№12(85) декабрь 2009 Системный администратор

Системный администратор

ежемесячный журнал www.samag.ru

№12(85) декабрь 2009

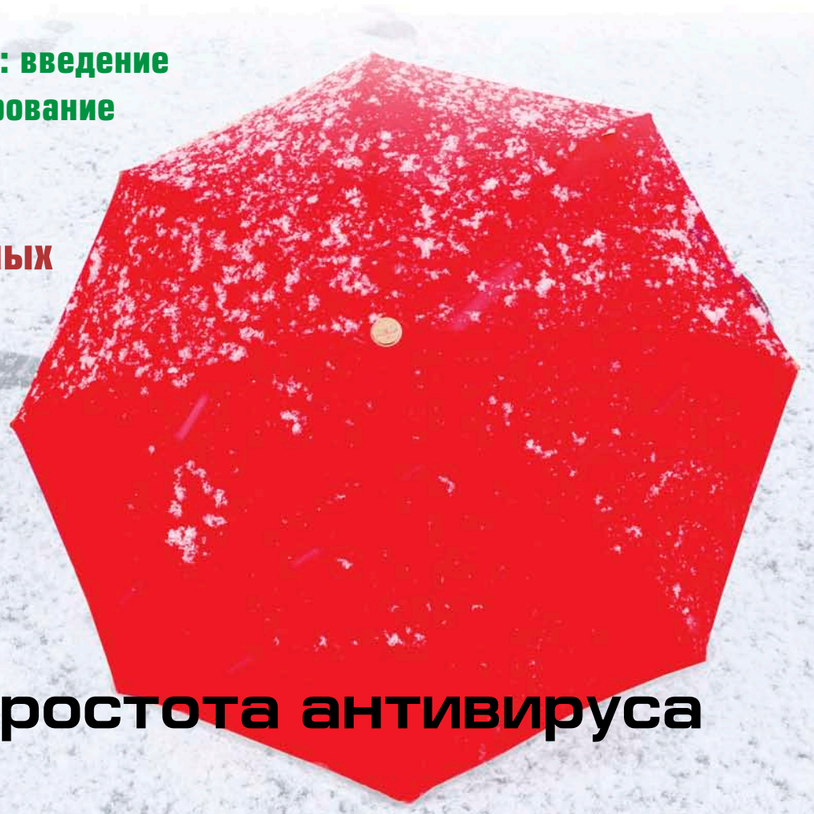
Обработка видео средствами веб-сервера

Теория и практика восстановления Exchnage Server

Управление мобильными устройствами на предприятии

Один UPS на двоих – FreeBSD в домене Windows

Вычислительная модель: введение в системное программирование



И снова о персональных данных

Мнимая простота антивируса

Третья попытка Пола Галвина

ISSN 1813-5579
09012

Системный администратор

УЧРЕДИТЕЛИ ИЗДАНИЯ
Частные лица

Генеральный директор
Владимир Положевец
Главный редактор
Галина Положевец
chief@samag.ru
Технический директор
Владимир Лукин

Главный редактор электронного приложения «Open Source»
Дмитрий Шурупов
osa@samag.ru

Дизайн-макет
Марина Рязанцева
Дмитрий Бессонов
Иллюстрации
Виктор Чумачев

Над номером работали:
Рашид Ачилов
Алексей Барабанов
Александр Емельянов
Валентин Синицын
Кирилл Сухов

Реклама и PR-служба
Дарья Зуморина, reklama@samag.ru,
Полина Гвоздь, pr@samag.ru,
тел./факс: (495) 628-82-53 (доб.120)

Распространение
Светлана Зобова
(495) 628-82-53 (доб.120)

Адрес редакции
107045, г. Москва, Ананьевский переулок, дом 4/2, стр.1,
тел./факс: (495) 628-82-53 (доб.120)
Сайт журнала: www.samag.ru

Издатель
ООО «С 13»

Отпечатано в типографии
ООО «Периодика»
Тираж 17000 экз.
Тираж электронной версии 62000 экз.

Всем – спасибо!

Накануне праздников не хочется думать о проблемах. Чем бы ни занимался, там, в подсознании, уже зреет предвкушение новогоднего веселья, смеха, музыки. И невольно торопишь события, подгоняешь дела, чтобы приблизить время всепобеждающей радости, которая бывает только в детстве.

Да, мы все под Новый год становимся немного детьми, ждем от встреч – только приятного, от новогодней елки – волшебства, от судьбы – чудесных подарков. Пусть же станут явью ваши надежды, уважаемые читатели! Пусть никого из вас не обманет новогодняя ночь. А первый день 2010-го положит начало счастливым переменам. Новый год – это чистый лист, что мы на нем напишем, то и сбудется.

Уходящий 2009 год стал для нас началом преобразований. Несмотря на кризис (а возможно благодаря ему), мы решили изменить многие рубрики, пригласить новых авторов, расширить круг обсуждаемых тем. Надо сказать, что вы очень помогли нам в этом. Барометром для нас служит оживленная полемика на форуме журнала – часто резкая, нелицеприятная, порой подкупающе доброжелательная (спасибо, спасибо!).

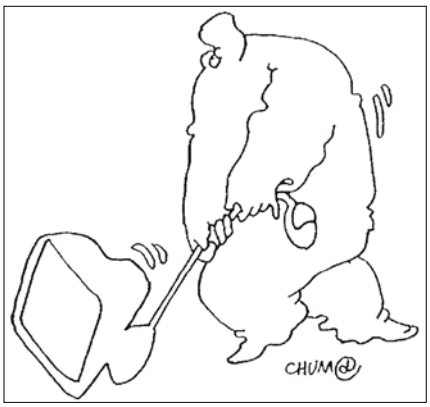
Все перемены в политике и содержании журнала имеют только одну цель – максимально улучшить «Системный администратор», сделать его настольным изданием для всех сисадминов страны. Вот такую мы поставили себе скромную задачу. Но без вас нам ее не выполнить. Пожалуйста, не изменяйте себе и в следующем году: читайте, ругайте, предлагайте, хвалите (если есть за что), но только не молчите! Чтобы «сверять часы» – наше представление, как делать журнал, и ваши ожидания от него – обещаем в 2010-м чаще встречаться с авторами и самыми активными читателями «Системного администратора». Ведь друзья дают нам жизненную силу. Так будем же друзьями!

Коллектив редакции «Системного администратора» поздравляет каждого из тысяч наших читателей с наступающими праздниками! До новых встреч в Новом году!

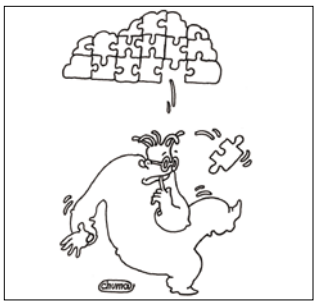
Галина Положевец, главный редактор

Где купить «Системный администратор»:
- г. Москва, выставочный компьютерный центр «Савеловский»;
- г. Москва, редакция журнала, Ананьевский пер, д. 4/2 стр. 1.

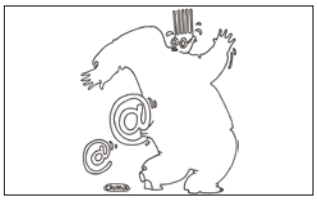
Подробную информацию о подписке смотрите на стр. 112.



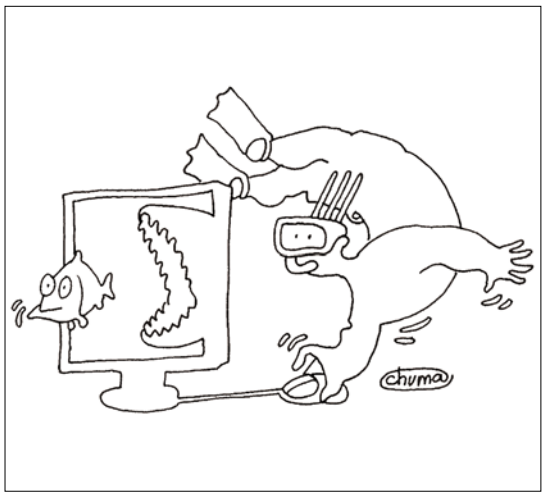
12



18



36



56

События

06 Все под контролем. Windows Server 2008 R2 экономит и деньги, и время. С 9 по 13 ноября в Берлине в выставочном центре Messe Berlin проходил ежегодный форум для разработчиков и ИТ-специалистов – Tech Ed Europe 2009, на котором компания Microsoft объявила о начале продаж Exchange Server 2010.

Полина Гвоздь

08 Форум Cisco Expo-2009 переписал собственные рекорды. Московский форум Cisco Expo-2009 ознаменовался беспрецедентным числом участников и целым рядом новшеств.

09 Информбюро

Звезды «СА»

10 2010-й – нас ждет много интересного… 2009 год был непростым. Однако нельзя сказать, что профессиональная жизнь системных администраторов была скучной и монотонной – это невозможно по определению.

Закон есть закон

12 Перегретая тема. И снова о персональных данных. На вопросы читателей «СА» отвечают сопредседатели комитета по информационной безопасности Союза ИТ-директоров России Виктор Минин и Юрий Шойдин.

17 Bugtraq

Кафедра

18 Вычислительная модель. Введение в системное программирование. Как установить точно, чем занимается системный администратор? Не определившись с этим, невозможно автоматизировать работу сисадмина.

Алексей Барабанов

23 Вызов XXI века: каким быть программному обеспечению? В конце 2006 – начале 2007 года наше общество всколыхнула ситуация, сложившаяся вокруг директора школы села Сепыч Верещагинского района Пермского края Александра Михайловича Поносова.

Игорь Штомпель

Острый угол

28 Мнимая простота антивируса. Что может быть ясней и спорней? Это слово найдешь в лексиконе любого ИТ-специалиста, а область знаний, казалось бы, давно изучена вдоль и поперек. Так, да не так!

Константин Черезов

31 Стоит только чихнуть… Каждый сегодня может стать носителям вируса. Тема так сильно перегрета, что любое высказывание может ее едва ли не взорвать. Но дело не в эмоциях.

Валерий Андреев

32 Безопасное завтра. Полной коллекции вирусов ни у кого нет. Проблемы антивирусного ПО обсуждают эксперты в области защиты от вирусов и вторжений.

34 Какие требования вы предъявляете к антивирусному ПО? На вопрос «СА» отвечают ИТ-специалисты.

Администрирование

36 После катастрофы. Теория и практика восстановления Exchnage Server. Почта в организации является одним из основных рабочих инструментов. Что будет, если вы его лишитесь? Готовы ли вы к такому повороту событий?

Михаил Даньшин

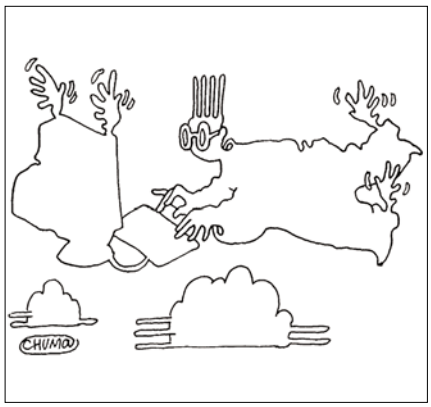
40 Расширение возможностей при работе с сетевыми хранилищами NETGEAR ReadyNAS. Не только приобрести и подключить сетевое хранилище, а попытаться добиться от него максимальной функциональности – вот профессиональный подход.

Алексей Бережной

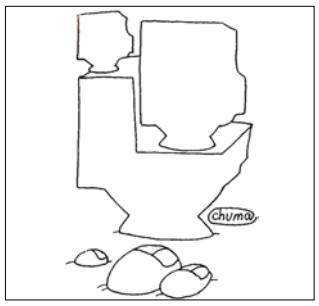
44 Технология Oracle Streams. Настраиваем потоки данных, экономим время и деньги. Режим DownStream технологии Oracle Streams помогает повысить эффективность использования ресурсов в высоконагруженных информационных системах.

Антон Пищулин

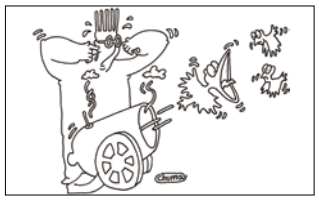
50 WDS поможет. Установка операционных систем. Часть 1. Разворачивание операционных систем является одной



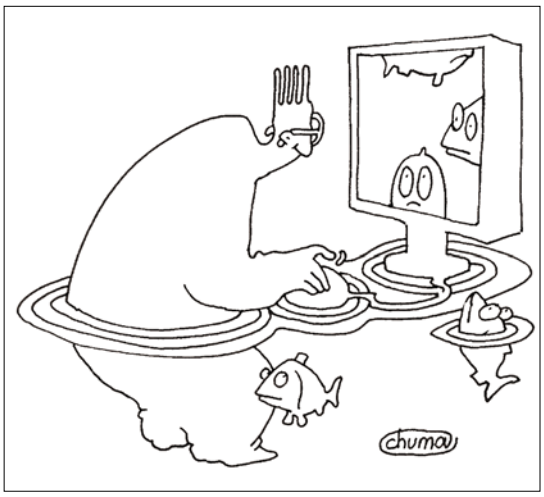
70



76



82



88

из задач любого системного администратора. Посмотрим, какие средства для этих целей предлагает Windows Server 2008.

Андрей Бирюков

56 Собираем свой дистрибутив с Calculate Linux Scratch. Практически каждый пользователь Linux хоть раз да собирал свой дистрибутив. Разработчики Calculate Linux предлагают свой вариант.

Сергей Яремчук

61 Sun VirtualBox как персональная система виртуализации. Виртуальная машина на компьютере системного администратора – давно не экзотика. А новый VirtualBox от компании SUN – отличный выбор для создания собственной системы виртуальных машин.

Алексей Бережной

66 Один UPS на двоих. FreeBSD в домене Windows. Ситуация, когда к одному UPS подключено несколько серверов, – не исключение, а скорее правило. Как вовремя отключить сервер, если программа для работы с данным UPS существует только для Windows?

Рашид Ачилов

70 Удобно, безопасно, недорого. Управление мобильными устройствами на предприятии. Мобильные устройства все чаще становятся заменой обычных ПК вне офиса, но можно ли управлять ими наравне с ПК, сохраняя удобство использования.

Алексей Ватутин

73 Тратим меньше, спим больше с VMware Sphere 4.0. Часть 2. Продолжение цикла статей по разворачиванию кластерной системы виртуальных серверов масштаба предприятия. Построение отказоустойчивых виртуальных машин.

