с Windows-серверов, которых в нашей сети явное большинство, будут копироваться посредством средств, встроенных в операционную систему Windows Server, на сетевой ресурс медиа-сервера, с серверов на базе Unix-систем – посредством программы tar. Причина довольно банальная: необходимо в кратчайшие сроки создать работающую backup-систему и делегировать полномочия по выполнению рутинных операций резервного копирования другому лицу. То есть система должна быть максимально простой и понятной, при этом не требующей изучать сложный программный комплекс, такой как Backup Exec или Bacula.

### 19.3.2. Выбор программного обеспечения

В качестве операционной системы был выбран Linux CentOS за его совместимость с широко известной операционной системой Linux Red Hat Enterprise (RHEL), поддержку rpm-пакетов и простоту установки и обслуживания.

Так как наш медиа-сервер просто обязан быть файловым сервером, мы должны будем установить программный пакет Samba для обеспечения функций файл-сервера в среде Windows, работающего по SMB-протоколу.

Для облегчения задач по управлению и удаленному администрированию, а также для дальнейшего делегирования полномочий другим сотрудникам (например, администратору Windows-систем) будем использовать протокол RDP. Для этого установим программу xrdp для поддержки терминального доступа посредством RDP-клиента. Для работы xrdp нужен поднятый VNC-сервер, поэтому мы должны установить и его.



Программа xrdp в некоторых реализациях не всегда корректно работает с тем или иным RDP-клиентом. Например, в случае с xrdp-0.4.0-1.el5.rf для CentOS не поддерживается работа с последней версией RDP-клиента, поставляемого с Service Pack 3 для Windows XP. В подобных случаях рекомендуется пользоваться программой rdesktop (в частности, ее Windows-версией rdesktop.exe).

Для облегчения работы с файлами имеет смысл установить Midnight Commander – файловый менеджер, аналог небезызвестного Norton Commander'a.

Ну и наконец, необходимо установить само программное обеспечение для поддержки работы с ленточным накопителем: HP Data Protector Express Single Server Edition.

### 19.3.4. Выбор оборудования

В качестве медиа-сервера был выбран старенький сервер SuperMicro в корпусе «big tower».

Аппаратное обеспечение сервера:

- материнская плата: Super X6DHE-XG2;
- объем ОЗУ: 1 Гб.

Дисковые контроллеры:

- встроенный в материнскую плату контроллер Adaptec Embedded Host Raid;
- дополнительный RAID-контроллер LSI Logic MegaRAID Ser523 в формате PCI-X-133 с четырьмя портами для подключения SATA-дисков;
- 2 жестких диска Seagate ST31000340AS емкостью 1 Тб, предназначенных для промежуточного хранения данных;
- 2 жестких диска Seagate ST3500320NS емкостью 500 Гб для объединения в RAID1 и установки на них операционной системы;
- RAID-контроллер SCSI для подключения ленточного накопителя: LSI Logic LSI20320-HP в формате PCI-X-133 с внешним SCSI-интерфейсом.

Ленточный накопитель HP Storage Works Ultrium 1840LTO4 вместе с кабелем для подключения к внешнему SCSI-интерфейсу.

Кратко прокомментирую выбор аппаратного обеспечения. Как можно догадаться из вышеописанной конфигурации сервера, его собирали из того, что было под рукой. А вот ленточный накопитель LTO4 вместе с SCSI с контроллером для его подключения и запасом сменных картриджей (кассет) был приобретен отдельно.

### 19.3.5. Подготовка медиа-сервера

После того как мы полностью определились с оборудованием и программным обеспечением, пришло время воплотить наш проект в жизнь. Первым шагом установим операционную систему на наш медиа-сервер. Так как в Linux-системах часто не поддерживаются те или иные модели host-RAID-контроллеров, имеет смысл не рисковать и для размещений непосредственно операционной системы собрать RAID1 на базе имеющегося контроллера известной модели LSI Logic MegaRAID Ser523 и двух SATA-дисков по 500 Гб. Оставшиеся 2 жестких диска по 1 Тб мы подключим к контроллеру на материнской плате без объединения в RAID-массив. Таким образом, сама опе-

рационная система и прикладное ПО будут установлены на RAID1, а два отдельных диска по 1 Тб предназначены для хранения временных резервных копий.

Сами жесткие диски по 1 Тб, предназначенные для хранения промежуточных копий, имеет смысл подключить и разметить во время установки системы. Создаем на каждом из дисков по одному тому и используем точки монтирования `/vol0` и `/vol1` соответственно.

Установка операционной системы производится в обычном режиме. При установке программного обеспечения в качестве шаблона инсталляции выбираем **Server GUI** (Сервер с графическим интерфейсом). Из оконных менеджеров выбираем **Gnome** за его простоту и экономичность. Из дополнительных программных пакетов необходимо установить Samba для организации файлового сервера в среде Windows, а также VNC-сервер для обеспечения работы xrdp. Также рекомендуется установить пакеты из категории **Development** (Разработка): **Development Libraries** (Библиотеки разработки), **Development tools** (Инструменты разработки), **Legacy Software development** (Разработка Legacy-программ) – чтобы иметь возможность в дальнейшем инсталлировать программное обеспечение. И еще установим супердемон `xinetd`, чтобы иметь возможность запускать программу SWAT для управления сервером Samba.

Сразу при инсталляции создадим учетную запись (например, `backuper`), из-под которой и будет производиться работа по резервному копированию (см. рис. 19.2).

Если по каким-либо причинам не удалось создать ее сразу, в дальнейшем можно будет воспользоваться инструментом **User Manager**, который запускается из основного меню: **System** ⇒ **Administration** ⇒ **Users and Groups** (Система ⇒ Администрирование ⇒ Пользователи и группы).

Далее входим в систему как `root` для выполнения первоначальной настройки системы.



При невозможности воспользоваться программой **User Manager** можно воспользоваться командой `useradd` с соответствующими ключами.

Стоит немного сказать об основной концепции настройки программного обеспечения в нашем случае – максимально использовать возможности графического интерфейса. Большинство Unix-гуров сразу бы запустили любимую оболочку (shell) командной строки и принялись вводить команды и редактировать конфигурационные файлы. Но в нашем случае предстоит непростой процесс передачи

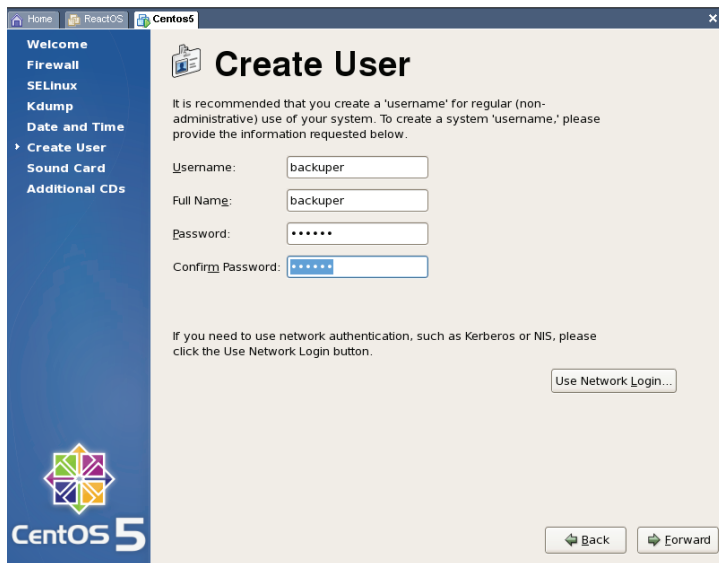
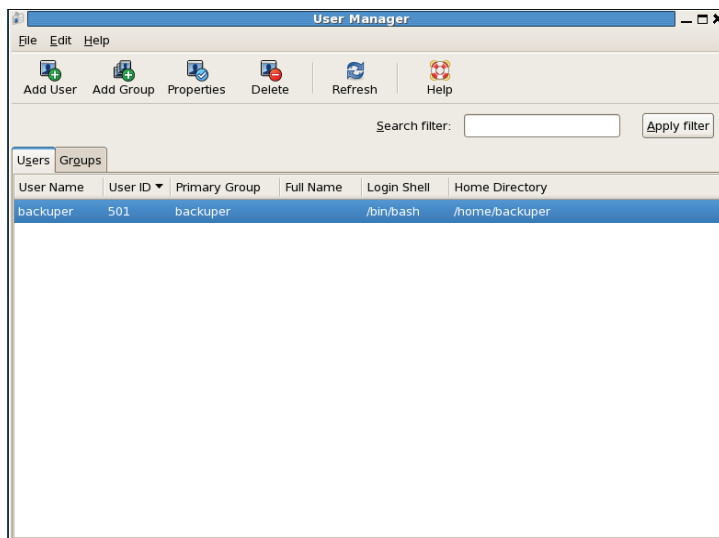


Рис. 19.2. Создание учетной записи при инсталляции системы

Рис. 19.3. Создание учетной записи при помощи инструмента **User Manager**

полномочий другому лицу, скорее всего, мало знакомому с работой с интерфейсом командной строки в Unix-системе. Поэтому все, что можно сделать через графический интерфейс, должно быть сделано именно так. Исходя из этих соображений, мы создадим на рабочем столе иконку для запуска Midnight Commander (см. рис. 19.4).