Виталий Банковский

76 Настройка Webacula. Веб-интерфейс к Bacula. Bacula – популярная Open Source-система резервного копирования. Использование веб-интерфейса Webacula позволит сделать работу с ней на порядок удобнее.

Сергей Яремчук

Безопасность

80 Dr.Web CureNet!. От идеи до воплощения. Идея создания Dr.Web CureNet! созрела в компании «Доктор Веб» летом 2009 года. По сути, хотелось создать продукт для страховки и усиления безопасности корпоративных сетей, в которых используется антивирус других производителей.

Павел Плотников

Программирование

82 Компонентная модель EJB. Преимущества и недостатки. В российской корпоративной среде возрос интерес к платформе Java 2 Enterprise Edition (J2EE) и, в частности, к входящей в её состав компонентной модели Enterprise Java Bean (EJB).

Арсен Ибрагимов

86 Bugtraq

Веб

88 Создаем свой YouTube. Как обработать видео средствами веб-сервера. Каждый веб-разработчик должен иметь представление о том, как можно обработать видео средствами веб-сервера и автоматизировать конвертирование из одного формата в другой.

Александр Майоров

97 Bugtraq

Ретроспектива

98 Тотальная мобилизация. Третья попытка Пола Галвина. Редкое упорство, которое многие называли упертостью, помогло создателю компании Motorola завоевать мир.

Владимир Гаков

103 Bugtraq

Творчество админа

104 POWER OFF. Рассказ.

Станислав Шпак

108 Содержание 2009

Участвуй сам и расскажи друзьям! Как получить Адмиский приз?

Редакция «СА» продолжает разыгрывать Адмиский приз, и приглашает к участию новых игроков. Вам необходимо зарегистрироваться на сайте www.samag.ru и активировать код, который можно получить, купив номера журнала (№№7-12, 2009). Чем больше у вас заветных кодов, тем выше шансы стать победителем розыгрыша. Дополнительные коды смогут получить самые активные участники форума на сайте www.samag.ru.

Успехов и удачи!

Адмиский Приз

Розыгрыш будет проходить в три этапа:

I – участвуют коды из журналов №7, 8, 9, полученные с июля по сентябрь 2009 г.

II – участвуют коды из журналов №10, 11, 12, полученные с октября по декабрь 2009 г.

III – участвуют коды из всех шести номеров журнала за 2-е полугодие 2009 г.

Адмиский Приз

Админский Приз

Призы:

I этап:

1 место – приз-сюрприз
2 место – учебные курсы
3 место – пакет программного обеспечения
4 место – почтовый сервер на 50 пользователей
5 место – виртуальные выделенные серверы

II этап:

1 место – приз-сюрприз
2 место – учебные курсы
3 место – пакет программного обеспечения
4 место – почтовый сервер на 50 пользователей
5 место – виртуальные выделенные серверы

III этап:

1 место – приз-сюрприз
2 место – учебные курсы
3 место – пакет программного обеспечения
4 место – почтовый сервер на 50 пользователей
5 место – виртуальные выделенные серверы

Специальный утешительный приз – электронная книга

Ваш код для участия
в розыгрыше призов:

Админский Приз

Визитка

ПОЛИНА ГВОЗДЬ, спецкор «Системного администратора», Берлин-Москва

Все под контролем

Windows Server 2008 R2 экономит и деньги, и время

С 9 по 13 ноября в Берлине в выставочном центре Messe Berlin проходил форум для разработчиков и ИТ-специалистов – Tech Ed Europe 2009. Microsoft объявила о начале продаж Exchange Server 2010

Руководитель отдела бизнес-систем Microsoft Стефан Элоп (Stephen Elop) охарактеризовал Microsoft Exchange Server 2010, как продукт поколения решений, созданных для значительного повышения бизнес-продуктивности и снижения расходов компаний. Срок окупаемости нового продукта составит не более шести месяцев. Его достоинства уже оценили Bank of America Corp., NEC Philips, Subaru Canada Inc., Telekom Austria Group и другие лидеры мирового экономического рынка.

Особенности нового Windows Server 2008 R2

О них по просьбе «СА» рассказывает директор Windows Server Marketing **Джейсон Гермитейж** (Jason W. Hermitage).

– Я работаю в подразделении Microsoft Business Server уже семь лет. В мои обязанности входят Open Marketing для Windows Sever 2008 R2, обеспечение безопасности ядра Windows Server, а также решение проблем межсерверной безопасности.

В связи с глобальным финансовым кризисом многие клиенты нашей компании пытаются сократить расходы, повышая одновременно эффективность своего бизнеса. Безусловно, мы учитывали их желание, создавая новый продукт, при этом стараясь не нарушать принцип «цена – качество». В конечном итоге мы внесли большое количество изменений в данный продукт, так как наши заказчики просили об этом.

Если заглянуть в будущее, которое открывается с Windows Server 2008 R2, то вы увидите, что виртуализация занимает там не последнее место. Мы обеспечили более высокий уровень администрирования, добавив технологию Live Migration. Я уверен, что ваши читатели заинтересованы в этой технологии.

Live Migration предоставляет возможность переносить виртуальные машины с одного физического узла на другой, не прерывая обслуживание подключенных клиентов, что обеспечивает уровень доступности и возможность для переоценки резервов системы. Таким образом, Live Migration позволяет существенно сократить ИТ-операции. Это одна из самых важнейших особенностей Windows Server 2008 R2.

Другая особенность – Remote Desktop Services (RDS). Это переименованные Terminal Services из арсенала Windows Server 2008, позволяющие удаленно запускать приложения на одном компьютере, а управлять ими с другого. Remote Desktop Services предоставляют администраторам и пользователям средства и возможности для создания наиболее надежной среды доступа в любых сценариях развертывания.

Чтобы расширить возможности Remote Desktop Services, корпорация Microsoft и ее партнеры разработали Virtual Desktop Infrastructure (VDI). VDI основан на способности выполнять приложения своих клиентов, других персональных компьютеров, на централизованном сервере, так, что вся информация, все образы, клиентские образы – будь это Windows Vista или Windows 7 – работают на централизованном сервере и передаются на персональный компьютер. Одним словом, все централизованно. И Remote Desktop Services помогает нам в этом.

Что касается сферы управления, то хочу обратить внимание ваших читателей на PowerShell 2.0. В PowerShell есть язык кодирования, который позволяет нашим заказчикам выполнять небольшие задачи. Его достоинство в том, что он значительно экономит время ИТ-процесса, не надо все время кликать туда-сюда… Вы можете автоматизировать процесс путем запуска команды ни один раз во множестве разных мест.

Кроме того, новый продукт позволяет экономить электроэнергию. Используя только модуль для сервера, мы можем увеличивать или уменьшать потребление энергии в зависимости от ситуации, что, безусловно, приносит реальную пользу заказчику, снижая его затраты.

У нас также есть функция, которая называется Core Parking. Она дает дополнительные возможности сокращать потребление энергии, которая расходуется впустую. Это довольно большое преимущество, особенно ощу-

тимое, если сравнивать Windows Server 2003 с Windows Server 2008. Если вы сделаете апгрейд, то при тех же задачах, на том же аппаратном обеспечении с Windows Server 2008 R2 вы можете сэкономить до 18% ваших расходов на оплату энергии. Это неоспоримая ценность для заказчика.

Другое значительное изменение – это возможность для сервера получить преимущества для большего количества процессов. Он сможет соответствовать уровню SAN-систем высшего класса, отвечать требованиям широкого круга бизнес-приложений.

Что касается Windows 7, то теперь вы сможете обеспечить прямой доступ к Windows Server, который дает возможность подключиться даже к самой удаленной корпоративной сети. Без VPN. Это означает, что всякий раз, когда я подключен к Сети, я могу немедленно начать работать и получить доступ к любому файлу в моей сети без регистрации и разрывов соединения.

Это прежде всего является преимуществом для опытных пользователей, теперь они всегда могут получить доступ к своим документам, к корпоративным (рабочим) ресурсам без дополнительных программ. PowerShell умеет подключаться к этим ресурсам.

И последняя особенность, на которую хотел бы обратить внимание, это реальное преимущество для пользователей, где бы они ни находились, иметь доступ ко всем необходимым им файлам.

Это также реальное преимущество для всего ИТ-процесса, так как теперь можно удаленно управлять персональным компьютером, поскольку он все время находится во внутренней сети. Можно безбоязненно пересылать конфиденциальную информацию, факсовые сообщения, электронные сообщения, так как компьютер надежно защищен.

Каким же образом заказчики получают преимущество от этих особенностей, сохраняя деньги и повышая продуктивность своего бизнеса?

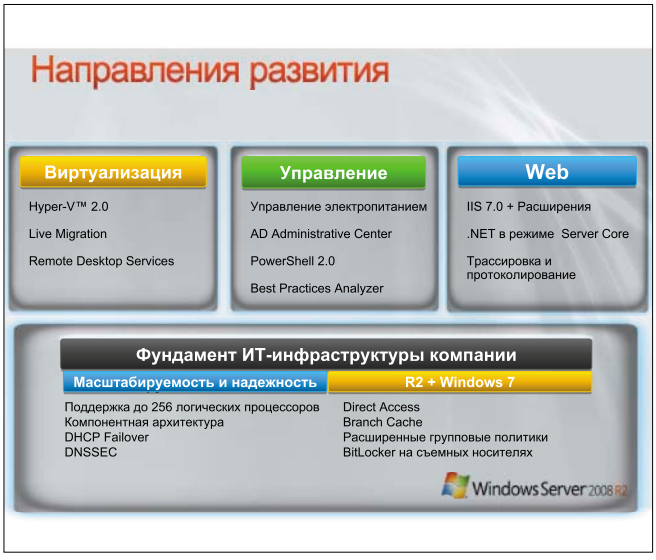
Возьмем одну из широко применяемых функций – BranchCache. Что конкретно выполняет BranchCache? Это сервер, установленный в местном отделении компании, который в основном кэширует часто запрашиваемые файлы. Представьте, что вы находитесь в отделении компании и записываете 20 Мб видео. В первый раз они идут из отделения через WAN-шлюз. Кешируются на локальном сервере, затем пользователь получает этот файл. Во второй раз файл можно получить напрямую: вместо того чтобы отсылать его через WAN, он идет прямо к серверу и локально передается по сети. Таким образом, экономится немало средств для удаленных и разбросанных офисов, которые тратят значительные суммы при использовании WAN.

Представьте, что вы записываете 20 Мб видео. В первый раз они идут из отделения компании через WAN-шлюз. Кешируются на локальном сервере, затем пользователь получает этот файл. Во второй раз файл можно получить напрямую

Только представьте на минуту: пятидесяти сотрудникам в местном офисе уже нет необходимости направлять один и тот же файл пятьдесят раз через WAN. Для клиентов это немалая экономия и времени, и денег. Я бы сказал, что это огромное преимущество, которое отличает Windows Server 2008 R2 от других продуктов. EOF

Рисунок 1. «Новая эффективность» по Microsoft

Рисунок 2. Направления развития Windows Server 2008 R2



Форум Cisco Expo-2009
переписал собственные рекорды

Московский форум Cisco Expo-2009 ознаменовался беспрецедентным числом участников и целым рядом новшеств

12-14 октября в московском Центре международной торговли (ЦМТ) прошла юбилейная конференция по информационным технологиям Cisco Expo-2009. Десятый по счету ежегодный форум побил собственные рекорды посещаемости и числа партнеров, спонсоров и средств массовой информации, оказавших поддержку этому мероприятию, еще в предыдущие годы снискавшему репутацию крупнейшего в индустрии информационно-коммуникационных технологий стран СНГ. В этот раз в московской Cisco Expo приняли участие 2298 ИТ-специалистов и журналистов – на 130 больше, чем год назад.

Впервые в истории этих форумов была также организована онлайн-трансляция с помощью интернет-сервиса Cisco WebEx. Это позволило более 140 ИТ-специалистам, не сумевшим попасть на конференцию, в режиме реального времени прослушать пленарные доклады топ-менеджеров компании Cisco и ее партнеров.

В полном соответствии с лозунгом московской Cisco Expo-2009 («Знание – сила») участникам конференции была предложена небывало обширная и разнообразная программа в виде 100 с лишним выступлений, демонстраций и презентаций в рамках девяти технологических потоков (сетевая инфраструктура, решения в области центров обработки данных, решения для операторов связи, информационная безопасность, унифицированные коммуникации, центры обработки вызовов, мобильные и беспроводные решения, оптические сети и впервые включенный в программу поток «Интегрированные решения», рассчитанный на компании, занимающиеся разработкой специализированных решений на базе платформ Cisco). Кроме того, отдельный тематический блок был посвящен новому направлению в деятельности Cisco в области физической безопасности – решениям, предназначенным для организации IP-видеонаблюдения.

Как и в предыдущие годы, для посетителей конференции была организована выставка продуктов Cisco и партнеров форума, где демонстрировались новые технологии и решения для оптимизации предоставляемых услуг и повышения качества обмена информацией. В частности, впервые широкой публике были показаны в действии системы Cisco TelePresence 3000 и Cisco TelePresence 500, позволяющие проводить виртуальные встречи с эффектом присутствия в режиме реального времени. Перед закрытием конференции с помощью обеих систем в ЦМТ был организован первый в мире трансконтинентальный виртуальный концерт с участием южноафриканской рок-группы The Parlotones.

Еще одним новшеством московских Cisco Expo стал магазин, где участники форума могли приобрести переведенную на русский язык специализированную литературу, выпущенную издательством Cisco Press, и разнообразные устройства для дома и офиса производства компаний Jabra, Linksys и Plantronics.

Второй год подряд в рамках конференции было организовано тестирование в соответствии с программой профессиональной сертификации Cisco. Этой возможностью воспользовались 143 участника форума. Организацию мобильного тестового центра и проведение экзаменов взяла на себя компания REDLAB/REDCENTER – авторизованный учебный центр Cisco уровня CLSP, авторизованный центр тестирования VUE.

Работу форума освещали 93 журналиста из Воронежа, Екатеринбурга Минска, Москвы, Самары, Санкт-Петербурга, Саратова, Ташкента, Уфы и Челябинска. Для них были организованы пресс-конференция с участием топ-менеджеров Cisco и компаний-партнеров форума, круглые столы по интегрированным решениям и центрам обработки данных. EOF

В случае появления организации Java Foundation компания SAP готова проводить значительные вливания в технологию Java для ее дальнейшего развития

Визитка

ДМИТРИЙ ШУРУПОВ,
ведущий рубрики

SAP: Java нуждается в независимой организации

В начале ноября Вишал Сикка (Vishal Sikka), технический директор SAP, раскритиковал существующую структуру управления развитием языка программирования Java и доминирование компании Sun Micrososystems в JCP (Java Community Process). В своем блоге Сикка сообщил, что в SAP сделали ставку на Java как на основу своего бизнеса еще в 2001 году, добавив, что сам он считает Java «кровью всей ИТ-индустрии». Затем он отметил, что будущее Java вне зависимости от принадлежности этого языка Sun или Oracle должно управляться открытой структурой, в которой не будет доминировать ни одна из корпораций. Сикка предлагает для новой организации Java Foundation модель управления, схожую с Eclipse Foundation, которая была создана в 2004 году, когда IBM передала открытый исходный код интегрированной среды разработки Eclipse IDE в руки одноименной организации. Вишал заявил, что в случае появления такой организации компания SAP готова проводить значительные вливания (как финансовые, так и инженерные) в технологию Java для ее дальнейшего развития. Позже, 30 ноября, SAP обнародовала комментарий по случаю подготовки Java EE 6, в котором поприветствовала некоторые изменения в JCP, однако выразила недовольство по поводу того, что Sun не подготовила обещанные «полные лицензионные условия» по Java EE 6 TCK до начала двухнедельного голосования за спецификацию Java EE 6. В итоге SAP воздержалась от голосования за спецификацию, напомнив всему сообществу о своей позиции по будущему Java. EOF

Релиз GNOME 3.0 отложили до сентября

После непродолжительного обсуждения разработчики популярной графической среды GNOME пришли к выводу, что стоит перенести следующий крупный релиз 3.0 на сентябрь 2010 года. Винсент Унц (Vincent Untz) из GNOME Release Team разослал по почтовой рассылке письмо, в котором просил всех разработчиков поделиться своим видением о том, когда GNOME 3.0 может быть готов. Ответственные за документацию GNOME сообщили, что Yelp 3.0 средство GNOME для просмотра справки будет готово к марту, однако успеть сделать саму документацию вряд ли получится до сентября. Команды, занимающиеся accessibility, keyring и Evolution, ответили в подобном ключе: предварительную версию можно успеть и к марту, но вот полноценный стабильный релиз лучше отложить до сентября. Оуэн Тэйлор (Owen Taylor) из команды GNOME Shell ключевого компонента GNOME 3.0 сообщил, что к марту будет готова только бета-версия GNOME Shell, а для завершения работы над ней потребуются еще месяцы тестирования и совершенствования. Впрочем, такую тщательность в подготовке релиза стоит скорее отнести к плюсам: GNOME традиционно славится стремлением к выпуску хорошо проверенных компонентов среды. Подводя итоги, Винсент высказал свое оптимистичное пожелание в следующем письме: «Давайте сделаем 2010 год сказочным для GNOME!». EOF

KDE провел ребрендинг своего названия

Проект популярной графической рабочей среды KDE провел ребрендинг с целью ассоциировать термин «KDE» в первую очередь с людьми, которые создают программные продукты, а не с результатом их творчества. Ребрендинг KDE заключается в том, что отныне проект перестанет использовать свою историческую расшифровку «K Desktop Environment». Кроме того, под «KDE» будут пониматься не только (и не столько) технологии, создаваемые сообществом проекта, но и само сообщество. Вместе с тем было решено использовать различные бренды для программного обеспечения, на которое раньше ссылались обобщенными терминами. Так, например, понятие KDE Workspaces отныне разделено на «KDE Plasma Desktop» и «KDE Plasma Netbook», а технологии KDE для создания приложений отныне именуются, как «KDE Platform». Таким образом, недавно вышедший релиз «KDE 4.3» является набором из Workspaces, Applications и Platform, а следующий релиз получит название «KDE Software Compilation 4.4». EOF

2010-й – нас ждет много интересного...

2009 год был непростым. Однако нельзя сказать, что профессиональная жизнь системных администраторов была скучной и монотонной — это невозможно по определению

Мы попросили наших постоянных авторов подвести итоги 2009 года и оценить перспективы следующего:

- **Какое профессиональное событие года вы считаете наиболее интересным с точки зрения сисадмина?**
- **Ваше самое большое профессиональное достижение года?**
- **Ваш прогноз на 2010 год.**
- **Ваши новогодние пожелания коллегам.**

АНДРЕЙ БИРЮКОВ, специалист по информационной безопасности. Работает в крупном системном интеграторе. Занимается внедрением решений по защите корпоративных ресурсов

«Сохраняйте оптимизм!»

Событие. На мой взгляд, самое интересное событие 2009 года – это выход Windows 7. Громоздкая и требовательная к ресурсам Windows Vista, к тому же имеющая ряд проблем с совместимостью, является не самым лучшим решением для настольных компьютеров. В Windows 7 ряд недостатков предыдущей версии был устранен разработчиками, что не может не радовать.

Прогноз. Несколько лет назад в России вступил в действие федеральный закон №152 «О персональных данных». С 1 января 2010 года информационные системы на предприятиях должны быть приведены в соответствие с требованиями этого закона. Понятно, что не все успели это сделать. Но требования данного закона могут существенно повлиять как на ИТ-инфраструктуру многих предприятий и должностные обязанности системных администраторов, так и на российский ИТ-рынок в целом.

Личные достижения. Так как я являюсь специалистом по информационной безопасности, то все достижения связаны с решением задач в этой области.

Новогодние поздравления. В новом году желаю всем коллегам успехов и профессионального развития, ведь настоящим профессионалам никакой кризис не страшен! Не терять оптимизма, так как все мы понимаем: в ближайшее время развитие рынка информационных технологий продолжится.

АЛЕКСАНДР ЕМЕЛЬЯНОВ, инженер группы информационных технологий Владимирского филиала ООО «Татнефть-АЗС-Запад»

«Не стойте на месте!»

Событие. Поскольку я отношусь к администраторам того полушария ИТ-сферы, где правит балом Microsoft, то, безусловно, важнейшим событием для меня был выход в свет Windows 7, а также нового релиза Windows Server 2008 R2. Но однозначно сказать, порадовало это событие меня или нет, не могу – считаю, что нужно подождать, поработать и посмотреть, насколько новый продукт получился удачным.

Пока есть первые неприятные звоночки (проблемы с новым протоколом SMB 2.0). Это наводит на нехорошие воспоминания о начале эры Windows Vista (которую я до сих пор игнорирую), несмотря на громкие заявления о рекордных продажах Windows 7.

Прогноз. Отрасль информационных технологий стремительно развивается, появляясь, казалось бы, в далеких от нее сферах. Так вот, мне кажется, что актуальным будет для любого айтишника не стоять на месте, а двигаться вперед, постигая новые грани информационных технологий!

Личные достижения. Я не сторонник получения каких-то регалий, например, всевозможных сертификатов. Но, как среднестатистический сисадмин, каждый день стараюсь открывать что-то новое и расти профессионально. Журнал «Системный администратор», кстати, мне в этом помогает.

Новогодние поздравления. Всем своим друзьям и близким людям я желаю здоровья – физического и эмоционального!

ВИТАЛИЙ БАНКОВСКИЙ, технический директор, автор более 20 статей, главный двигатель прогресса в одной из интернет-компаний

«Просыпайтесь и действуйте!»

Событие. Самым важным и интересным профессиональным событием, на мой взгляд, стало то, что разработчики ядра Linux осознали важность не только функционала последнего, но и стабильность, и производительность, таким образом, значительно подняв репутацию Linux в глазах системных администраторов.

Прогноз. Без прогноза.

Личные достижения. Наконец-то я осознал две простые вещи – лучше вложить средства в более производительное оборудование и построение кластерных систем, чем быть «пожарником» всю оставшуюся жизнь…

Новогодние пожелания. Коллегам желаю «пробуждения» инвесторов и персонала, а также появления новых хороших проектов!

АЛЕКСЕЙ БАРАБАНОВ, системный администратор, аутсорсер и консультант по вопросам применения информационных технологий. Хобби – программирование

«Преодолевайте зависимость и страх!»

Событие. Выделять событие, на мой взгляд, некорректно, так как оценка все равно будет субъективной. Поэтому я могу говорить лишь о явлении, которое стало очевидным. В 2009 году остановился количественный рост спама, он ушел из списка важных угроз. А вот бот-неты, напротив, заняли первое место. Важно то, что больше нельзя утверждать, что бот-неты создаются для рассылок спама. Тогда зачем же их создают? Если нет коммерции, значит, их создают государственные структуры. Оцените парадокс: ОС производятся в США, а управляются из стран Юго-Восточной Азии. Предположу, что первая же сетевая «война ботов» поставит крест на проприетарных операционных системах. Так сбудется мечта RMS – ОС, доступная как воздух.

Прогноз. В наступающем году очень важно будет, кто победит: те, кто за интеграцию, или те, кто за изоляцию? Те, кто внедряет открытые стандарты, например доступные CRM, или те, кто внедряет федеральный закон №152 «О персональных данных» в исходной редакции, который ставит барьеры для создания всякой автоматизации не по сценарию ФСТЭК, одновременно с тем, что vtiger CRM ФСТЭК-эдишн пока даже не в проекте.

Личные достижения. Я успешно преодолел чрезмерную привязанность к openSUSE, перевел все свои проекты на CentOS, и новые серверы уже начали разъезжаться по стране.

Новогодние пожелания. Желаю в новом году всем также успешно преодолевать собственные профессиональные зависимости и страхи: зависимость от Windows и страх Linux, зависимость от одного работодателя и страх аутсорсинга и вообще все, что сдерживает профессиональное развитие!

РАШИД АЧИЛОВ, поклонник FreeBSD с 14-летним опытом использования ее в совмещенных с Windows сетях и сторонник Open Source. Администратор сетей и средств защиты крупной торговой сети

«Больше коннектов – стабильных и разных!»

Событие. Поскольку я большую часть времени занимаюсь OpenSource Software, неудивительно, что мне интересны события, связанные с ним. Для меня наиболее важными, пожалуй, были выход FreeBSD 7.2 и OpenOffice 3.1.1. В первой – наконец-то появился планировщик, работающий на многоядерных процессорах, а во втором – поддержка формата Office Open (.docx)

Прогноз. Как мне кажется, борьба с контрафактом станет самым актуальным трендом будущего года. Особенно если государство (мэрия, милиция, здравоохранение и т.д.) полностью перейдет на лицензированные продукты. Естественно, это вызовет рост популярности администраторов Lunix и UNIX-систем. Впрочем, хорошие специалисты всегда ценятся.

Личные достижения. Мне трудно выделить что-то одно, в профессиональном смысле год был весьма продуктивным. Пожалуй, это то, что Perl и PHP перестали быть для меня «загадочным не пойми чем», а стали языками программирования, на которых можно писать, и с помощью которых можно будет решить задачи, ранее казавшиеся неподъемными.

Новогодние поздравления. В Новый год хотелось бы всем пожелать – больше коннектов, стабильных и разных!

КИРИЛЛ СУХОВ, веб-программист в дистрибьюторской компании MICS. Занимаюсь проектированием и разработкой различных интернет-сервисов. Круг интересов банален – веб-технологии, RIA, Framework-среды

«Не теряйте интереса к делу!»

Событие. Так как я разработчик, а не системный администратор, со стороны мне кажется, что, скорее всего, наиболее значимым было появление новых операционных систем семейства Windows – Windows 2008 Server и Windows 7. Лично мне больше интересно «взросление» различных технологий Rich Internet Application – появление стабильной JavaFX, новых версий Microsoft Silverlight и фрэймворка Google Web Toolkit. Впрочем, дело не в конкретной среде, а в тенденции, которая, как мне кажется, набрала обороты.

Прогноз. Я думаю, что вышеупомянутая тенденция сохранится. Веб-технологии и веб-приложения будут завоевывать мир, теснить традиционный подход к разработке ПО. Chome OS, JavaFX TV... в общем, нас ждет много интересного.

Личные достижения. На этот вопрос так и тянет ответить – сохранил свое рабочее место! На самом деле один интересный проект я бы выделил – разработка ядра браузерной игры. Может, это покажется несерьезным, но я вполне могу говорить об этом как о самом большом достижении за 2009 год. Экономические условия сказывались, поэтому каких-то масштабных проектов, о которых стоит вспомнить, практически не было.

Новогодние поздравления. Коллегам желаю не потерять интереса к своей работе, к делу, которым занимаются! С наступающим!

Перегретая тема
И снова о персональных данных

На вопросы читателей «СА» отвечают сопредседатели комитета по информационной безопасности Союза ИТ-директоров России Виктор Минин и Юрий Шойдин

До наступления ответственности по Федеральному закону от 27 июля 2006 г. № 152-ФЗ «О персональных данных» остаются считанные дни. Специалисты ИТ-сообщества понимают, что аттестация своей информационной системы неизбежна.

В такой ситуации, когда тема персональных данных, по сути, «перегрета», абсолютно естественно, что некоторые коммерческие компании за баснословные деньги предлагают взволнованным руководителям ИТ и подразделений безопасности свои услуги по подготовке и аттестации ИС предприятия. Давайте постараемся разобраться в этом вопросе и понять, что можно сделать самостоятельно, а что желательно отдать стороннему консультанту.

Для начала синхронизируем понятия и разберемся, кто попадает под действие данного закона. Ведь руководители очень многих компаний склонны считать, что их организации не являются операторами персональных данных и их настоящий закон не касается.

– Кто все-таки по закону считается оператором персональных данных? Любое ли лицо, юридическое или физическое, не говоря уже о госструктурах, является оператором, если ведет хранение и обработку ПДн?

– Прежде чем говорить об операторе персональных данных и его обязанностях, напомним, что включает в себя понятие «персональные данные». В соответствии со ст. 2 Федерального закона от 20 февраля 1995 г. № 24-ФЗ «Об информации, информатизации и защите информации» с изм. на 10 января 2003 г. (СЗ РФ. 1995. № 8. Ст. 609; 2003. № 2. Ст. 167) информация о работниках (персональные данные) – это сведения о фактах, событиях и обстоятельствах жизни работника, позволяющие идентифицировать его личность.

В статье 3 Федерального закона говорится:

Персональные данные – любая информация, относящаяся к определенному или определяемому на основании такой информации физическому лицу (субъекту персональных данных), в том числе его фамилия, имя, отчество, год, месяц, дата и место рождения, адрес, семейное, социальное, имущественное положение, образование, профессия, доходы, другая информация.

Оператор – государственный орган, муниципальный орган, юридическое или физическое лицо, организующие и (или) осуществляющие обработку персональных данных, а также определяющие цели и содержание обработки персональных данных.