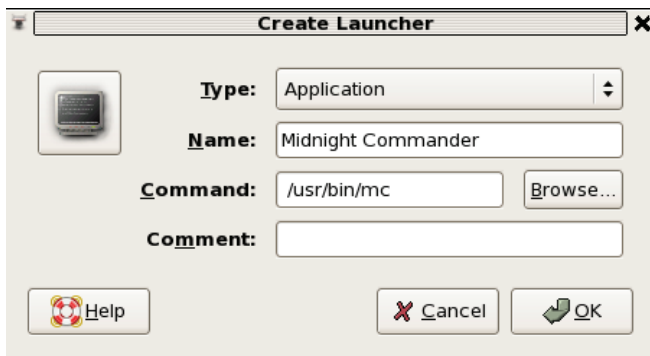


Рис. 19.4. Создание ярлыка для запуска Midnight Commander

Как уже упоминалось выше, при настройке Samba мы также не будем отклоняться от намеченной стратегии и воспользуемся веб-интерфейсом под названием SWAT.

Так как SWAT запускается посредством супердемона xinetd, необходимо отредактировать файл `/etc/xinet.d/swat`.

Вот как примерно должен выглядеть файл после редактирования:

```
# default: off
# description: SWAT is the Samba Web Admin Tool. Use swat \
# to configure your Samba server. To use SWAT, \
# connect to port 901 with your favorite web browser.
service swat
{
    port                = 901
    socket_type         = stream
    wait                = no
    user                = root
    server              = /usr/sbin/swat
    log_on_failure += USERID
    disable             = no
}
```

Теперь можно запустить сам веб-интерфейс. Набираем в строке браузера (например, Mozilla Firefox) адрес `http://127.0.0.1:901` и,

введя имя пользователя root с соответствующим паролем, попадаем в окно SWAT. Далее, воспользовавшись вкладкой **Password** (Пароль), необходимо задать имя пользователя для доступа к назначаемым сетевым ресурсам. Для того чтобы избежать дополнительных проблем, настоятельно рекомендую использовать те же имя и пароль пользователя, которые уже заведены в системе. Поэтому вводим имя пользователя backupер, тот же пароль, что и при создании пользователя Unix, и нажимаем кнопку **Add New User** (Добавить нового пользователя), далее необходимо разрешить данную учетную запись, поэтому нажимаем кнопку **Enable User** (Разрешить пользователя) (см. рис. 19.5).

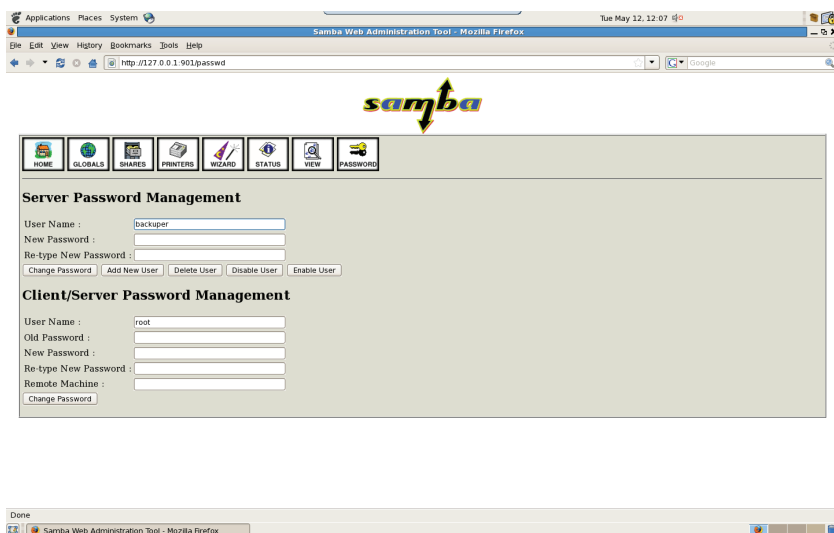


Рис. 19.5. Добавление пользователя Samba посредством SWAT



При невозможности воспользоваться программой SWAT можно применить команду «`smbpasswd`» с ключами «-a» для добавления и «-e» для разрешения учетной записи пользователя Samba.

Для того чтобы иметь возможность более гибко распорядиться дисковыми ресурсами (например, мы в дальнейшем не захотим отдавать все два терабайта дискового пространства под общие ресурсы), создадим на подключенных дисках vol0 и vol1 каталоги /vol0/backup0 и /vol1/backup1. Для того чтобы пользователь backupер имел доступ к этим ресурсам, сменим владельца каталога на backupер и за-

дадим соответствующие права. Для этих целей довольно удобно использовать Midnight Commander.



При невозможности воспользоваться программой Midnight Commander нужно использовать команды: `mkdir` – для создания каталога, команды `chmod` и `chown` изменения разрешений и смены владельца (группы) соответственно.

В случае если по каким-то причинам не удалось сразу создать необходимые разделы и подключить диски, чтобы сделать это после инсталляции, придется немного поработать в командной строке.

Запускаем программу Terminal или любую другую оболочку командной строки.

При необходимости переходим в режим суперпользователя:

```
[root@backupsml vol0]# ls /dev/sd*
Password:
```

Получаем список файлов SATA-устройств:

```
[root@backupsml vol0]# ls /dev/sd*
/dev/sda /dev/sda1 /dev/sda2 /dev/sda3 /dev/sdb /dev/sdc
```

Файлы устройств: `/dev/sda /dev/sda1 /dev/sda2 /dev/sda3` – нас не интересуют, так как относятся к уже подключенному дисковому массиву RAID1 с операционной файловой системой. А вот `/dev/sdb` и `/dev/sdc` как раз и являются файлами устройств тех самых дисков, которые предстоит подключить.

Для создания необходимого раздела на диске вызываем программу `fdisk`.

```
[root@backupsml vol0]# fdisk /dev/sdb
```

(Первоначально программа выдает сообщение, подобное этому:)

```
Device contains neither a valid DOS partition table, nor Sun, SGI or
OSF disklabel Building a new DOS disklabel. Changes will remain in
memory only, until you decide to write them. After that, of course, the
previous content won't be recoverable.
```

```
The number of cylinders for this disk is set to 121601.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
```

- 1) software that runs at boot time (e.g., old versions of LILO)
- 2) booting and partitioning software from other OSs  
(e.g., DOS FDISK, OS/2 FDISK)

```
Warning: invalid flag 0x0000 of partition table 4 will be corrected by
w(rite)
```

```
Command (m for help): m
```

**(Вызываем подсказку.)**

Command action

```
a toggle a bootable flag
b edit bsd disklabel
c toggle the dos compatibility flag
d delete a partition
l list known partition types
m print this menu
n add a new partition
o create a new empty DOS partition table
p print the partition table
q quit without saving changes
s create a new empty Sun disklabel
t change a partition's system id
u change display/entry units
v verify the partition table
w write table to disk and exit
x extra functionality (experts only)
```

**(Создаем новый раздел.)**

Command (m for help): n

(Выбираем тип раздела: первичный или расширенный)

Command action

```
e extended
p primary partition (1-4)
p
```

**(Выбираем тип раздела: первичный (primary) или расширенный (extended).)**

Partition number (1-4): 1

(Так как на создаваемый раздел отводится весь объем жесткого диска, первый и последний цилиндры выбираем по умолчанию.)

First cylinder (1-121601, default 1):

Using default value 1

Last cylinder or +size or +sizeM or +sizeK (1-121601, default 121601):

Using default value 121601

**(Проверяем список созданных разделов.)**

Command (m for help): p

**(В ответ программа выдаст информацию о созданных разделах.)**

```
Disk /dev/sdb: 1000.2 GB, 1000204886016 bytes
255 heads, 63 sectors/track, 121601 cylinders
```

Units = cylinders of 16065 \* 512 = 8225280 bytes

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1	1	121601	976760001	83		Linux

(Записываем изменения и выходим из программы.)

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

Создаем файловую систему на новом дисковом разделе.

```
[root@backupsml vol0]# mkfs /dev/sdb1
```

(В ответ программа выдаст сообщение, подобное этому:)

```
mke2fs 1.39 (29-May-2006)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
122109952 inodes, 244190000 blocks
12209500 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
7453 block groups
32768 blocks per group, 32768 fragments per group
16384 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632,
    2654208, 4096000, 7962624, 11239424, 20480000, 23887872, 71663616,
    78675968, 102400000, 214990848
```

```
Writing inode tables: done
```

```
Writing superblocks and filesystem accounting information: done
```

This filesystem will be automatically checked every 39 mounts or 180 days, whichever comes first. Use tune2fs -c or -i to override.