Итак, исходя из буквы закона персональные данные – это любые сведения, по которым можно однозначно идентифицировать физическое лицо; при этом любая организация, имеющая (обрабатывающая, накапливающая и проч.) у себя в информационной системе персональные данные даже только своих сотрудников, уже является оператором. И согласно статье 13 государственные органы, муниципальные органы также являются операторами персональных данных и в пределах своих полномочий, установленных в соответствии с федеральными законами, создают государственные или муниципальные информационные системы персональных данных.

Все построились и стройными рядами пошли подавать уведомление об обработке персональных данных в уполномоченный орган, кроме исключений, предусмотренных законом (статья 22, пункт 2).

– Оператор должен принимать меры по защите персональных данных. Для этого у ФСТЭК разработаны методические материалы. Многие из них имеют гриф «ДСП» и распространяются только по государствен-

Досье

ВИКТОР МИНИН, советник председателя Совета МОО АЗИ, гендиректор компании «Миктера»

ЮРИЙ ШОЙДИН, директор по ИТ ГК «Интарсия», член СоДИТ, член правления Санкт-Петербургского клуба ИТ-директоров, независимый консультант в области управления изменениями в компании

Все построились и стройными рядами пошли подавать уведомление об обработке персональных данных в уполномоченный орган

ным органам власти. Что делать остальным операторам?

– Не вдаваясь в подробности всех шестнадцати руководящих документов, имеющих прямое отношение к обработке персональных данных, остановимся только на основных моментах.

В соответствии с Федеральным законом № 152-ФЗ, который уже вступил в силу 26 января 2007 года, оператор при обработке персональных данных обязан принимать необходимые организационные и технические меры для защиты персональных данных от неправомерного или случайного доступа к ним, уничтожения, изменения, блокирования, копирования, распространения, а также от иных неправомерных действий (статья 19).

При этом информационные системы персональных данных, созданные до дня вступления в силу настоящего Федерального закона, должны быть приведены в соответствие с его требованиями не позднее 1 января 2010 года (статья 25). И здесь уже можно говорить о тех уступках, на которые пошли наши законодатели.

Так, 20 ноября 2009 года в Государственной Думе состоялось рассмотрение и принятие, практически единогласно, в первом чтении законопроекта «О внесении изменений в Федеральный закон «О персональных данных» в части исключения требования об использовании криптографических средств защиты персональных данных и продления срока на один год, в течение которого ранее созданные информационные системы персональных данных подлежат приведению в соответствие с Федеральным законом. То есть дата 01.01.2010 переносится на 01.01.2011 и отменены требования об обязательном использовании шифровальных (криптографических) средств для обеспечения безопасности персональных данных при их обработке. Последнее изменение обусловлено тем, что защита персональных данных в зависимости от способа их обработки (например, неавтоматическая обработка) может осуществляться и без использования шифровальных (криптографических) средств.

Теперь о предъявляемых требованиях к обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных, которые устанавливает Правительство Российской Федерации (статья 19, пункт 2).

Контроль и надзор за выполнением требований осуществляются федеральным органом исполнительной власти в пределах их полномочий и без права ознакомления с персональными данными, обрабатываемыми в информационных системах персональных данных:

- Федеральный орган исполнительной власти, уполномоченный в области обеспечения безопасности, – ФСБ России. Традиционно курирует вопросы защиты информации с использованием средств шифрования (криптографии).
- Федеральный орган исполнительной власти, уполномоченный в области противодействия техническим разведкам и технической защиты информации, – ФСТЭК России. Осуществляет контроль защиты информации с использованием технических средств.

Согласно поручению Правительства РФ были разработаны следующие методические документы:

- «Методика определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных».
- «Базовая модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных».
- «Основные мероприятия по организации и техническому обеспечению безопасности персональных данных, обрабатываемых в информационных системах персональных данных».
- «Рекомендации по обеспечению безопасности персональных данных при обработке при их обработке в информационных системах персональных данных».
- «Типовые требования по организации и обеспечению функционирования шифровальных (криптографических) средств, предназначенных для защиты информации, не содержащей сведений, составляющих госу-

дарственную тайну, в случае их использования для обеспечения безопасности персональных данных при их обработке в информационных системах персональных данных».

> «Методические рекомендации по обеспечению с помощью криптосредств безопасности персональных данных при их обработке в информационных системах персональных данных с использованием средств автоматизации».

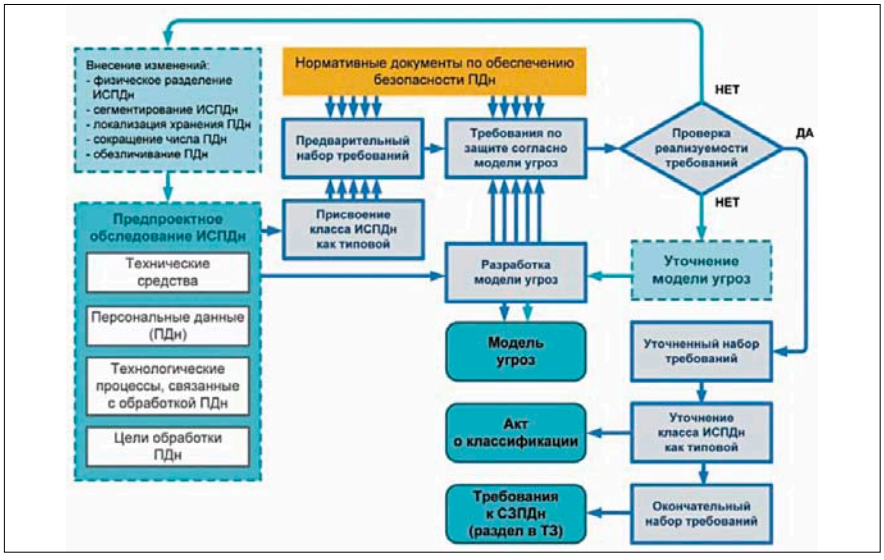
Первые четыре – это «четырехкнижье» было разработано ФСТЭК России и имеет гриф «ДСП». Последние два документа, или «двухкнижье», были разработаны ФСБ России, не имеет ограничительного грифа и находятся в свободном доступе. Указанные «методички» были разработаны на основе уже действующих документов, регулирующих вопросы обеспечения безопасности конфиденциальной информации, в большей своей части имеющих ограничительный гриф и распространяемых установленным порядком. Персональные данные относятся к категории конфиденциальной информации (документированной информации, доступ к которой ограничивается в соответствии с законодательством Российской Федерации) (ст. 2, 11 Федерального закона «Об информации, информатизации и защите информации»). Они указаны в Перечне сведений конфиденциального характера, утвержденном Указом Президента РФ от 6 марта 1997 г. № 188 «Об утверждении перечня сведений конфиденциального характера» (СЗ РФ. 1997. № 10. Ст. 1127), за исключением сведений, подлежащих распространению в средствах массовой информации в установленных федеральным законом случаях.

Абсолютно неверное утверждение о невозможности получения документов ограниченного распространения и доступности их только для ОГВ. Сейчас любая организация, независимо от форм собственности, на основании запроса во ФСТЭК России получит вышеуказанные документы. При этом выписки из данных документов уже доступны на сайте ФСТЭК России – http://www.fstec.ru/_razd/_ispo.htm.

В целях обеспечения контроля и надзора за выполнением требований Федерального закона «О персональных данных» назначен Уполномоченный орган по защите прав субъектов персональных данных – Федеральная служба по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзор Министерства связи и массовых коммуникаций). Он является основным исполнительным и надзорным органом по защите прав физических лиц, чьи персональные данные обрабатываются.

Представительства этих уважаемых организаций есть во всех крупных городах, и найти их координаты в Интернете или телефонном справочнике не составляет большой сложности.

Порядок действий для правильной классификации ИСПДн



Что делать? Прежде всего надо подать уведомление. Федеральный закон говорит (статья 22), что оператор до начала обработки персональных данных обязан уведомить уполномоченный орган по защите прав субъектов персональных данных о своем намерении осуществлять обработку персональных данных, за исключением случаев, предусмотренных частью 2 этой статьи. Второе: чтобы правильно произвести классификацию вашей ИСПДн, нужно четко понимать порядок действий (см. рисунок) и использовать методические документы регуляторов.

Что же касается распространенного мнения, что не надо спешить с уведомлением уполномоченного органа, потому что «как только мы подадим уведомление, нас сразу же начнут проверять», то это не так. На самом деле все с точностью до наоборот. Уполномоченный орган в первую очередь при составлении плана проверок будет обращать внимание на организации, не подавшие уведомление.

В дальнейшем, как указывает Роскомнадзор, проверки будут формироваться по принципу: 30% – плановые и 70% – внеплановые. Причем под внеплановые проверки в первую очередь попадут компании, у которых были зафиксированы утечки персональных данных, на которые были поданы жалобы и иски от граждан, и те организации, которым ранее уже были выписаны предписания.

На сайте Роскомнадзора http://www.rsoc.ru есть раздел, где в открытом доступе можно ознакомиться со списком компаний, попавших в план ближайших проверок. Не поленитесь заглянуть и найти свою компанию в этом списке.

– Системный администратор не несет ответственности за деятельность нанимателя. Однако зачастую в служебных регламентах для системного администратора прописана функция «защита информации». А если и не так, то все равно защитой информации придется заниматься ему.

– Не будем детально обсуждать весь спектр возможных последствий нарушения ФЗ № 152 (это дело юристов), остановимся только на двух статьях.

На основании статьи 23 Федерального закона Уполномоченный орган по защите прав субъектов персональных данных имеет право:

- направлять заявление в орган, осуществляющий лицензирование деятельности оператора, для рассмотрения вопроса о принятии мер по приостановлению действия или аннулированию соответствующей лицензии в установленном законодательством Российской Федерации порядке, если условием лицензии на осуществление такой деятельности является запрет на передачу персональных данных третьим лицам без согласия в письменной форме субъекта персональных данных;
- направлять в органы прокуратуры, другие правоохранительные органы материалы для решения вопроса о возбуждении уголовных дел по признакам преступлений, связанных с нарушением прав субъектов персональных данных, в соответствии с подведомственностью.

Статья 24 гласит, что лица, виновные в нарушении требований настоящего Федерального закона, несут гражданскую, уголовную, административную, дисциплинарную и иную предусмотренную законодательством Российской Федерации ответственность.

Несмотря на обилие статей гражданского, уголовного и административного кодексов, по которым оператор может быть привлечен к ответственности, выделим главное. Уполномоченный орган может ходатайствовать о приостановке действия лицензии на основной вид деятельности оператора на срок до 90 дней. Дальше решать вам, т.е. вашему руководителю компании, так как ответственность согласно выше перечисленному несет лично он.

Исходя из имеющейся статистики, далеко не каждая компания имеет в своем штате специалиста по информационной безопасности, в чьи непосредственные обязанности должна входить подготовка и аттестация ИС компании. Чаще всего в компаниях существуют подразделения ИТ и ИБ. Кто же должен заниматься в компании подготовкой и аттестацией ИС по требованиям Федерального закона?

Конечно, в каждой компании данный вопрос будет решаться индивидуально, но общую рекомендацию, наверное, можно дать. Описание ИС, а точнее, ее части, в которой обрабатываются персональные данные (ИСПДн), конечно, должен делать специалист ИТ-подразделения, а вот общую документацию и меры защиты должен разработать и контролировать сотрудник безопасности. Четкое взаимодействие этих подразделений позволит компании минимизировать свои расходы при проведении работ на соответствие требованиям и собственно проведение самой аттестации ИС.

В любом случае обязательным (на основании ФЗ) является назначение ответственного за обеспечение защиты персональных данных, и неважно, будет это специалист из ИТ-подразделения или из безопасности, главное, чтобы он был в организации. Правильное распределение функций и ответственности между подразделениями обязательно принесет свои плоды. Конечно, всегда остается возможность привлечения стороннего консультанта, который сделает эту работу за вас.

А теперь представьте, что вас назначили ответственным в компании за ИБ ПДн. Готовы ли вы отвечать по действующему законодательству за работу консультанта? Если нет, тогда давайте посмотрим, что можно сделать самостоятельно.

– Как избежать больших расходов при переходе на нормы закона о персональных данных? На чем можно экономить, а на чем нет?

– Для сокращения затрат организации на подготовку ИСПДн к аттестации можно часть работ из приведенного списка выполнить самостоятельно. Объем работ, которые вы можете взять на себя, зависит только от квалификации ваших сотрудников и желания разобраться с руководящими документами. Условно работу можно разделить на два блока.

Выполнить самостоятельно:

- анализ процессов компании на предмет выявления ПДн;
- аудит ИС на предмет оценки защищенности ПДн;
- подготовка необходимых внутренних документов;
- разработка плана технической защиты;
- подготовка и подача уведомления в уполномоченный орган.

Передать лицензиату ФСТЭК следующие мероприятия:

- определение категории информации;
- определение класса системы;
- разработка модели актуальных угроз;
- разработка мер по компенсации угроз.

Если вы решили передать консультантам весь комплекс работ по подготовке вашей ИСПДн к аттестации, то условно проект разделится на два этапа. Один из главных методов снижения затрат – это оптимизация бизнес-процессов и обработки персональных данных в ИС. И второй метод – четкое определение границ ИСПДн, при этом в компании может быть несколько ИСПДн различных классов в рамках единой ИС.

Общий порядок действий при подготовке ИСПДн к аттестации

Конечно, данный порядок достаточно условен и в каждой конкретной ситуации может быть уникален, но не грех еще раз проверить готовность каждого блока:

- Оценка законности обработки ПДн и наличие согласия субъектов на обработку.
- Контроль и корректировка договорных отношений с субъектами.
- Формирование перечня ПДн и проведение категорирования.
- Определение сроков и условий прекращения обработки ПДн.
- Инвентаризация ИС, обрабатывающих ПДн.
- Присвоение класса ИСПДн.
- Разграничение доступа пользователей к ПДн в ИСПДн.
- Формирование документов, регламентирующих работу с ПДн.
- Формирование модели актуальных угроз.
- Разработка мероприятий по компенсации угроз.
- Направление уведомления об обработке ПД в уполномоченный орган.
- Приведение системы защиты ПДн в соответствие требованиям регуляторов.
- При необходимости, определенной методическими документами ФСТЭК России и ФСБ России, получить необходимые лицензии.
- Аттестация ИСПДн.
- Эксплуатация ИС – мониторинг, выявление и реагирование на инциденты ИБ.
- Контроль за соблюдением условий использования средств защиты информации, предусмотренных документацией.

– Реализовать защиту ПДн без затруднения бизнес-процессов, а в некоторых случаях нарушения или остановки бизнес-процессов, практически невозможно (вернее, возможно, но для этого требуется огромное количество денег).

– Да, безусловно, одно из основных требований руководства к ИТ- и ИБ-службам – это обеспечение непрерывности бизнеса. В различных компаниях уровень автоматизации бизнес-процессов может колебаться от 10 до 99%, и с повышением автоматизации требования к ИТ-технологиям, используемым в ИС, ужесточаются. В некоторых информационных системах бизнес-процессы действительно могут быть усложнены, хотя вариантов избежать этого достаточно много. Чтобы правильно и безболезненно для организации подчистить все бизнес-процессы избыточного и, как правило, неоправданного использования персональных данных, желательно обратиться к профессионалам.

– Какие ПК попадают в классификацию ИСПДн? Нужно ли классифицировать все ПК, имеющие доступ к серверу (в которых база не хранится, а только открывается – для обработки, просмотра и т.д.)?

– Классифицировать надо ИСПДн, а не ПК. Другое дело, что в ИСПДн должны входить только ПК и серверы, на которых производится ввод и обработка ПДн. В отдельных случаях ИСПДн может состоять из одного ПК, это можно использовать при проведении процедуры оптимизации обработки персональных данных в ИС.

– Компании, обещающие консультации и проведение подготовки к аттестации: кто они и кому стоит доверять?

– Наверное, по этому поводу сказать что-то новое в принципе невозможно, к выбору стороннего исполнителя надо подходить точно так же, как и к выбору подрядчика для любых других работ. Желательно, чтобы компания, которой вы отдадите преимущество, имела необходимый опыт проведения таких работ и являлась лицензиатом ФСТЭК России и ФСБ России, что позволит ей заниматься вашим проектом.

Какие лицензии должен иметь ваш консультант? Приведенный ниже список, несомненно, избыточен, но в данном случае мы постарались указать лицензии для максимального комплекса работ консультанта. Познакомиться со списком лицензиатов ФСТЭК России и их лицензиями можно на соответствующем информационном ресурсе http://www.fstec.ru.

1. Лицензии Федеральной службы безопасности Российской Федерации на осуществление:
   - разработки, производства шифровальных (криптографических) средств, защищенных с использованием шифровальных (криптографических) средств информационных и телекоммуникационных систем;
   - технического обслуживания шифровальных (криптографических) средств;
   - распространения шифровальных (криптографических) средств.
2. Лицензия Федеральной службы безопасности Российской Федерации на осуществление разработки и (или) производства средств защиты конфиденциальной информации.
3. Лицензия Федеральной службы по техническому и экспортному контролю на проведение работ, связанных с созданием средств защиты информации:
   - разработка, производство, реализация, установка, монтаж, наладка, испытания, ремонт, сервисное обслуживание.
4. Лицензия Федеральной службы по техническому и экспортному контролю на деятельность по технической защите конфиденциальной информации:
   - осуществление мероприятий и оказание услуг по технической защите конфиденциальной информации.
8. Лицензия Федеральной службы по техническому и экспортному контролю на деятельность по разработке и (или) производству средств защиты конфиденциальной информации:
   - осуществление разработки и производства средств защиты конфиденциальной информации.

Наличие у компании лицензий ФСТЭК России и ФСБ России на право деятельности и выполнения работ в области защиты государственной тайны только еще больше подтверждает уровень квалификации и компетенции в части защиты информации.

Общая рекомендация – это, безусловно, квалификация и компетенция специалистов, работающих в компании, которую вы приглашаете. И еще: при самостоятельном подборе средств защиты информации внимательно читайте содержимое сертификатов на эти средства.

Что в итоге?

Уважаемые коллеги, данная статья не претендует на решение конкретных проблем, но надеемся, дает понимание, что и как можно успеть сделать до наступления дедлайна. Несмотря на общую неразбериху и непонимание того, что вы должны сделать в первую очередь, предлагаем подумать и не торопиться пугать генерального директора космическими цифрами коммерческих предложений консультантов. Во всяком случае, это не ускорит принятия положительного решения в части выполнения требований законодательства.

Как члены консультативного совета при уполномоченном органе по защите прав субъектов персональных данных (Роскомнадзор) можем с уверенностью сказать, что у регуляторов нет цели устраивать массовые репрессии. Конечно, исполнение закона обязательно, но текущая ситуация показывает, что регуляторы при проведении проверок всеми силами стремятся помочь организациям, а не наказать их. На наш взгляд – это главное.

И последнее. При появлении конкретного вопроса его можно задать напрямую любому из трех регуляторов, для этого есть все возможности: от личной встречи до обсуждения через указанные сайты. При этом нужно понимать, что количество операторов ПДн намного превышает численность подразделений регуляторов, скорость ответов не всегда будет приемлема для вашей ситуации. Чтобы хоть как-то помочь ИТ-директорам в части пояснений нормативных актов и методических консультаций, Комитет по ИБ Российского союза ИТ-директоров выпустил для членов клубов ИТ-директоров «Рекомендации по защите персональных данных» и готов отвечать на ваши вопросы на сайте СоДИТ (http://www.rucio.ru) или по адресу itsec@rucio.ru.

От редакции: «СА» продолжит обсуждение всех тонкостей применения Закона о персональных данных в 2010 году. EOF

Множественные уязвимости в Mozilla Firefox 3.5

Программа: Mozilla Firefox версии до 3.5.4.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: 1. Уязвимость существует из-за ошибки при формировании истории для содержимого веб-форм и строки состояния. Удаленный пользователь может с помощью специально сформированного веб-сайта синтезировать некоторые события с помощью пользователя (например, нажатие мыши и клавиш) и заставить браузер автоматически заполнить поля форм данными, содержащимися в истории браузера, и затем прочесть эти данные.

2. Уязвимость существует из-за предсказуемости имен временных файлов в Download Manager. Злоумышленник, который владеет информацией о ранее загруженных файлах пользователя и открытых с помощью Download Manager, может записать файл во временную директорию на системе для загрузки файлов и обманом заставить пользователя запустить ранее загруженное приложение через Download Manager. Для успешной эксплуатации уязвимости требуется локальный доступ к системе.

3. Уязвимость существует из-за ошибки при рекурсивном создании JavaScript web-workers. Удаленный пользователь может спровоцировать использование освобожденной памяти и аварийно завершить работу браузера или выполнить произвольный код на целевой системе.

4. Уязвимость существует из-за ошибки при обработке регулярных выражений, используемых в файлах Proxy Auto-configuration (PAC). Удаленный пользователь может с помощью специально сформированного PAC файла аварийно завершить работу приложения и выполнить произвольный код на целевой системе.

5. Уязвимость существует из-за ошибки в цветовых GIF-картах в обработчике GIF-изображений. Удаленный пользователь может с помощью специально сформированного GIF изображения вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе.

6. Уязвимость существует из-за ошибки в методе XPCVariant::VariantDataToJS() в утилите XPCOM. Удаленный пользователь может с помощью специально сформированного Web сайта выполнить произвольный Javascript сценарий с привилегиями chrome.

7. Уязвимость существует из-за ошибки индексации массива при выделении пространства для чисел с плавающей точкой. Удаленный пользователь может вызвать повреждение памяти при обработке определенного числа с плавающей точкой и выполнить произвольный код на целевой системе.

8. Уязвимость существует из-за ошибки в реализации Javascript функции document.getSelection(). Удаленный пользователь может прочитать выделенный на странице текст с другого домена.

9. Уязвимость существует из-за ошибки при обработке имен загружаемых файлов, содержащих в имени RTL (right-to-left) символы. Удаленный пользователь может подменить название и расширение отображаемого файла в окне загрузки и потенциально скомпрометировать целевую систему.

10. Уязвимость существует из-за ошибки во встроенной библиотеке liboggz. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

11. Уязвимость существует из-за множественных ошибок во встроенной библиотеке libvorbis. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

12. Уязвимость существует из-за ошибки во встроенной библиотеке liboggplay. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

13. Уязвимость существует из-за множественных ошибок в browser engine. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

14. Уязвимость существует из-за неизвестной ошибки в browser engine. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

15. Уязвимость существует из-за множественных ошибок в JavaScript engine. Удаленный пользователь может вызвать отказ в обслуживании и скомпрометировать целевую систему.

URL производителя: www.mozilla.com/firefox.
Решение: Установите последнюю версию 3.5.4 с сайта производителя.

Множественные уязвимости в Opera

Программа: Opera версии до 10.01.
Опасность: Высокая.
Наличие эксплоита: Нет.
Описание: 1. Уязвимость существует из-за ошибки при обработке доменных имен. Удаленный пользователь может вызвать повреждение памяти и выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за ошибки при выполнении сценариев на странице подписки на новостные ленты. Удаленный пользователь может получить доступ к данным новостной ленты или автоматически подписать на новостную ленту пользователя.

3. Уязвимость существует из-за ошибки при обработке веб-шрифтов в Windows. Удаленный пользователь может изменить шрифт адресного поля и отобразить произвольное доменное имя в качестве адреса.

URL производителя: www.opera.com.
Решение: Установите последнюю версию 10.01 с сайта производителя.

Составил Александр Антипов

Визитка

АЛЕКСЕЙ БАРАБАНОВ, системный администратор, аутсорсер и консультант по вопросам применения информационных технологий. Хобби – программирование

Вычислительная модель

Введение в системное программирование

Как установить точно, чем занимается системный администратор? Не определившись с этим, невозможно автоматизировать работу сисадмина

Часть 2

Итак, напомню, что в конце первой части [1] был определен список из семи типовых задач, решаемых системными администраторами. Поскольку в данной части будет построена общая вычислительная модель, характерная для всех видов операций системного администрирования, то в ходе рассуждений придется найти то общее, что определяет схожесть всех перечисленных в конце первой части задач системного администрирования. Нет лучшего способа сделать это, чем пойти в рассуждениях сразу с другой стороны и при этом совсем не учитывать возможное разбиение на функциональные подзадачи, а потом проверить, что получилось, сравнив, как это «ляжет» на список из первой части.

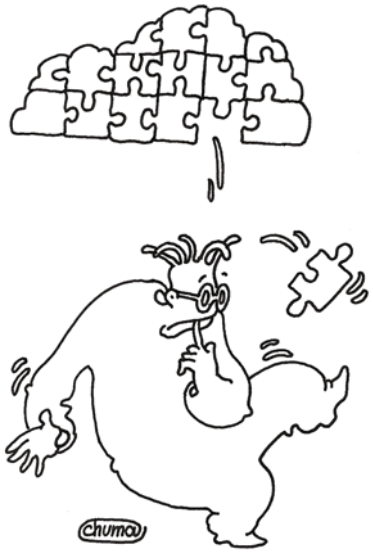
Программы – алгоритмы = данные

Снова напомню, что в первой части была введена такая характеристика информационного объекта, как «сложность». Ключевым свойством этого параметра является то, что штатное и даже нештатное прерывание работы информационных систем не меняет уровня их сложности! Получается, что сложность объекта сохраняется и в состоянии офлайн (off-line). Та самая сложность, что формируется в результате труда системного администратора. Другими словами, следует принять офлайн за точку сингулярности и согласиться, что дальнейшее поведение системы зависит только от её свойств в состоянии офлайн, что и называется настройкой информационной системы. Будем считать, что рассматриваемые информационные системы состоят из информационных объектов или компонентов, имеющих архитектуру фон Неймана [2]. Иначе говоря, вся информация хранится в памяти таких систем. А какая память сохраняет свое значение в состоянии офлайн? Очевидно, все, что сделает системный администратор, так или иначе должно быть размещено и запомнено в энергонезависимой памяти. В общем, ничего удивительного, так как и программисты, и все остальные пользователи компьютеров в результате своего труда, как правило, просто меняют состояние секторов жесткого диска на сервере или своем персональном компьютере. Но теперь выходит, что весь процесс администрирования можно разбить на некоторую последовательность операций, которые сводятся к изменению состояния памяти. То есть ненастроенный компьютер отличается от того, который настроен, наполнением устройств памяти в состоянии офлайн. Количество этих отличий, конечно, и составляет именно ту работу, что нужно выполнить системному администратору, чтобы, как это было указано в первой части, подготовить компьютер к эксплуатации. Все вышесказанное также справедливо и в отношении любых информационных систем. Тут можно возразить, ведь согласно подобной логике программный расчет интеграла сводится к тривиальному изменению состояния ячейки, в которой хранится результат. Вся разница в том, что в случае задания на административную настройку указывается именно требуемое конечное состояние настроенной системы, и ничего рассчитывать, как правило, не нужно. То есть сисадмин всегда заранее знает, какие параметры готового информационного объекта должны получиться в итоге. Вот теперь настало время сравнить, как сочетается все вышесказанное со списком семи задач из первой части.

Для наглядности составим таблицу (см. таблицу 1), где в первой колонке перечислим задачи из [1], во второй укажем место офлайнового хранения данных, используемых или создаваемых в ходе выполнения каждой задачи, а в последней колонке представим формат задания для выполнения каждой задачи.

Если внимательно рассмотреть полученную таблицу, то становится видно, что последняя колонка, «Формат задания», фактически описывает или в точности совпадает с предпоследней, «Формой хранения». Это именно тот результат, который и ожидался. Чтобы окончательно убедиться в этом, предлагаю выполнить в предпочитаемой поисковой системе Интернета запрос «руководство по настройке сервера» и убедиться, что внутри полученных по ссылкам текстов не содержится формул сложней тривиальных арифметических расчетов объема памяти или простых граничных прикидок «если меньше, то» и «если больше, то».

Таким образом, принимаем за верное предположение о том, что все операции администрирования можно свести



Создание программ автоматического администрирования под силу сисадмину практически любого уровня

к установкам определенных, заранее заданных значений элементов и структур памяти. Сисадмин получает в техническом задании (или сам создает его) как раз те значения, которые и будут далее использованы в процессе работы, когда эти значения почти в неизменном виде окажутся положенными в основу разметки файловой системы, определят список установленных пакетов, укажут перечень учетных записей, опишут настройки конфигурационных файлов и многое другое. На данном этапе рассуждений весь процесс администрирования можно представить как последовательность неких абстрактных операций, состоящих из «черного ящика» алгоритмического действия, на вход которого поступает точное описание требуемого результата в терминах выбранной платформы программирования, а на выходе получается, очевидно, ожидаемый результат (см. рис. 1). И хотя такой алгоритм, по сути, программирует тавтологию, дальнейшее раскрытие его свойств, возможно, заставит некоторых читателей пересмотреть собственные ранее написанные административные скрипты.