Создаем каталог для монтирования вновь созданного раздела с файловой системой:

```
[root@backupsml vol0]# mkdir /vol0
```

И монтируем файловую систему:

```
[root@backupsml vol0]# mount -t ext2 /dev/sdb1 /vol0
```

По аналогии размечаем и монтируем второй диск (/dev/sdc).

Для того чтобы созданные нами разделы автоматически монтировались при запуске, необходимо отредактировать файл /etc/fstab.

Ниже приведен пример файла /etc/fstab.

```
[root@backupsml vol0]# vi /etc/fstab
LABEL=/          /              ext3 defaults 1 1
LABEL=/reserve   /reserve       ext3 defaults 1 2
/dev/sdb1        /vol0          ext3 defaults 1 2
/dev/sdc1        /vol1          ext3 defaults 1 2
Tmpfs            /dev/shm       tmpfs defaults 0 0
devpts           /dev/pts       devpts gid=5,mode=620 0 0
sysfs            /sys           sysfs defaults 0 0
proc             /proc          proc defaults 0 0
LABEL=SWAP-sda2  swap           swap defaults 0 0
```

Далее, снова воспользовавшись SWAT, создаем общий ресурс для резервных копий. Переходим на вкладку **Share**. Задаем имя ресурса (в нашем случае): «backup0». Указываем путь к каталогу (в нашем случае) «/vol0/backup0» и пользователей, имеющих право доступа к каталогу: valid users и admin users (см. рис. 19.6).

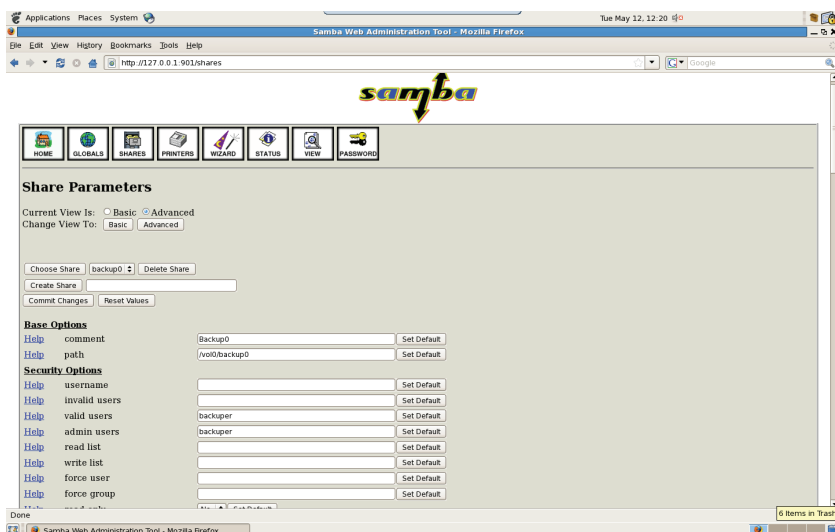


Рис. 19.6. Создание сетевого ресурса Samba посредством SWAT

В конечном итоге конфигурационный файл Samba – smb.conf должен выглядеть примерно следующим образом:

```
[global]
    workgroup = VAI.LAN
    server string = Backup Server LTO4
```

```
passdb backend = tdbsam
username map = /etc/samba/smbusers
ldap ssl = no
cups options = raw

[printers]
comment = All Printers
path = /var/spool/samba
printable = Yes
browseable = No

[backup0]
comment = Backup0
path = /vol0/backup0
valid users = backuper
admin users = backuper
read only = No
browseable = No

[backup1]
comment = Backup0
path = /vol1/backup1
valid users = backuper
admin users = backuper
read only = No
browseable = No
```

(где vol0 и vol1, собственно, и есть вышеописанные диски по 1 Тб для хранения резервных копий.)

Ну и, наконец, переходим к настройке программного обеспечения для выполнения резервного копирования на ленту – «HP Data Protector Express Single Server Edition». Для начала зайдём на страничку регистрации программы <http://www.hp.com/go/dataprotectorexpress/sse> для получения регистрации и получения ключа (valid key). Маленький секрет – желательно использовать MS Internet Explorer, так как в других браузерах скрипт может работать некорректно. Ответив на необходимые вопросы и получив e-mail с valid key, приступаем к установке программы.

Для этого входим в систему как пользователь backuper, вставляем CD с программой, получаем доступ к CD (при необходимости монтируем носитель) и запускаем программу Terminal (командную строку).

Набираем команду su и вводим пароль пользователя «root» для перехода в режим суперпользователя. И, перейдя в каталог с примонтированным CD, запускаем программу «install». Запустится окно установки программы (см. рис. 19.7).

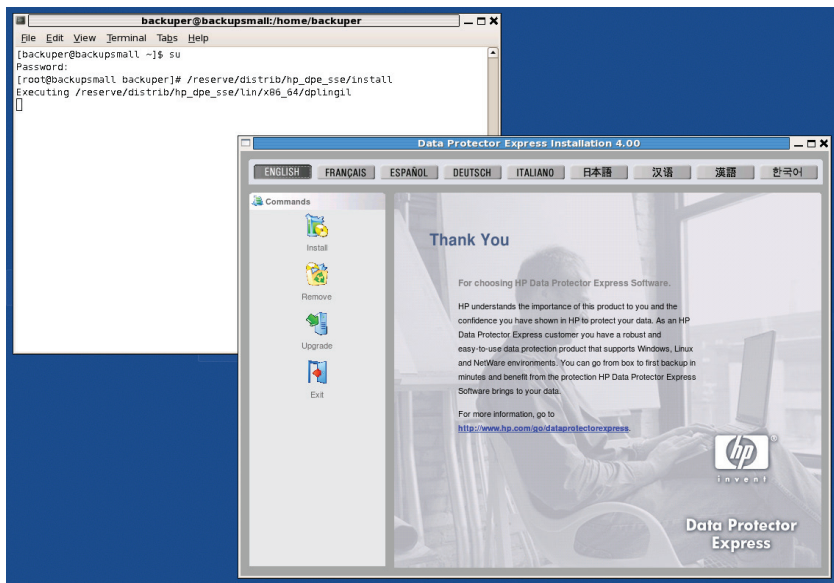


Рис. 19.7. Окно установки программы  
HP Data Protector Express Single Server Edition

В принципе, здесь особо описывать нечего. Установка программы проходит штатным путем, с использованием понятного графического интерфейса. Отдельного внимания заслуживает окно ввода регистрационного номера (см. рис. 19.8).

В конце программа-инсталлятор предложит запустить HP Data Protector Express Single Server Edition.

К сожалению, так как программа была запущена из-под учетной записи root, то и ярлык для запуска программы также был создан в домашнем каталоге пользователя «root». Для устранения этого нюанса нужно всего лишь скопировать ярлык с рабочего стола (Desktop) из домашнего каталога root на рабочий стол пользователя backuper и назначить соответствующие права.

### 19.3.5. Планирование заданий

Как было описано выше, резервное копирование на ленту будет выполняться раз в день по схеме Full Backup. Поэтому мы не будем решать вопросы, связанные со сложной схемой планирования, ротацией носителей и т. д. Все, что нам нужно, — это создать задачу по пол-

ному копированию выбранных каталогов медиа-сервера на ленту раз в день, при этом ленточный носитель каждый раз будет перезаписан.

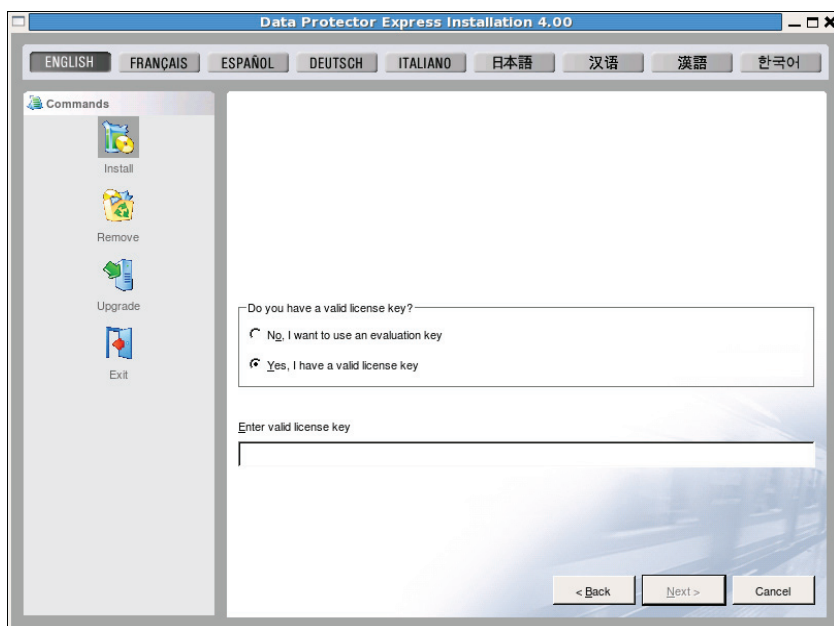


Рис. 19.8. Окно ввода регистрационного номера при установке программы HP Data Protector Express Single Server Edition

Для начала запускаем программу HP Data Protector Express Single Server Edition. (Если программа уже запущена, данный шаг можно пропустить.)