Симметричность и рефлексивность

Теперь договоримся о терминологии. Назовем связанную и законченную последовательность упорядоченных административных операций решением, а сами операции, из которых состоят решения, станем называть процедурами (см. рис. 1). Пример сложного решения: настройка сервера с функцией шлюза локальной сети в Интернет. Такое решение включает множество элементарных процедур. А вот пример очень простого решения: восстановление поврежденного файла зон DNS. Это решение может состоять буквально из одной процедуры–восстановления из резервной копии. Казалось бы, полный произвол, и вообще непонятно, к чему такие терминологические новации.

Чтобы определиться с этим, рассмотрим две типичные административные задачи: первая – установка ECM Alfresco [3] и вторая – установка jabber-сервера Openfire [4]. Я даю ссылки на статьи в журнале «Системный администратор», но пытливым умам предлагаю расширить репрезентативную выборку и воспользоваться поиском в Сети для получения альтернативных установочных рекомендаций. Оба эти продукта требуют в своей работе СУБД. Выберем варианты установки с использованием СУБД MySQL. Допустим, что требуется установить и Alfresco, и Openfire. А вот теперь представим, как должны измениться рекомендации по установке [3] и [4], или найденные самостоятельно в Сети, если один из продуктов ставится после второго. В этом случае как минимум должен учитываться факт существования уже настроенного сервера MySQL. И в корректных инструкциях по установке так и пишется, что-то вроде «установить MySQL или использовать уже установленный». В статье [3]

Таблица 1. Формы хранения и форматы задания

	Типовые задачи	Форма хранения	Формат задания
1	Модификация файловой системы	Файлы и папки	Имена файлов и папок
2	Установка и удаление пакетов	Пакеты и база пакетного менеджера	Список URL пакетов или сами пакеты
3	Создание и удаление пользовательских учетных записей	База учетных записей	Список учетных записей с атрибутами
4	Создание и модификация конфигурационных файлов	Конфигурационные файлы и базы	Настройки в формате конфигурационных файлов
5	Последовательное выполнение настроек согласно спецификации	Спецификация настроек	Установочные параметры
6	Откат к исходному состоянию настроек	Резервные копии	Имена файлов и папок для резервирования
7	Удаленное или автоматические выполнение всех перечисленных действий на нелокальной системе	База паролей и ключей доступа	Пароли, ключи, адреса и порты

о том, откуда взялся установленный MySQL, не сказано ничего, а в [4] MySQL устанавливается заново, а потом без какой-нибудь настройки сразу к нему подключается Openfire. Для журнальных статей, цель которых дать общие представления о продуктах, подобная поверхностность допустима. Но создание программы автоматической установки по таким материалам невозможно. Строгий алгоритм установки должен учитывать все возможные коллизии и иные проблемы, которые могут возникнуть в процессе работы. Кроме очевидной зависимости от свойств исходного объекта (например, платформы, характеристик оборудования), законченные решения автоматизированного системного администрирования должны учитывать также и влияние на условия их работы других, ранее завершившихся, алгоритмических последовательностей. Что значит учитывать? В данном случае решения системного администрирования должны быть независимыми один от другого. Тогда порядок их применения станет не важен. Независимость выполнимости от порядка применения в математической логике называется симметричностью. Таким образом, все решения должны обладать свойством симметричности.

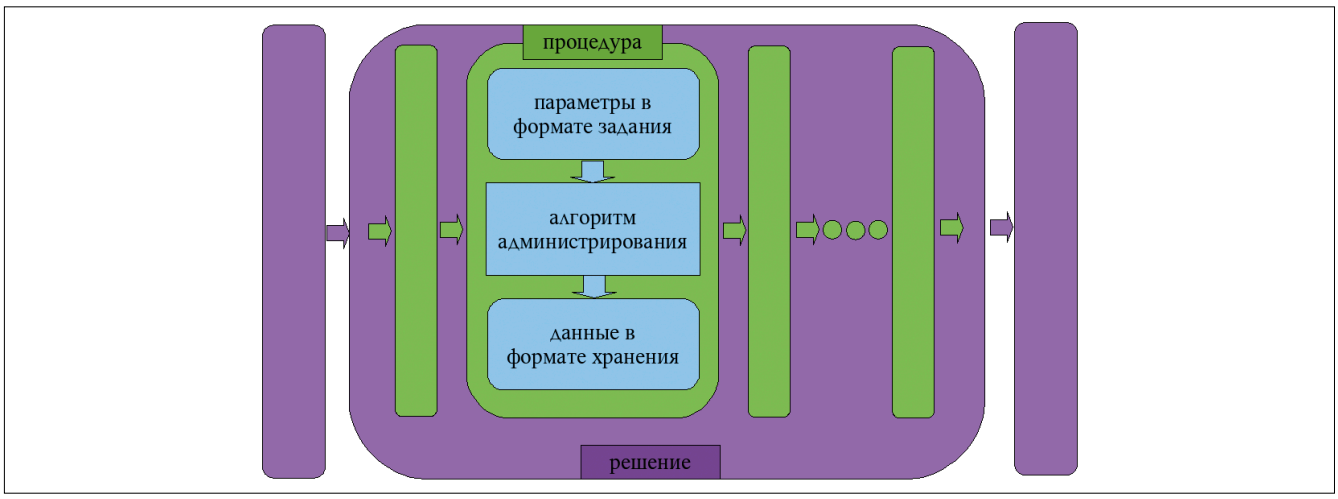
Нарушение этого требования в ходе проектирования алгоритма системного администрирования чревато трудно детектируемыми ошибками. В чем это может выражаться? Типичный случай: вы решили воспользоваться рекомендациями по установке, найденными в Сети, и на очередном шаге был получен результат, отличный от авторских. Это значит, при написании статьи автор не учел какой-то важный фактор. Печально, но не смертельно. Можно воспользоваться документацией, поиском в Интернете, списаться с автором и так или иначе преодолеть проблему. Совсем иное дело, если какой-то важный фактор не будет учтен в процессе проектирования алгоритма автоматической установки или настройки. Это уже категорически не допустимо, так как кроме фатального завершения программы настройки сама информационная система установится в неработоспособное или вообще в неопределенное состояние.

Теперь рассмотрим такое решение, как восстановление из резервной копии. Что произойдет, если это решение применить дважды? Сравните: что произойдет, если бэкап дважды сохранить? Очевидно, и в первом, и во втором случае можно считать, что результат будет идентичен или почти идентичен (напоминаю, речь идет о системных настройках). А если иначе, используется некий алгоритм автоматической установки. И вот он указывает на фатальную ошибку, требующую вмешательства оператора. Исправили. Дальше что делать? Исправлять теперь программу установки так, чтобы она стартовала с точки прерывания? Здесь ответ тоже очевиден – надо обеспечить корректную повторную исполнимость. И хотя на практике в текстовой документации нечасто можно встретить рекомендации, учитывающие возможность многократного исполнения (про типичные установочные скрипты молчу вовсе), для автоматизированных систем надо выдвинуть такое требование к алгоритмическому воплощению решений, чтобы получить в конце концов программы, пригодные к эффективному практическому применению. Свойство корректной повторной исполнимости называется рефлексивностью. Как было указано выше, в решениях резервного копирования это свойство достигается by design. Теперь примем, что рефлексивностью должны обладать все решения автоматического системного администрирования.

Есть интересный пример, демонстрирующий именно такое свойство решений автоматизированной установки и настройки, основанных на Cfengine [4]. В видеоролике показано, что фактически автоматизированное решение может заменить резервное копирование, так как оно или «устанавливает», или «восстанавливает» заданные спецификацией информационной системы параметры.

Требование симметричности и рефлексивности не абстрактно. Оно накладывает определенные ограничения на методы реализации. Например, утилита rpm обладает только свойством рефлексивности, но в силу того, что её результат всецело зависит от порядка вызова, или, говоря иначе, передавать ей названия или адреса пакетов следует в определенном порядке, то на rpm без особых ухищрений, скорее всего, не получится построить практические установочные скрипты, обладающие свойством симметричности. А вот более мощные пакетные менеджеры, например yum, за счет использования индексов репозиториев могут «на лету» разрешать пакетные зависимости, а значит,

Рисунок 1. Разбиение процесса системного администрирования на субоперации



их можно применять к аргументам, следующим в произвольном порядке – недостающее будет добавлено, лишнее проигнорировано. Следовательно, в практическом программировании использование rpm придется свести к минимуму, заменяя его на более развитые пакетные менеджеры типа yum, smart, apt-get и прочие.

Когда свойства решений определены, можно объяснить, в чем смысл разбиения одного большого решения на множество маленьких процедур. Каждая процедура должна алгоритмически строиться так, чтобы соблюдать требование симметричности и рефлексивности для решения в целом. Например, если резервное копирование или работу развитого пакетного менеджера можно сразу считать корректными с этих точек зрения, в отношении других задач системного администрирования соблюдение симметричности и рефлексивности должно достигаться за счет дополнительных алгоритмических инкапсуляций, скрытых в процедурах.

Программы – данные = алгоритмы

Приступим к проектированию алгоритма администрирования. Абстрагируемся от используемых данных, будем считать, что все они представлены в некотором «формате хранения», а параметры задаются в некотором «формате задания». Воспользуемся общепринятой нотацией – блок-схемами. Только в этих блок-схемах точка входа будет одна, а вот число выходов будет определяться числом состояний настраиваемой системы после выполнения такого алгоритма. И нумеровать элементы блок-схем будем по мере их упоминания в процессе разработки алгоритма, а не в порядке исполнения.

Положим в основу каждой процедуры некую алгоритмическую операцию, принимающую на входе согласно таблице 1 параметры в формате задания и изменяющую нужным образом соответствующую форму хранения. Пусть эта алгоритмическая операция пока останется в виде абстрактного «черного ящика» – делается нечто и получается то, что нужно. Изобразим это на блок-схеме 1 (см. рис. 2) в виде элемента 1. Но в предыдущем разделе было выдвинуто требование рефлексивности. Чтобы соблюсти это, в блок-схему добавим проверку, нужна ли эта операция вообще, может, требуемая настройка уже была произведена ранее – элемент 2. Таким образом построенный алгоритм можно применять к информационному объекту многократно без какого-либо побочного эффекта. Затем, после установки, произведем контрольную проверку – элемент 3. На самом деле ошибкой может завершиться даже блок 1, но если далее все равно будет проверяться, успешной ли была установка, можно игнорировать коды возврата от блока 1. И тем самым блок-схема становится независимой от реального кодового наполнение установочного элемента 1. Еще одним важным элементом в поступательном прохождении рассматриваемой блок-схемы будет 4 – сохранение исходного состояния модифицируемых данных. Этот элемент нужен для реализации ветки аварийного выхода из алгоритма, которая должна производиться через блок восстановления исходного состояния 5. Дополнительно на блок-схеме 1 изображены источники данных и направления их движения (пунктирные стрелки).

В итоге блок-схема процедуры будет иметь один вход и два выхода – успешный и аварийный, что соответствует семантике блока выбора, который обычно изображается в виде ромба, и именно так, в виде ромба (элемент 1, блок-схемы 2 (см. рис. 3), все процедуры будут инкапсулированы в блок-схему решения. Как уже было сказано, процедур может быть много или она может быть одна. Если их будет много, то все они выполняются строго упорядоченно, и каждая последующая должна вызываться после успешного срабатывания предыдущей, а первый же сбой, или аварийный выход, должен прерывать всю последовательность. Здесь снова вспомним о рефлексивности: процедуры могут как срабатывать, так и пропускать действие, если установка была ранее уже произведена или вообще являлась исходным состоянием информационного объекта. Так появляется необходимость регистрации факта срабатывания – элемент 6 в блок-схеме процедуры. Но и это не все. Аварийное завершение одной из процедур должно приводить к откату всех ранее произведенных изменений. Значит, элемент 5 должен быть вынесен из алгоритма процедуры на уровень всего решения – элемент 2 на блок-схеме 2. К сожалению, развернутое изображение этого элемента займет много места, потому ограничимся его описанием: этот элемент алгоритма должен будет восстанавливать не все упомянутые в алгоритме дан-

Рисунок 2. Блок-схема 1. Алгоритм процедуры

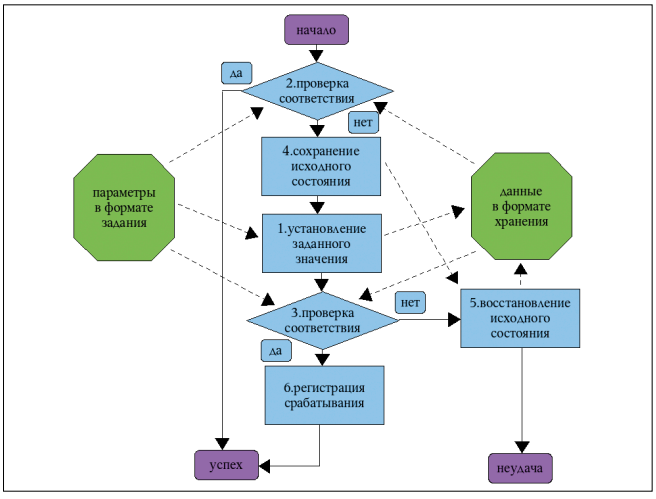
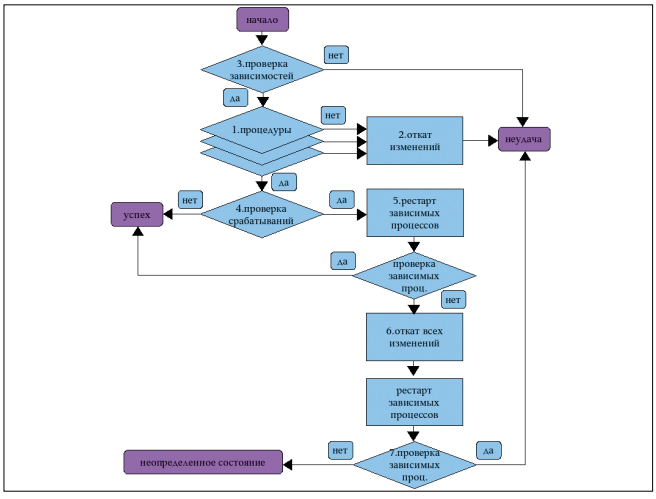


Рисунок 3. Блок-схема 2. Алгоритм решения



ные, а лишь те, что описаны в процедуре, завершившейся неудачей, и всех ранее пройденных. Здесь возможна проблема инструментального порядка. Доступны две стратегии. Первая – восстанавливать только сохраненные ранее данные, вторая – можно вынести в каждой процедуре элемент 4 выше элемента 2 и тем самым обеспечить формирование резервной копии для отката всех определяющих решение данных. Этот выбор нужно сделать на этапе кодирования.