При старте программы потребуется ввести пароль доступа. По умолчанию реквизиты для доступа: сервер «localhost», имя пользователя «admin», без пароля.

Далее запустится основное окно программы HP Data Protector Express Single Server Edition. Надо отдать должное разработчикам – программа имеет простой и понятный интерфейс, большинство операций выполняется при помощи «помощников» («Wizards»). Для получения информации по работе с программой на установочном диске имеется подробная документация.

По умолчанию при запуске программы сразу открывается раздел **Wizards** (Мастера). (Для перехода в данный раздел можно восполь-

зоваться иконкой **Wizards** в правой части окна.) Кликаем на значке Backup и переходим в окно выбора способа копирования. Кликаем на самой нижней иконке в левой части окна **Backup Specific** (Настройки резервного копирования) и переходим в окно «**Wizard – Welcome**» (Мастер – Добро пожаловать). В поле ввода **Enter the name for the command being created** (Введите имя создаваемого задания) набираем имя нашего задания и щелкаем кнопку **Next**.

Теперь мы попадаем в окно **Select files and folders** (Выбор файлов и папок), где выбираются файлы и каталоги, подлежащие копированию на ленту (см. рис. 19.9).

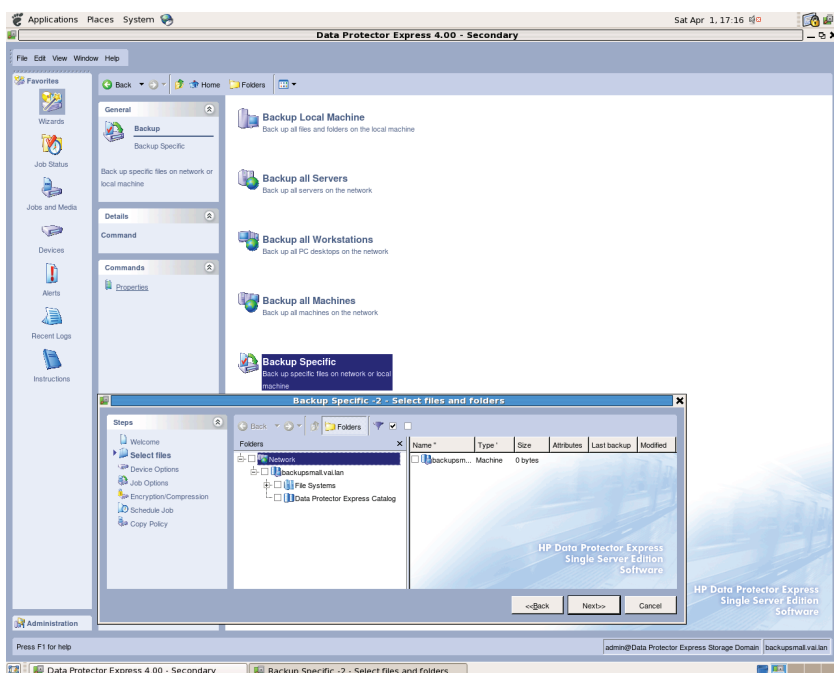


Рис. 19.9. Окно **Select files and folders**

Далее после нажатия кнопки **Next** (Далее) появляется окно **Device Options** (Настройки оборудования). Здесь необходимо выбрать **Device to be used** (Используемое устройство), а также установить некоторые параметры: **Auto format – Auto format mode** (Автоформатирование), **New media name** (Имя нового носителя) и др.

Снова нажимаем кнопку **Next** (Далее). В открывшемся окне **Job Options** (Параметры задания) задаем следующие параметры:

- **Backup Mode** ⇒ **Full** (Тип резервной копии – Полная);
- **Auto verify mode** ⇒ **Quick verify** (Режим автопроверки ⇒ Быстрая проверка). По умолчанию стоит **Full verify** (Полная проверка), но режим полной проверки архива занимает весьма продолжительное время, сравнимое со временем выполнения копирования, поэтому в данном случае ограничимся проверкой на читаемость носителя, учитывая тот факт, что мы каждый день создаем полную копию данных. Следует помнить, что для твердой уверенности в качестве записанной копии нужно обязательно использовать режим **Full verify**;
- **Span mode** ⇒ **Split file** (в случае если файл слишком велик для использования на одном носителе, следует использовать несколько носителей. В другом случае, при выборе «Restart file», программа будет просить загрузить в накопитель иной носитель большего размера);
- **Change mode** – **Prompt to another media** (Режим замены ⇒ Запросить новый носитель). Если программа не обнаруживает требуемого носителя, она высылает предупреждение с просьбой загрузить искомый носитель в накопитель.

Окно **Encryption/Compression** (Шифрование/Сжатие) появляется сразу после **Job options**. Здесь задаются параметры шифрования (encryption) и сжатия (компрессии). Следует заметить, что заявленный объем для носителей LTO (1,6 Гб для LTO4, 800 Мб для LTO3 и т. д.) будет доступен только при включении аппаратного сжатия (и то при условии, что информация на носителе может быть успешно сжата).

Предпоследнее окно **Schedule job** (Расписание заданий), как и следует из названия, служит для планирования выполнения задания в нужное время. Так как мы собираемся всего лишь запускать Full backup в нужное время, при этом просто перезаписывая вставленный носитель, и не обращаем при этом внимания на дополнительные нюансы типа ротации носителей, то ограничимся самым простым расписанием. Для этого выбираем:

- **Schedule type** ⇒ **Run repeatedly** (Тип расписания ⇒ Раздельный запуск);
- **Start time** – «10:30» (Время старта задания). К этому времени у нас в любом случае должен завершиться ночной Full backup с рабочих серверов на кэш-сервер;

- **Start date** – «15/05/2009» (Дата начала задания);
  - **Rotation type** ⇒ **No rotation** (Тип ротации ⇒ Без ротации).
- В нашем случае ротация носителей не требуется.

И в последнем окне **Copy Policy** (Политика копирования) остается нажать кнопку **Finish** (Завершить).

Все, задание по резервному копированию создано. Осталось контролировать его выполнение и, при необходимости, внести корректировки для обеспечения требуемой функциональности.

## 19.4. Тестирование восстановления

Для проверки возможности восстановления поступим следующим образом. Скопируем на кассету один небольшой файл, а потом попробуем его восстановить.

Для этого снова перейдем в окно **Wizards** и выберем иконку **Restore** (Восстановить). Это приведет нас в следующее окно, где нам будет предложен всего один-единственный метод восстановления **Restore Specific** (Параметры восстановления). Жмем на одноименную иконку и переходим в окно **Select files and folders**.

Здесь выбираются файлы, подлежащие восстановлению (в нашем тестовом случае это один-единственный файл, test.txt).

После нажатия кнопки **Next** мы переходим в окно **Device options**, где выбираются устройство и носитель, откуда будут восстанавливаться файлы (в нашем случае это «HP Ultrium 4 SCSI»).

Далее по кнопке **Next** перед нами появляется окно **Job options**. Основная деталь, на которую надо обратить внимание, – флажок **Restore files that are in use** (Восстанавливать ли файлы, использующиеся в данный момент). Дополнительные параметры, устанавливаемые в окне **Advanced options**, вызываемом по нажатии одноименной кнопки, в большинстве случаев лучше оставить без изменения.

Ну и, наконец, уже знакомое нам окно планировщика **Schedule job**.

Все, далее после нажатия кнопки **Finis** начнется процесс восстановления выбранного нами файла.

Конечно, весьма огорчает тот факт, что программа лишена возможности восстанавливать файлы в другое место (как минимум, мне лично так и не удалось обнаружить данную функцию). Придется учитывать этот факт в работе и переименовывать файлы и каталоги, которые желательно не затирать при восстановлении из резервной копии.

## 19.5. Завершающая настройка системы резервного копирования

После окончательной настройки медиа-сервера и системы записи на ленту работы осталось немного. Проверить доступность сетевых ресурсов на нашем сервере резервного копирования, настроить локальное ПО для создания резервных копий, еще нужно обязательно составить расписание резервного копирования, назначить ответственных лиц, передать им соответствующие полномочия, ну и, собственно, все.

## 19.6. Заключение

Рассматриваемая в этой главе система не слишком подходит в качестве решения с учетом будущего развития предприятия. По своей сути это скорее «затыкание бреши» в виде создания временной системы резервного копирования, нежели серьезная перспективная разработка.

Все же, несмотря на жесткие экономические условия, можно создать приемлемо работающую систему резервного копирования, затратив минимум средств (только на покупку ленточного накопителя и кассет) на приобретение необходимых компонентов. При этом созданная система позволяет делегировать полномочия другому лицу и при необходимости легко восстанавливать потерянные данные.

В следующей главе пойдет речь о том, как избежать подобных недостатков и сделать эффективную систему резервного копирования для системы с учетом роста предприятия.

# Глава 20

## Система резервного копирования крупной компании с многофилиальной структурой

*Существуют две взаимоисключающие мысли в голове человека, выполняющего обязанности сисадмина:*

- Когда же они научатся делать все сами?*
- С какого перепуга они пытались это сделать сами?!*

### 20.1. Предисловие

Для любого бизнеса существует та самая точка невозвращения, когда он перестает быть маленькой фирмой и становится средней, а еще немного погодя – и крупной компанией. Если говорить о государственных и других структурах, то такой момент возникает, когда к компьютерной сети центрального отделения начинают активно добавляться ИТ-инфраструктуры на периферии.

Хотим мы этого или нет, но в жизни не существует «ровной полосы». Или развитие, или деградация; или рост, или спад.

Момент, когда наступает такой переход, чаще всего проходит почти незаметно. Казалось, еще вчера заключили первый крупный контракт, а уже сегодня в наличии многофилиальная сеть, и сотрудники периферийных офисов уже звонят со своими проблемами.

Очень часто при таком развитии событий забывают про то, что старая ИТ-инфраструктура так и остается старой. Да, она кое-где разрослась, появились новые инфраструктуры, такие как корпоративный портал или общая база логистики, но такие сферы, как резервное копирование, дополнительный канал в Интернет, и другие необходимые вещи и процессы остаются «как есть» в течение определенного времени.

Хуже всего, когда развитие идет само по себе, и каждый филиал, каждая торговая точка решает эти проблемы самостоятельно.

У кого-то установлено новейшее оборудование и программное обеспечение, а кто-то до сих пор работает «по старинке», на древней технике, купленной много лет назад.

К сожалению, до сих пор системы отказоустойчивости, и в том числе система резервного копирования, традиционно находятся в «отстающих». И уж если начальство проникается проблемами их модернизации, то чаще всего, когда случилась какая-нибудь неприятность.

Говоря о развитии ИТ-инфраструктуры, обычно забывают про людей, которые обеспечивают ее работу. А ведь число и квалификация сотрудников, работающих в этой сфере, также могут отличаться.

Так и случилось в одной компании. Бурный рост вызвал частичное отставание определенных фрагментов ИТ-инфраструктуры, в том числе системы резервного копирования. Так продолжалось, пока не случилась маленькая беда. «Просто-напросто» умер RAID-массив с почтовой базой данных Exchange. Когда сисадмины ценой бессонных ночей и простоя нескольких участков все же восстановили большую часть переписки, возник резонный вопрос: «А как же дела в регионах?» А в регионах дела с резервными копиями обстояли еще хуже.

На этом предыстория заканчивается, и начинается постановка задачи. А задача была следующая: провести аудит и разработать недорогую, эффективную систему резервного копирования, чтобы большую часть работы выполняли региональные «айтишники», но резервные копии хотя бы раз в несколько месяцев передавались в центральный офис.

Я не буду утомлять читателя долгим детальным рассказом о результатах аудита. Скажу только, что у всех филиалов при массе отливаний было два схожих обстоятельства.

Первое – это использование платформы Microsoft как основы для бизнес-среды. То есть если речь шла о SQL-сервере, то можно было быть уверенным, что это – MS SQL, если о почтовом сервисе – MS Exchange и т. д.

Второе – крайне узкие каналы связи между филиалами и центральным офисом. Для репликации базы данных и удаленного терминального доступа по RDP такая пропускная способность еще удовлетворяла, а вот для копирования больших объемов данных – уже нет. И качество услуг местных провайдеров оставляло желать лучшего.

На плечи проектировщика легла непростая задача – предоставить гибкое решение, способное учитывать все эти нюансы.

## 20.2. Рассмотренные варианты

### 20.2.1. Вариант от EMC

Первое предложила компания EMC. По ее замыслу, в каждом филиале было необходимо установить медиа-сервер резервного копирования с большим дисковым массивом, который, собственно, и являлся бы основным хранилищем резервных копий.

Центр всей системы должен был располагаться в штаб-квартире. Там находятся управляющий сервер и мощная дисковая система хранения данных, в несколько раз превосходящая филиальную. Соответственно, весь архив предполагалось разместить на этих жестких дисках.

Передача данных между центральным офисом и филиалами должна производиться исключительно по каналам связи в свободное от основной работы время.

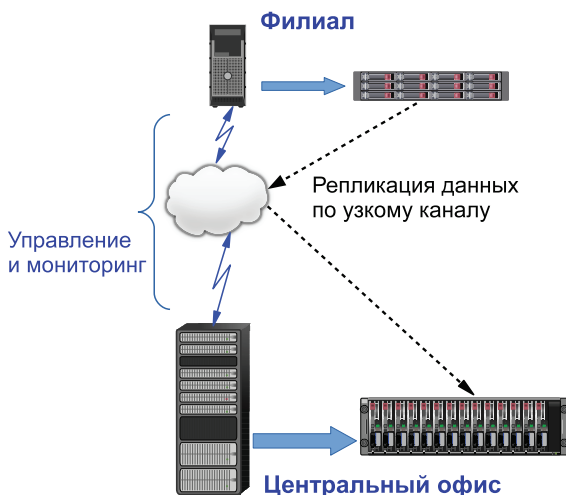


Рис. 20.1. Вариант решения от EMC

### ***Какие минусы у этого решения?***

Конечно, продукция ЕМС стоит весьма недешево. Когда инфраструктура, построенная на таких продуктах, растет постепенно, то финансовые затраты, пусть и весьма высокие, равномерно распределяются в течение длительного времени. Но заплатить такую сумму за один раз чаще всего нереально.

Еще один важный момент – во всех рекламных буклетах такого рода предложений не учитываются узкие каналы связи. Да, представители ЕМС долго и одухотворенно говорили о прекрасных алгоритмах дедупликации, которые есть в их продуктах. Если говорить о передаче инкрементальных копий, это прекрасно, алгоритмы сжатия и дедупликации здорово помогают в этом случае.

Но никто из них так и не ответил на вопрос: «А как же быть с полной резервной копией? Как этот самый Full Backup попадет из филиала в штаб-квартиру?»

Даже если положиться на везение и верить, что за время передачи связь не прервется, все равно процесс передачи всех данных занял бы около недели. При этом загрузка и без того слабых линий была бы настолько велика, что полностью лишила бы филиал доступа в сеть.

Абсолютно было непонятно, как предполагается выполнять Disaster Recovery. Основой для хранения резервных копий являлись дисковые массивы, которые размещались в серверной и в случае форс-мажорных обстоятельств, например пожара, должны были неминуемо погибнуть вместе с ней. Передача данных off-site, по тонким каналам в центральный офис занимает массу времени, и столько же потребуется, чтобы выполнить полное восстановление.

Еще один вопрос, который возник, заключался в подготовке и обучении персонала. В некоторых представительствах не было системных администраторов на постоянной основе. И для обслуживания привлекались сотрудники по аутсорсинговой модели. В других филиалах были свои системные администраторы общего профиля, но специалистов по работе со специализированными продуктами, такими как ЕМС Avamar, просто не было. И если, рассуждая теоретически, трудоустроенных специалистов еще можно отправить на дорогостоящие курсы, то что делать с «приходящими админами», не было понятно никому.

Поэтому данное предложение не подошло, и внимание всей команды переключилось на второй вариант.

### 20.2.2. Вариант автономной системы резервного копирования

В качестве второго варианта рассматривался вопрос установки локальных систем резервного копирования в каждый филиал, при этом время от времени комплект носителей, содержащих Full Backup, должен был отправляться в штаб-квартиру.

В качестве приложения для резервного копирования был выбран CA ARCserve Backup for Windows от компании CA Technologies (ранее она имела полное название «CA, Inc., Computer Associates»).

Причин для такого выбора было несколько. Во-первых, этот продукт реализует классическую централизованную схему, при этом в наличии есть большое количество программ-агентов для большинства известных платформ и приложений. От VMware ESX(i) до MS Exchange.

Кстати, об Exchange. Возможность выполнять так называемый granular backup, чтобы впоследствии иметь возможность восстановить отдельные письма из почтовых ящиков, явилась несомненным плюсом.

Широкий спектр поддержки оборудования и отсутствие необходимости специальным образом разыскивать драйверы для различных устройств, например ленточных накопителей, тоже был расценен как преимущество.



Рис. 20.2. Вариант решения с использованием съемных носителей

Вопрос с off-site и Disaster Recovery решался очень просто. Один комплект носителей с Full Backup должен все время храниться за пределами офиса. Где именно – это решала служба безопасности. Возможность сохранения состояния системы (System State) давала вероятность быстро восстановить систему, пусть и не на «голое железо», но довольно быстро.

Простой интерфейс управления и хорошая документация также сыграли свою далеко не последнюю роль.

Обратите внимание, что данная система не предполагала разделения на архивное копирование и снятие образа системы для Disaster Recovery. Это был явный минус системы.

Но, найдя утешение в словах: «Невозможно объять необъятное», – было решено отложить создание системы снятия образов на более поздний срок, и инженерная команда притупила к закупке и внедрению данного варианта.