Дальнейшее обсуждение будет касаться только алгоритма решения (блок-схема 2). Как было сказано выше, надо обеспечить симметричность и рефлексивность алгоритма. Первое будет обеспечиваться элементом 3, в котором проверяются все зависимости создаваемого алгоритма. А за второе станет отвечать последующая проверка срабатывания (элемент 4): если решение запущено повторно, то оно должно завершиться успешно на уже настроенной системе. Теперь снова обратимся к свойству симметричности. В списке зависимостей для некоторого решения могут быть не только определенные установочные параметры, но и процессы, настройки которых предполагается изменять в ходе выполнения алгоритма решения. Кроме того, целью решения может быть (и очень часто так и есть) настройка некоторого сервисного процесса или даже группы процессов. Так появляется элемент 5 и вся последующая цепь в блок-схеме 2. В этой части алгоритм на первый взгляд достаточно очевиден: элемент 6 должен обеспечивать восстановление всех исходных данных, а проверка 7 в случае неудачи будет детектировать неопределенное состояние системы. Главным является то, что здесь для проверки настраиваемые части информационной системы (или даже вся система в случае перезагрузки) проводятся через точку сингулярности – рестартуют!

Что в итоге?

На этом обобщенный алгоритм автоматизации системного администрирования практически построен. На следующем этапе (в следующей части статьи) будет рассмотрено кодирование этого алгоритма. Безусловно, в него еще будут внесены обоснованные изменения, поскольку более детальное рассмотрение обязательно исправит что-то в той модели, что была построена в этой части. Но уже сейчас можно сделать небольшие выводы. И вот первый – вычислительная сложность алгоритмов и применяемых методов (не путать с числом шагов или размером алгоритмов) системного администрирования очень невысока! Создание программ автоматического администрирования под силу сисадмину практически любого уровня.

Вывод второй. К процедурам можно относиться, как к предикатам на множестве значений состояний информационной системы, выраженным некоторой функцией с набором параметров в формате представления. Такие предикаты-процедуры производят отображение состояния системы на множество {успех, неудача}, или, как это принято записывать в математике, в инкрементном порядке – {0, 1}. Этот вывод напрямую следует из блок-схемы 1, которая изображает типовой алгоритм процедуры, завершающийся двумя выходами. А вот блок-схема 2, демонстрирующая алгоритм решения, может завершаться уже тремя путями – {успех, неудача, неопределенное состояние}. Иначе говоря, если процедуры можно представлять предикатами в классической двоичной логике, то решениям соответствуют предикаты в логике троичной. И здесь уже можно утверждать, что первоначальное и весьма условное разбиение структуры алгоритма администрирования на решения и процедуры имеет под собой серьезные основания, определяющиеся природой информационных систем, и потому должно сохраниться так или иначе и на этапе кодирования.

Третий вывод такой: можно предположить, что полученный алгоритм универсален в том, что позволяет описывать решение всех возможных задач администрирования информационных систем, построенных на компонентах архитектуры фон Неймана [2]. Описывать, да! Но можно ли при этом утверждать, что все такие описания будут верными? Не совсем, или не всегда. Однозначное предположение о верности таких алгоритмов можно сделать лишь на множестве состояний информационной системы и наборе входных параметров, которые приводят к получению ответов {успех, неудача}. Если же применение алгоритма переводит систему в неопределенное состояние, требующее каких-то дополнительных интерактивных действий оператора, что в контексте математической логики эквивалентно божественному вмешательству, то однозначно утверждать правильность алгоритма невозможно. Причина может содержаться как в природе системы, так и в неполноте учтенных факторов, что соответствует неверному или неполному набору параметров. Технически такая ошибка, как неполнота учтенных факторов, неизбежно приводит к нарушению требования симметричности алгоритмов администрирования. Таким образом, снова подтверждается важность свойства симметричности для правильных алгоритмов.

Уже ясно, что должно быть внутри программных систем, реализующих алгоритмы системного администрирования. И перед тем как конкретная реализация будет описана в следующей части, предлагаю самостоятельно проверить, какие вышеперечисленные элементы алгоритмического решения присутствуют в реальных, встречаемых на практике скриптах, а каких нет, и к чему это приводит. Тому, кто еще не имеет скриптов собственной разработки, рекомендую рассмотреть программу Геннадия Калашникова [6] или поискать другие скрипты комплексной установки систем в Сети. Полученные выводы сверим в следующей части. EOF

1. Барабанов А. Введение в системное программирование. Часть 1. Постановка задачи. //Системный администратор, №4, 2008 г. – С. 8-13.
2. Архитектура фон Неймана. Статья Википедии – http://ru.wikipedia.org/wiki/Архитектура_фон_Неймана.
3. Яремчук С. Обзор Open Source ECM-системы Alfresco. //Системный администратор, №3, 2009 г. – С. 37-43.
4. Яремчук С. Строим Jabber-сервер с OpenFire. //Системный администратор, №5, 2007 г. – С. 44-48.
5. Демонстрационный ролик восстановления резолвера DNS с помощью Cfengine – http://cfengine.com/pages/demos?view=Cfengine_DNS_Resolver.
6. Новость с ссылкой на пакет автоматической установки OpenXchange, Samba PDC и проч., разработанный Геннадием Калашниковым (04.01.2006) – http://www.opennet.ru/openforum/vsluhforumID3/13191.html.
7. System Programming Introduction. Part 2. август 2007 – октябрь 2009.

Визитка

ИГОРЬ ШТОМПЕЛЬ, инженер, системный администратор. Сфера профессиональных интересов – GNU/Linux, функциональное программирование

Вызов XXI века:
Каким быть программному обеспечению?

В конце 2006 – начале 2007 года наше общество всколыхнула ситуация, сложившаяся вокруг директора школы села Сепыч Верещагинского района Пермского края Александра Михайловича Поносова

> Человечество, очевидно, должно выбрать одно из двух: или совершить самоубийство, или научиться жить как одна семья
>
> **А. Тойнби**

Вкратце напомню, что в основе этого дела лежали вопросы, связанные с обвинением его в незаконном использовании программ фирмы Microsoft в данной школе [1]. Если взглянуть глубже, то можно увидеть, что конфликт стал следствием вставшего перед обществом нового вызова – каким быть программному обеспечению в XXI веке, – на который оно должно дать ответ.

Идею «Вызов-и-Ответ» применительно к развитию цивилизаций предложил Арнольд Тойнби в своей работе «Постижение истории». Он писал: «Вызов побуждает к росту. Ответом на вызов общество решает вставшую перед ним задачу, чем переводит себя в более высокое и более совершенное с точки зрения усложнения структуры состояние» [2].

Настало время поиска ответа...

Программное обеспечение в первом приближении: категории

> Если не я за себя, то кто за меня?
> Если я только за себя, то зачем я?
> Если не сейчас, то когда же?
>
> **Гиллель**

Чтобы понять суть вызова, необходимо рассмотреть категории существующего ПО. На рисунке приведена диаграмма, позаимствованная с сайта проекта GNU, которая наглядно показывает, какие категории программного обеспечения существуют, а также их соотношение относительно друг друга. Опираясь на нее и на статью «Категории свободных и несвободных программ», мы можем четко говорить о двух категориях программного обеспечения: свободном и несвободном [3].

Что касается «ПО с открытыми исходными текстами» или Open Source, то, на мой взгляд, оно является или свободным (если удовлетворяет четырем принципам свободного программного обеспечения – о которых будет сказано далее), или несвободным (если не удовлетворяет данным принципам).

Кроме того, часто знакомясь с тем или иным материалом, обратил внимание, что имеет место использование термина Open Source вместо Free Software. Считаю необходимым заострить на них внимание.

Первый употребляется для обозначения программного обеспечения с открытым исходным кодом, а второй – для определения свободного программного обеспечения. Очевидно, что понятие Open Source гораздо шире понятия Free Software. Например, оно включает полусвободное программное обеспечение, имеющее ряд ограничений, не удовлетворяющих принципам свободного программного обеспечения. Поэтому использование термина «ПО с открытыми исходными текстами» (Open Source) вместо термина «Свободное программное обеспечение» является ошибочным, приводящим к неоднозначному толкованию и смешению понятий.

Рассмотрим категории. Свободное программное обеспечение – это то, которое удовлетворяет четырем принципам:

- Свобода запускать программу в любых целях (свобода 0).
- Свобода изучения работы программы и адаптация ее к вашим нуждам (свобода 1). Доступ к исходным текстам является необходимым условием.
- Свобода распространять копии так, что вы можете помочь вашему товарищу (свобода 2).
- Свобода улучшать программу и публиковать ваши улучшения, так что все общество выиграет от этого (свобода 3). Доступ к исходным текстам является необходимым условием.

Таким образом, программа будет свободной тогда, и только тогда, когда пользователь обладает всеми четырьмя свободами. Но, кроме того, можно выделить ряд разновидностей свободного программного обеспечения. Это:

- свободное программное обеспечение, подчиняющееся «авторскому леву»;
- свободное программное обеспечение, не подчиняющееся «авторскому леву».

Первая разновидность использует метод «авторского лева» или Copyleft (копилефт). Название последнего отражает противопоставление «авторскому праву» или Copyright (слово left – левый, употребляемое вместо омонима right – право в юридическом значении; право, правый и другое, подчеркивает это). Последний необходим для того, чтобы программное обеспечение оставалось свободным, так как, например, он требует, чтобы все изменения, вносимые в него, а также все новые версии оставались свободными. Отметим, что копилефт не означает отказ от авторского права на программное обеспечение – это один из способов его использования. Метод копилефт имеет несколько форм.

Первая, например, отражена в лицензии **GNU General Public License (универсальная общественная лицензия GNU)** – недопущение ограничения или отмены для любого пользователя принципов, являющихся фундаментом свободного программного обеспечения (запускать для любых целей, изучение и адаптация, распространение копий и модификация). С неофициальным переводом третьей версии лицензии можно ознакомиться по адресу: http://code.google.com/p/gpl3rus.

Другая форма копилефт закреплена в **GNU Lesser General Public License (стандартная общественная лицензия ограниченного применения GNU)**. Эта лицензия позволяет связывать с данной библиотекой или программой программное обеспечение, несовместимое с GNU GPL. Но есть одно условие – последнее не должно быть производным от лицензированного под LGPL/GPL ПО, за исключением лишь связывания. Таким образом, действие копилефт распространяется только на свободную программу и не распространяется на связываемое программное обеспечение. С текстом лицензии версии 3 можно ознакомиться по адресу: http://www.gnu.org/licenses/lgpl.html.

Еще одна форма копилефт нашла отражение в лицензии **GNU Free Documentation License (лицензия свободной документации GNU)**. GNU FDL ориентирована на защиту документации, руководств, распространяемых с программой, и позволяет копировать, тиражировать, вносить изменения, объединять документы. Она может быть применена «к любому руководству пользователя или иному произведению на любом носителе, которое в соответствии с уведомлением, помещенным правообладателем, может распространяться на условиях настоящей лицензии» [4]. Например, в нашей стране книги под лицензией GNU FDL выпускает компания ALT Linux [5]. Ознакомиться с текстом последней версии лицензии (1.3) можно по адресу: http://www.gnu.org/licenses/gfdl.html, а также с русским переводом версии 1.2 – http://ru.wikisource.org/wiki/GNU_FDL.

Вторая разновидность свободного программного обеспечения не использует метод «авторского лева». Наиболее ярким представителем являются «лицензии в стиле BSD». Эти лицензии не требуют того, чтобы изменения, вносимые в программу, также были свободным программным обеспечением. Что, в свою очередь, приводит к созданию на базе свободного ПО несвободных модификаций. В то же время само данное ПО будет оставаться свободным пока удовлетворяет четырем принципам, о которых мы говорили выше.

Приведем примеры лицензий на свободное программное обеспечение, не подчиняющиеся «авторскому леву»:

- BSD License (http://www.freebsd.org/copyright/freebsd-license.html);
- MIT License (http://www.opensource.org/licenses/mit-license.php);
- Boost Software License (http://www.boost.org/LICENSE_1_0.txt);
- Modified BSD License (http://www.xfree86.org/3.3.6/COPYRIGHT2.html#5);
- Cryptix General License (http://www.cryptix.org/LICENSE.TXT);
- X11 License (http://www.xfree86.org/3.3.6/COPYRIGHT2.html#3);
- Xfree86 1.1 License (http://www.xfree86.org/current/LICENSE4.html).

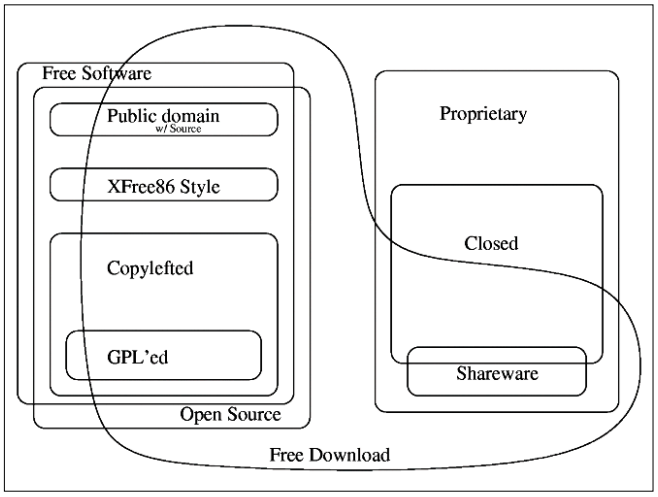
Внимательный читатель может справедливо заметить: насчет свободного программного обеспечения ясно, а есть ли свободные операционные системы (все составляющее ПО которых удовлетворяют четырем принципам)? Да, свободные системы есть. Сегодня таких дистрибутивов GNU/Linux девять [6]:

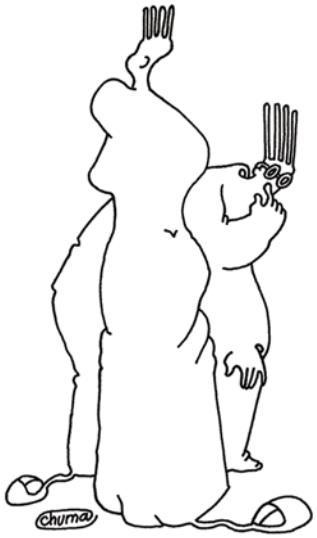
- BLAG;
- Dragora;
- Dynebolic;
- gNewSense;
- Kongoni;
- Musix GNU+Linux;
- Trisquel;
- Ututo;
- Venenux.

О некоторых из них мы уже писали как в журнале, так и в электронном приложении Open Source [7].