## 20.3. Решение

В качестве основной схемы решения была принята централизованная схема среднего предприятия, когда один сервер играет роль и управляющего, и медиа-сервера.

В качестве основного носителя были выбраны кассеты LTO. Для упрощения манипуляций версиями была выбрана стандартная ежемесячная схема «дед–отец–сын». При этом ежемесячные полные копии должны были передаваться в центральный офис, а еженедельные полные копии вместе с инкрементальными – оставаться в филиале. При этом предыдущая недельная полная копия изымается службой безопасности и хранится за пределами офиса.

При такой схеме ротации для размещения промежуточных копий использовались картриджи LTO2, а для хранения полных – LTO3. Объем картриджа (200 Тб без сжатия LTO2 и 400 Тб без сжатия LTO3) на момент внедрения системы вполне хватало. Согласно выбранной схеме ротации, изымались только картриджи LTO3, а LTO2 оставались в накопителе для следующей перезаписи.

Примечательно, что системы резервного копирования в филиалах и штаб-квартире были построены по одному принципу. Различия были не качественные, а количественные: медиа-сервер в центральном офисе был мощнее, у него было больше жестких дисков, к нему было подключено больше ленточных накопителей и т. д.

Перемещение резервных копий между периферией и центральным офисом не должно было составить большого труда, так как сотрудники филиалов и штаб-квартиры постоянно летали на встречи друг с другом. Оставалось позаботиться о мерах безопасности, чтобы при краже или утере носителя информация на нем осталась недоступной для злоумышленника. В качестве основной меры защиты носителя использовалось шифрование, дополнительно применялась «безликая» схема маркировки, когда по надписи на кассете нельзя установить, что за данные там записаны.

Разумеется, резервные копии обязательно необходимо проверять. Поэтому в центральном офисе была организована целая процедура, согласно которой привезенные резервные копии первоначально передавались службе безопасности, она, в свою очередь, привлекала администратора резервного копирования для проверки возможности восстановления данных, после чего изымала носители обратно.

Еще один важный момент – все сотрудники, осуществляющие доставку резервных копий, были проинструктированы на предмет прохождения контроля пассажиров на авиалиниях. Ленточные картриджи они были обязаны перевозить в ручной клади (не сдавать в багаж!), при этом ни в коем случае не держать при себе при прохождении магнитных рамок. Эти простые меры позволяли уберечь ленточные носители от возможных нежелательных воздействий магнитных полей, а также от банальной кражи.

Роль управления всей сложной системой лежала на двух инженерах из центрального офиса, игравших роль администраторов резервного копирования в центральном офисе. Функции замены носителей и контроля за выполнением резервного копирования были делегированы местным системным администраторам в филиалах, которые выполняли роль операторов резервного копирования.

## 20.4. Процесс внедрения

Условно процесс внедрения можно разделить на несколько этапов:

- планирование;
- закупка оборудования;
- установка ПО и тестирование;
- передача в эксплуатацию и последующее сопровождение.

Далее расскажем обо всех пунктах по порядку.

### 20.4.1. Процесс планирования

Этот этап непосредственно связан со спецификой бизнеса. И это неудивительно, потому что выполнение работ зависит от выделяемых ресурсов и характера взаимодействия с другими подразделениями.

В данном случае для успешного выполнения были в наличии все необходимые составляющие: высшее руководство компании (как я уже писал выше) объявило резервное копирование зоной особого внимания. Поэтому задача имела высокий приоритет, деньги на закупку оборудования выделялись в достаточном количестве, а все остальные подразделения и сотрудники были обязаны оказывать необходимую помощь при внедрении.

В таких благоприятных условиях процесс планирования сводится к приблизительному подсчету времени, необходимого для выполнения работ, и составлению графика.

### 20.4.2. Закупка оборудования

В качестве сервера резервного копирования для филиалов был выбран сервер для монтажа в стойку высотой 1U и объемом памяти 8 Гб. Подключение к сети осуществлялось посредством двух LAN-интерфейсов Gigabit Ethernet, встроенных в серверную материнскую плату.

Важной составляющей был контроллер UltraSCSI-320 с разъемом для подключения внешних устройств.

В качестве ленточного накопителя использовался HP StorageWorks 1/8 G2 Tape Autoloader, подключаемый к серверу по интерфейсу UltraSCSI-320.

Для системы резервного копирования штаб-квартиры покупалось аналогичное, но более мощное оборудование. Сервер имел более мощный процессор, объем оперативной памяти 16 Гб, соответственно, его размер был уже 2U, и к нему было подключено 2 автолоадера HP StorageWorks 1/8 G2 Tape.

Говоря о закупках, не следует забывать о программном обеспечении. Так как основной платформой для работы была Windows, то на каждый сервер резервного копирования была куплена соответствующая лицензия для операционной системы. Также был приобретен комплект ПО CA ARCserve, состоящий из основной программы для сервера резервного копирования и набора программ-агентов. Состав и количество сильно отличались для каждого филиала. Все же можно выделить некоторые общие черты, например для каждого представи-

тельства была зарезервирована возможность копирования MS SQL Server, MS Exchange (включая дополнительный модуль Granular Backup для копирования и восстановления писем по отдельности), а также модель Open Files для файл-сервера, позволяющая копировать файлы, открытые для записи.

Все оборудование и программное обеспечение закупалось в центральном офисе, настраивалось и впоследствии передавалось в филиалы. Такой порядок упрощал унификацию решения, а наработанные заготовки, например образ системы с установленным ПО до ввода лицензии, значительно упрощали развертывание.

### 20.4.3. Установка ПО и тестирование

Как уже было сказано выше, инсталляция программного обеспечения производилась в штаб-квартире. Помимо простой установки, проводилось тестирование функций копирования-восстановления. Особое внимание уделялось работе устройств записи на ленту как наиболее «капризной» части решения.



Ни в коем случае нельзя перевозить ленточные накопители с неизвлеченными картриджами. Перед началом перевозки устройство должно быть полностью освобождено от носителей информации. Иначе при перевозке происходят мелкие механические повреждения, по вине которых устройство не просто не способно записать и считать информацию, но даже извлечь ленту из записывающего устройства становится невозможно.

### 20.4.4. Передача в эксплуатацию и последующее сопровождение

Функция самой передачи сводилась к написанию подробной инструкции (дополнительно к штатной документации на устройства и ПО), перевозке и инструктированию по телефону или Skype.

Сопровождение на первом этапе заключалось в постоянном контроле работы системы резервного копирования как в центральном офисе, так и на периферии. Для осуществления контроля использовалась штатная консоль RDP, традиционная для Windows-систем. Подключившись к удаленному серверу резервного копирования, администратор мог видеть состояние заданий и ознакомиться с журналами работы.

## 20.5. О печальном опыте использования DFS

Говоря о филиальной структуре, было бы несправедливо умолчать один факт. В данной компании штаб-квартира не представляла собой монолитного явления. Помимо центрального офиса, существовал еще и вспомогательный, где располагалось небольшое вспомогательное подразделение. Первоначально этому не придали значения, так как большую часть работы они выполняли на сервере в офисе компании, подключаясь удаленно по RDP. Но впоследствии оказалось, что там тоже существует файл-сервер, где хранятся некие данные для локального использования. И в соответствии с политикой компании их также нужно копировать.

И тут снова возник дуализм мнений.

Самым простым решением было установить локальную систему резервного копирования, тем более что на складе валялся старый автолоадер LTO2 и соответствующих картриджей было в достаточном количестве. Если учесть малый объем копируемых данных, то полной загрузки кассетами этого хоть и старого, но весьма надежного устройства хватило бы мини-филиалу на несколько месяцев.

Нужно было всего лишь купить контроллер UltraSCSI, подключить лоадер к файл-серверу. При этом не нужно было покупать дорогостоящее ПО – для поставленных целей с лихвой хватило бы штатных средств операционной системы.

Недостатком данного решения была необходимость приобретать контроллер (он стоил хоть и небольших денег, но их все равно нужно было платить) и время от времени забирать кассеты с резервными копиями из автолоадера. Если вспомнить, что перед этим была организована доставка носителей из удаленных филиалов, то этим обстоятельством вполне можно было пренебречь.

Но местный апологет Microsoft предложил использовать распределенную файловую систему DFS для организации репликации.

Основными аргументами были отсутствие необходимости покупать контроллер и забирать кассеты. Достаточно только настроить репликацию файлов в центральный офис и копировать файлы на ленту уже внедренной системой на базе ARCserve.

Горящие глаза от предвкушения попробовать эту волшебную технологию придали нашему любителю Microsoft большей убедительности, и руководство решило пойти ему навстречу.

Компьютеры и мини-сервер этого филиала были в том же доме – не Active Directory, что и в штаб-квартире. Поэтому нашему герою оставалось только выделить дисковое пространство для хранения редуцированной копии, создать и запустить схему DFS – и вот оно, готовое решение!

Поначалу все шло нормально. Копии файлов из филиала хоть и с некоторым запозданием, но оказывались в резервном каталоге, что давало возможность делать бэкап. Так продолжалось несколько месяцев.

Но потом произошло нечто странное. Репликация просто перестала работать. Не помогли ни анализ логов, ни даже повторное развертывание DFS. Решение проблемы осложнял тот факт, что, кроме этого поклонника Microsoft, из всего ИТ-отдела с данной проблемой возиться никто не хотел, мало того, коллеги постоянно напоминали, что ради его «супертехнологии» пришлось пожертвовать простым и надежным решением.

Так продолжалось еще несколько месяцев. В итоге было принято решение купить большой USB-диск и делать на него добавочную копию. А еще через некоторое время этот мини-филиал просто закрыли, а всех сотрудников перевели в центральный офис. И проблема неработающей репликации отпала сама собой.

Не стоит забывать, что все это время, когда прекратилась репликация DFS, резервное копирование выполнялось с перебоями, а порой и вовсе не проводилось. Поэтому очень важно быть последовательными в своих решениях и придерживаться проектных директив. Метания из стороны в сторону под девизом: «Как это интересно, а давайте попробуем...» – обычно ни к чему хорошему не приводят.

## 20.6. Заключение

Какие выводы можно сделать из данной истории?

Во-первых, не нужно гнаться за широко разрекламированными решениями. Принципа «Чем проще – тем надежнее» пока еще никто не отменял.

Во-вторых, очень важно видеть всю картину и весь процесс целиком, от закупки оборудования до инструкции по перевозке носителей авиарейсами.

Остается пожелать читателям успехов в непростом деле самостоятельного создания отказоустойчивых систем.

# Вопросы к пятой части

## для закрепления

## материала

1. Приведите примеры ошибок при создании или обслуживании систем резервного копирования.
2. Для чего используются системы создания образа диска?
3. В чем основные трудности при создании системы резервного копирования для малого предприятия?
4. С какими вариантами системы резервного копирования для филиальных сетей вы познакомились?
5. Почему нужно придерживаться единой концепции?

# Вместо послесловия

Когда я писал эту книгу, основной целью было посвятить читателя в основы систем сбережения данных.

И мой читатель теперь сможет сам построить свою систему сохранения и восстановления данных. Так, как он хочет, как позволяют финансы и требует обстановка. Не важно, где вы работаете, важно иметь желание сберечь информацию от разрушения. И моя книга призвана помочь в этом деле.

Разумеется, данное творение нельзя считать исчерпывающим руководством «от А до Я». Формат небольшой книги не может вместить в себя базу знаний человечества по вопросам сохранения данных.

Поэтому я не прощаюсь с читателями. Со временем, накопив информацию и опыт, я начну писать новую книгу о таких простых и очень важных вещах.

Остается только пожелать читателям успехов в славном деле создания отказоустойчивых систем.

# Литература и ИСТОЧНИКИ

1. Сайт кафедры АСУТП НИУ «МЭИ» <http://www.asutp-mpei.ru/>.
2. Документация Common RAID Disk Data Format Specification: [http://www.snia.org/sites/default/files/SNIA\\_DDF\\_Technical\\_Position\\_v2.0.pdf](http://www.snia.org/sites/default/files/SNIA_DDF_Technical_Position_v2.0.pdf).
3. Описание файловой системы ZFS на сайте Oracle: <https://docs.oracle.com/cd/E19253-01/819-5461/zfsover-2/>.
4. Введение в файловую систему BTRFS: <https://lwn.net/Articles/576276/>.
5. Статья на сайте Technet Microsoft о распределенной файловой системе DFS: <http://technet.microsoft.com/ru-ru/library/cc731089.aspx>.
6. Коршунов А. Vox Backup – горячие резервные копии // Системный администратор. 2006. № 5. С. 6–11. Режим доступа: [http://www.samag.ru/art/05.2006/05.2006\\_09.html](http://www.samag.ru/art/05.2006/05.2006_09.html).
7. Статья на сайте Technet Microsoft, посвященная Shadow copy: <http://technet.microsoft.com/en-us/library/cc738819%28v=ws.10%29.aspx>.
8. Страницка на официальном сайте компании Acronis, посвященная продукту Acronis Backup & Recovery: <http://www.acronis.ru/backup-recovery/smallbusiness.html>.
9. Страницка на официальном сайте компании Paragon-Software, посвященная продукту Paragon Drive Backup Server: <http://www.paragon.ru/corporate/db-server/>.
10. Блог компании Veeam Software, статья о дедупликации: <http://habrahabr.ru/company/veeam/blog/203614/>.
11. Барамба С. Disaster Recovery Plan, или В ожидании катастрофы // Системный администратор. 2013. № 10. Режим доступа: <http://samag.ru/archive/article/2536>.
12. Методичка по технологии D2D2T Backup Architectures от Quantum: <http://hosteddocs.ittoolbox.com/wp00118v2.pdf>.
13. Канал на Youtube компании «Чип и Дип», видеоролик, рассказывающий о стримерах: [http://www.youtube.com/watch?v=daVy6gNid\\_w](http://www.youtube.com/watch?v=daVy6gNid_w).

14. Статья на сайте компании DATARC «Сравнение интерфейсов SCSI, SAS и SATA»: [http://www.dataarc.ru/articles/sravnenie\\_interfeisov\\_scsi\\_sas\\_i\\_sata.html](http://www.dataarc.ru/articles/sravnenie_interfeisov_scsi_sas_i_sata.html).
15. Сайт Fibre Channel Industry Association: <http://fibrechannel.org/>.
16. Найманн Д., Браун К., Авелар В. Влияние изоляции горячего и холодного коридоров на температуру и эффективность ЦОД // Журнал сетевых решений/LAN. 2013. № 10. Режим доступа: <http://www.osp.ru/lan/2013/10/13037887/>.
17. Шапиро Л. Active Directory Domain Services. Двухфакторная аутентификация. Теоретические основы // Системный администратор. 2010. № 7–8. С. 100–105.
18. Бирюков А. А. Информационная безопасность: защита и нападение. Режим доступа: <http://dmkpress.com/catalog/computer/security/978-5-94074-647-8/>.
19. Описание VLAN (Virtual Local Area Network): <http://xgu.ru/wiki/VLAN>.
20. Статья ACL: списки контроля доступа в Cisco IOS: <http://habrahabr.ru/post/121806/>.
21. Статья о технологии LAN-Free: [http://www.storagenews.ru/11/lan\\_free\\_backup.pdf](http://www.storagenews.ru/11/lan_free_backup.pdf).
22. Описание атаки «Man in the Middle»: <http://www.veracode.com/security/man-middle-attack>.
23. Logical Volume Manager Administration – руководство от Red Hat: [https://access.redhat.com/site/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Logical\\_Volume\\_Manager\\_Administration/](https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Logical_Volume_Manager_Administration/).
24. Информация о программе **PCManFM** на сайте ArchLinux: <https://wiki.archlinux.org/index.php/PCManFM>.
25. Страница описания **HP StorageWorks 1/8 G2 Tape Autoloader** на сайте HP: <http://www8.hp.com/us/en/products/tape-automation/product-detail.html?oid=3319912>.
26. MS TechNet: Распределенная файловая система (DFS) в системе Windows Server 2003 с пакетом обновления 1 (SP1): <https://technet.microsoft.com/ru-ru/library/cc736868%28v=ws.10%29.aspx>.
27. Бережной А. Н. Организуем систему резервного копирования для малого и среднего офиса // Системный администратор. 2009. № 6. Режим доступа: <http://samag.ru/archive/article/2020>.

28. Бережной А. Н. Альтернатива файловому серверу – это дисковое хранилище NETGEAR ReadyNAS // Системный администратор. 2009. № 7. Режим доступа: <http://samag.ru/archive/article/861>.
29. Бережной А. Н. NTI Shadow for ReadyNAS: проводим резервное копирование данных // Системный администратор. 2009. № 8. Режим доступа: <http://samag.ru/archive/article/876>.
30. Бережной А. Н. NETGEAR ReadyNAS Pro. Установка дополнительных модулей и программ // Системный администратор. 2010. № 2. Режим доступа: <http://samag.ru/archive/article/949>.
31. Бережной А. Н. Резервное копирование. Теория и практика. Краткое изложение. Ч. 1, 2 // Системный администратор. 2010. № 12. Режим доступа: <http://bit.samag.ru/archive/article/1074>.
32. Бережной А. Н. Резервное копирование. Теория и практика. Краткое изложение. Ч. 3 // Системный администратор. 2011. № 1–2.
33. Бережной А. Н. Кошмар, который закончится. Как выбраться из лабиринтов проблем // Системный администратор. 2012. № 10. Режим доступа: <http://samag.ru/archive/article/2303>.
34. Бережной А. Н. Резервное копирование виртуальных машин // Системный администратор. 2013. № 10. Режим доступа: <http://samag.ru/archive/article/2535>.
35. Бережной А. Н. Средства резервного копирования для виртуальных систем // Системный администратор. 2014. № 1. Режим доступа: <http://samag.ru/archive/article/2609>.
36. Бережной А. Н. Резервное копирование для больших и маленьких // Системный администратор. 2014. № 3. Режим доступа: <http://samag.ru/archive/article/2637>.
37. Бережной А. Н. Redo Backup and Recovery. Изящный инструмент для снятия образов с дисковой подсистемы // Системный администратор. 2014. № 4. Режим доступа: <http://samag.ru/archive/article/2659>.
38. Бережной А. Н. Не теряя управления // Системный администратор. 2014. № 5. Режим доступа: <http://samag.ru/archive/article/2685>.
39. Бережной А. Н. Не теряя управления. Ч. 2: Обзор средств удаленного управления // Системный администратор. 2014. № 6. Режим доступа: <http://samag.ru/archive/article/2709>.

40. Бережной А. Н. Почему нельзя просто скопировать файлы. О резервном копировании, а также об инкрементальном парадоксе // Системный администратор. 2014. № 7–8. Режим доступа: <http://samag.ru/archive/article/2735>.
41. Бережной А. Н. Настройка современной системы хранения // Системный администратор. 2014. № 9. Режим доступа: <http://samag.ru/archive/article/2768>.
42. Бережной А. Н. Настройка современной системы хранения. Ч. 2: Добавляем поддержку дискового хранилища // Системный администратор. 2014. № 11. Режим доступа: <http://samag.ru/archive/article/2812>.
43. Бережной А. Н. О схемах ротации, управляемом хаосе, и почему нужно иметь больше носителей для копирования // Системный администратор. 2014. № 12. Режим доступа: <http://samag.ru/archive/article/2831>.
44. Бережной А. Н. Магнитная лента в резервном копировании // Системный администратор. 2015. № 1–2. Режим доступа: <http://samag.ru/archive/article/2856>.
45. Бережной А. Н. Диски против лент. Выбор средств резервного копирования // Системный администратор. 2015. № 3. Режим доступа: <http://samag.ru/archive/more/150>.
46. Бережной А. Н. Проектирование отказоустойчивых систем. Ч. 1: Термины и определения // Системный администратор. 2015. № 4. Режим доступа: <http://samag.ru/archive/article/2920>.
47. Бережной А. Н. Проектирование отказоустойчивых систем. Ч. 2: Этапы проектирования // Системный администратор. 2015. № 5. Режим доступа: <http://samag.ru/archive/article/2944>.
48. Бережной А. Н. Проектирование отказоустойчивых систем. Ч. 3: До и после внедрения // Системный администратор. 2015. № 6. Режим доступа: <http://samag.ru/archive/article/2959>.
49. Бережной А. Н. Поучительные истории о резервном копировании // Системный администратор. 2015. № 7–8. Режим доступа: <http://samag.ru/archive/article/2985>.
50. Бережной А. Н. Вопросы безопасности систем резервного копирования // Системный администратор. 2015. № 7–8. Режим доступа: <http://samag.ru/archive/article/2989>.
51. Бережной А. Н. Восстановление ИТ-инфраструктуры после ава-

- рии // Системный администратор. 2015. № 9. Режим доступа: <http://samag.ru/archive/article/3017>.
52. Бережной А. Н. Меры отказоустойчивости в дисковых массивах // Системный администратор. 2015. № 10. Режим доступа: <http://samag.ru/archive/article/3043>.
53. Бережной А. Н. Построение системы резервного копирования в многофилиальной структуре // Системный администратор. 2015. № 11. Режим доступа: <http://samag.ru/archive/article/3043>.
54. QuadStor about Virtual Tape Library. Режим доступа: <http://www.quadstor.com/virtual-tape-library.html>.
55. Host Bus Adapter (HBA) definition. Режим доступа: <http://searchstorage.techtarget.com/definition/host-bus-adapter>.

# Предметный указатель

## A

Active-Active, 43  
Active-Passive, 43

## B

Backup window, 41  
BCP, 36  
BEA, 33  
Business Continuity Planning, 36  
Business Environment Analysis, 33

## C

CA ARCserve Backup, 296  
Continuous backup, 99  
Copy-On-Write, 56

## D

D2D2T, 137  
Data Protector Express, 285  
Differential backup, 83  
Disaster Recovery, 37  
Disaster Recovery Plan, 232  
Disk-to-Disk-to-Tape, 137

## E

EMC, 294  
Enterprise, 66

## F

freezing, 117  
Full backup, 77

## G

Gparted, 268

## H

HA-cluster, 43  
High Availability cluster, 43

## I

Incremental backup, 79

## L

LAN-free, 174  
LTO, 145  
LTO-CM, 149

## M

Maximum tolerable period of  
disruption, 33  
Merge, 97  
MTPOD, 33

## N

NAS, 131

## O

Off-site, 42

## P

Primary-Standby, 44

## R

RAID, 46  
RAID-0, 46  
RAID-1, 47  
RAID-1E, 48  
RAID-5, 49  
RAID-6, 50

ReadyNAS Pro, 153  
 Recovery Point Objective, 34  
 Recovery Time Objective, 34  
 Redirect-On-Write, 58  
 Redo Backup and Recovery, 260  
 Replication, 40  
 Reversed incremental backup, 84  
 RPO, 34  
 RTO, 34

## **S**

SAN, 132  
 Server-free, 175  
 Service Level Agreement, 34  
 SLA, 34  
 snapshots, 56  
 Streamer, 141

## **A**

Автозагрузчик, 142  
 Альбом со схемами, 205  
 Аппаратный RAID, 53  
 Асинхронная репликация, 40  
 Аудит, 196

## **Б**

Бизнес-процесс, 32

## **В**

Виртуальный шторм, 122  
 Внедрение системы, 211

## **Г**

Глубина хранения, 42  
 График работ, 206

## **Д**

Дед-отец-сын, 94

Децентрализованная топология, 61  
 Дисковые хранилища, 128  
 Дифференциальное резервное копирование, 83  
 Дожигание, 191

## **З**

Заморозка, 117

## **И**

Инкрементальное копирование, 79  
 Инкрементальный парадокс, 81  
 Исполнительная документация, 213

## **Л**

Ленточная библиотека, 143  
 Ленточные системы хранения, 133

## **М**

Мгновенный снимок, 56  
 Мера терпения, 33

## **Н**

Надкусывание, 189

## **О**

Обратное инкрементальное резервное копирование, 84  
 Одиночный ленточный накопитель, 141  
 Окно бэкапа, 41  
 Опытная эксплуатация, 215  
 Отец-сын, 90

## **П**

Пилотный проект, 207  
 Полное резервное копирование, 77

Пояснительная записка, 205  
Приемо-сдаточные  
испытания, 214  
Проверка на тестовом стенде, 240  
Программный RAID, 52  
Псевдоцентрализованная  
топология, 62

## **Р**

Рабочая документация, 209  
Рабочие документы DRP, 237  
Разделение ролей, 221  
Репликация, 40

## **С**

Сетевая система хранения  
данных, 131  
Сеть хранения данных, 132  
Синхронная репликация, 40  
Система снятия образа, 259  
Смета проекта, 207  
Смешанная топология, 63  
Снапшот, 117  
Спецификация оборудования, 206  
Стратегия восстановления, 234  
Стример, 141  
Схемы ротации, 88

## **Т**

Тактика восстановления, 235  
Тестирование восстановления, 240  
Технический проект, 203  
Техническое задание, 199  
Техническое обслуживание, 215  
Технология RAID+, 54  
Технология блочного  
копирования, 107  
Технология файлового  
копирования, 106  
Топология резервного  
копирования, 60

## **У**

Условное тестирование, 240  
Учения на production-системе, 241

## **Ф**

Формирование кадрового  
резерва, 239

## **Ц**

Централизованная топология, 61  
Центр обработки данных  
(ЦОД), 37

Книги издательства «ДМК Пресс» можно заказать  
в торгово-издательском холдинге «Планета Альянс» наложенным платежом,  
выслав открытку или письмо по почтовому адресу:

115487, г. Москва, 2-й Нагатинский пр-д, д. 6А.

При оформлении заказа следует указать адрес (полностью),  
по которому должны быть высланы книги;  
фамилию, имя и отчество получателя.

Желательно также указать свой телефон и электронный адрес.

Эти книги вы можете заказать и в интернет-магазине: **www.aliants-kniga.ru**.

Оптовые закупки: тел. **(499) 782-38-89**.

Электронный адрес: **books@aliants-kniga.ru**.

Бережной Алексей Николаевич

## **Сохранение данных: теория и практика**

Главный редактор *Мовчан Д. А.*  
dmkpress@gmail.com

Корректор *Синяева Г. И.*

Верстка *Чаннова А. А.*

Дизайн обложки *Мовчан А. Г.*

Формат 60×90 1/16 .

Гарнитура «Петербург». Печать офсетная.

Усл. печ. л. 23. Тираж 200 экз.

Веб-сайт издательства: [www.dmk.ru](http://www.dmk.ru)