Перейдем к рассмотрению несвободного программного обеспечения. Оно имеет больше разновидностей, нежели свободное. Выделим их (основываясь на уже упоминав-

Категории программного обеспечения. Взято с сайта GNU





Использование ПО в отрыве от понимания социальных последствий опасно для частной жизни людей

шейся статье «Категории свободных и несвободных программ»):

- **Полусвободное программное обеспечение (Semi-free software)** – характеризуется разрешением частным лицам использовать, копировать, распространять и вносить изменения, в том числе распространять модифицированные версии в некоммерческих целях (например, PGP – http://www.pgp.com). Эти условия заметно лучше для пользователя, в отличие от ограничивающих собственническую программу, но не делают ее свободной.
- **Проприетарное (собственническое) программное обеспечение (Proprietary software)** – отличается либо запретом использования, распространения или модификации, либо требует для этого запроса специального разрешения, которое все равно не сделает программное обеспечение свободным. Таким образом, владелец прав на данное ПО выступает монополистом в области его использования, распространения и внесения изменений. Это имеет как видимые, так и менее заметные, но глобальные последствия – о них ниже. Примером данного ПО может служить операционная система MS Windows.
- **Бесплатное программное обеспечение (Freeware)** – обозначает программное обеспечение, которое можно распространять, но нельзя модифицировать. При этом нет доступа к исходным текстам программы. Поскольку сам термин не имеет четкого определения, не стоит его путать с Free software или использовать для его обозначения. Одной из таких программ является Foxit Reader (http://www.foxitsoftware.com).
- **Условно-бесплатное программное обеспечение (Shareware)** – также не является свободным и даже полусвободным. Копии этого ПО можно распространять, но пользователь или организация, которые хотят его использовать, должны заплатить за лицензирование или испытывать ограничение в использовании/функционале продукта. Shareware-программой является, например, WinRar (http://www.rarlab.com).

Часто, говоря о собственническом программном обеспечении, употребляют понятие «коммерческое ПО». Это является ошибкой. Коммерческим может быть как свободное, так и несвободное программное обеспечение.

Например, в преамбуле GNU General Public License прямо сказано: «When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.» («Когда мы говорим о свободном ПО, мы говорим о свободе, а не цене. Наши лицензии спроектированы так, чтобы удостовериться в вашем праве распространять копии свободного ПО (и взимать за это плату по своему желанию), чтобы вы получали исходный код или могли получить его при желании. Чтобы вы могли изменять ПО или использовать его части в новых свободных программах и чтобы вы знали, что вы можете это сделать.»)

Приведем пример. Например, разработчики выпускают свободное программное обеспечение и продают услуги по его технической поддержке. В этом случае данное ПО, оставаясь свободным, становится еще и коммерческим. Таким образом, коммерческое и собственническое программное обеспечение – это не одно и то же.

Программное обеспечение во втором приближении: сущность и последствия применения

> ...вон там оно спряталось, за большим деревом, и бросает оттуда в темный лес свои ранние утренние косые лучи...
>
> **М. Пришвин**

В 50-х годах XX века Норберт Винер в работе «Кибернетика и общество» подчеркивал: «Я пишу эту книгу главным обра-

зом для американцев, в жизненных условиях которых вопросы информации будут оцениваться в соответствии со стандартным американским критерием: цена вещи измеряется товаром, на который она будет обменена на свободном рынке... Удел информации в типичном американском мире состоит в том, чтобы превратиться в нечто такое, что может быть куплено или продано....

В мою задачу не входит скрупулезный разбор того, является ли эта торгашеская точка зрения моральной или аморальной, невежественной или разумной. Моя задача состоит в том, чтобы показать, что эта точка зрения приводит к неправильному пониманию информации и связанных с ней понятий и к дурному обращению с ними» [8].

Любопытно, что это утверждение было сделано на заре пятой информационной революции, характеризующейся использованием средств цифровой вычислительной техники в социальной практике. Норберт Винер уже тогда предупреждал об опасности такого подхода к информации. Посмотрим, какое отношение это имеет к программному обеспечению.

Сейчас развитые страны находятся в стадии перехода от постиндустриального к информационному обществу. Именно они во многом определяют развитие ИТ-технологий. Как известно, в информационном обществе производство информационных продуктов и оказание информационных услуг являются основными в социально-экономической области. Одним из основных направлений развития такого общества является создание и «развитие интеллектуальных систем и технологий, их массового применения непрофессиональными пользователями» [9]. Но основная проблема здесь кроется вот в чем: с объемом продуцируемой и используемой информации в экспоненциальной зависимости находится благосостояние общества [10]. Таким образом, если компания монопольно владеет программным обеспечением, как в случае собственнического ПО, то это не может не сказаться на положении пользователей в данном обществе. При использовании свободного программного обеспечения данная проблема не актуальна.

Кроме того, являясь монополистом того или иного программного обеспечения, собственник данного ПО выступает также монополистом средств производства (например, операционная система и среда разработки – напомню, что речь идет об информационном обществе), а возможно, и предметов труда. Как сказал Ричард Столлмен, основатель Фонда свободного программного обеспечения (http://www.fsf.org) и проекта GNU (http://www.gnu.org): «Если вы используете проприетарную программу или чей-то веб-сервер, то вы беззащитны. Вы становитесь «удобным материалом» в руках того, кто разработал эту программу» [11].

Как известно, отношение к средствам труда порождает отношения собственности. Например, когда разработчик компании теряет свои права в отношении создаваемого им кода «в обмен» на зарплату. Это не может не сказаться негативно на производительных силах общества.

Какими же средствами достигается обеспечение монопольного положения в сфере информационных технологий? Средствами защиты так называемой интеллектуальной собственности. На некорректность данного термина и подхода уже давно было указано Ричардом Столлменом [12]. Кроме того, одной из составляющих этой концепции является патентное право. К слову сказать, в России и в Европе патенты на программное обеспечения не применяются. А вот в США получили распространение. В 2009 году против патентования математических идей и программного обеспечения выступил Дональд Кнут, он даже назвал такую практику разрушительной [13].

Помимо правовых форм защиты могут применяться технологические. Например, DRM или технологические средства защиты авторских прав. Или Regional Protection Code (региональный код защиты), который представляет собой маркировку диска взаимозависимости от зоны распространения. В отношении последней Линус Торвальдс писал: «Это прекрасный пример того, как закон о защите ИС (интеллектуальной собственности. – **Прим. автора**) используется не для внедрения новшества, а для защиты места на рынке, для контроля за тем, что могут и чего не могут делать пользователи. Пример порочного использования закона об ИС» [14].

Собственническое отношение к информации, программному обеспечению неизбежно приводит к тем или иным попыткам получения доступа к частным данным граждан. Например, компания Microsoft стала широко применять систему удаленной проверки компьютера пользователя – Windows Genuine Advantage (WGA), которая является обязательной. Она собирает информацию о модели и производителе компьютера, серийном номере жесткого диска, региональных и языковых параметрах операционной системы, ее версии, локальных настройках пользователя, о ключах продуктов Windows и MS Office и другую [15]. Впрочем, не так давно в окружной суд Вашингтона с обвинением в незаконном сборе личной информации пользователей компанией Microsoft обратились представители ряда компаний [16].

На государственном уровне также имеет место подобная практика. Например, в Евросоюзе действует директива, позволяющая осуществлять резервное хранение данных (телефонные звонки, работа в Интернете и т.д.) и хотя эти действия оправдываются борьбой с преступностью, разницы между «невиновен» и «подозрителен» не делается [17]. Кстати, существует организация, которая занимается защитой неприкосновенности личной жизни от вторжения со стороны правительств и корпораций – Privacy International (PI – http://www.privacyinternational.org). В Австралии используется специальная база данных OneSchool (http://education.qld.gov.au/oneschool), которая позволяет получить информацию о личных и контактных данных, академических докладах, записи о внеклассных достижениях студента, поведении учащихся (положительное и отрицательное), контактные данные родителей/опекунов. Подобных примеров можно приводить множество, но очевидно одно – наше общество активно переступает черту, за которой начинает стираться понятие презумпции невиновности.

Таким образом, рассматриваемый вызов заключается в том, что суть обоих категорий программного обеспечения обуславливает характер и условия функционирования информационных систем, на них основанных. Определяет, каким быть обществу: закрытым и контролируемым собственниками ПО и монополистами или свободным, социальные отношения в котором будут основаны на технологиях, не ориентированных на контроль пользователей, а направ-

ленных на взаимодействие. Также определяет степень контроля государством личной жизни его граждан, в том числе информационной.

Вместо заключения...

Автор отдает себе отчет в том, что были затронуты далеко не все аспекты проблемы. Например, насколько оправдана разработка дублирующей функциональности из-за закрытости технологий? Грани и этическая сторона использования программного обеспечения для отслеживания действий людей и другие. На первый взгляд эти проблемы могут показаться слишком далекими от нужд человека, использующего достижения информационных технологий как потребитель. Но дело в том, что решение их в том или ином ключе, прямо или опосредовано, скажется на нем и его свободе.

Сейчас, когда наблюдается тенденция миграции на свободное программное обеспечение, хотел бы предостеречь от того, чтобы миграция не была переходом «из одной крайности в другую». Можно ли отделять этическую сторону от практической? Так, переход на свободное ПО из практических соображений может легко превратиться в обратный переход на проприетарные продукты.

Конечно, можно говорить о том, что, например, представители бизнеса или частные пользователи вряд ли станут переходить на свободное программное обеспечение только из этических соображений. Им это неудобно, невыгодно. Но в этом и заключается вызов: осознать, что использование ПО в отрыве от понимания социальных последствий опасно для частной жизни людей. Отдельные примеры есть уже сегодня, но гораздо труднее представить общую картину, которая может стать реальностью уже завтра... EOF

1. http://ru.wikipedia.org/wiki/Дело_Поносова.
2. http://www.lib.ru/HISTORY/TOYNBEE/history.txt#Toynbee104.htm.
3. http://www.gnu.org/philosophy/categories.ru.html.
4. http://ru.wikisource.org/wiki/GNU_FDL.
5. http://www.altlinux.org/Books:Main_page.
6. http://www.gnu.org/distros/free-distros.html.
7. Штомпель И. Обзор операционной системы gNewSense GNU/Linux 2.2 Deltah. //Системный администратор, № 5, 2009 г. – С. 60-64.
   Штомпель И. Обзор Linux-дистрибутива Trisquel 3.0 STS. //Open Source, № 051. – С. 2-4.
8. http://grachev62.narod.ru/wiener/cybsoc07.htm.
9. Колин К.К. Фундаментальные основы информатики: социальная информатика. – М., 2000 г. – С. 189.
10. Колин К.К. Фундаментальные основы информатики: социальная информатика. – М., 2000 г. – С. 176.
11. http://www.guardian.co.uk/technology/2008/sep/29/cloud.computing.richard.stallman.
12. http://citkit.ru/articles/230.
13. http://www.groklaw.net/article.php?story=20090603224807259.
14. Торвальдс Л., Даймонд Д. Ради удовольствия. – М., 2002. – С. 237.
15. http://en.windows7sins.org/#7.
16. http://www.rg.ru/2009/09/10/microsoft-site-anons.html.
17. Мир в руках Больших Братьев //CHIP, № 12, 2008 г. – С. 44-45.

Визитка

КОНСТАНТИН ЧЕРЕЗОВ, менеджер по продуктам информационной безопасности LETA IT-company

Мнимая простота антивируса

Что может быть ясней и спорней?

Это слово найдешь в лексиконе любого ИТ-специалиста, а область знаний, казалось бы, давно изучена вдоль и поперек. Так, да не так!

Что мы имеем на российском рынке?

Средство обеспечения антивирусной защиты первым развертывается в ИТ-инфраструктуре нового офиса. Антивирус – это де-факто один из столпов любой корпоративной информационной системы любой организации.

Казалось бы, что может быть проще? Ведь антивирусы – это:

Устоявшиеся лидеры-производители – спросите любого айтишника, он с ходу назовет с десяток компаний, разрабатывающих антивирусное ПО, и приведет аргументы в пользу своего выбора антивируса (как домашнего, так и корпоративного).

Самая освещенная тема в специализированной прессе и на просторах Интернета – доступная в любой момент информация о новых версиях ПО, новых угрозах, технологиях, сравнениях, тестированиях, о новых проблемах и прочее.

Технологическое совершенство антивирусного ПО (относительно других программных и аппаратных средств обеспечения безопасности) – действительно удобный и отлаженный инструмент по развертыванию решений, простая и наглядная настройка параметров защиты, широкие возможности по конфигурированию для самых требовательных запросов, четкая схема обновления антивирусных баз и самого ПО.

Доступность самого ПО – простота обоснования бюджета на приобретение или продление антивируса, гибкие политики лицензирования у каждого производителя, приемлемые стоимости, а также наличие бесплатных программ, имеющих положительную репутацию и успешный опыт применения в «боевой» обстановке.

Но, несмотря на всё это, проблемы антивирусной безопасности вот уже длительный срок продолжают оставаться одной из самых актуальных тем информационной безопасности (ИБ). Почему? Потому что антивирус – это всё же не так просто.

Лидеры продаж

В середине 2009 года Anti-Malware.ru опубликовал результаты исследования российского рынка антивирусной защиты. Рынок оценивался на основе экспертных мнений и информации, взятой из открытых источников, критерий оценки – объемы продаж в ценах для конечного пользователя.

Основные лидеры уверенно держатся рядом из года в год:

- «Лаборатория Касперского»;
- ESET;
- Symantec;
- «Доктор Веб»;
- Trend Micro;
- McAfee.

По большому счёту, результаты таких исследований часто игнорируются айтишниками, т.к. на первый взгляд не несут в себе никакой полезной составляющей для технических специалистов, эксплуатирующих тот или иной антивирусный продукт от производителя, возможно, даже отсутствующего в списках лидеров. Но на самом деле лидерство в подобных обзорах, где основным мерилом являются конкретные деньги конкретных потребителей, говорит о многом. Начиная от того, что в нынешние неспокойные времена такой лидер продаж скорее всего не объявит завтра о банкротстве и самоликвидации, оставив без поддержки своих пользователей. И заканчивая тем, что, выбрав продукт от одного из лидирующих производителей, можно быть уверенным в отсутствии большинства проблем, присущих всем не очень успешным компаниям.

Например:

Отсутствие проблем при закупке – канал продаж налажен, нештатные ситуации давно стали штатными. Скажем, бухгалтерии требуются дополнительные документы от производителя или дистрибьютора.

Высокий уровень технической поддержки – отлаженный процесс приёма запросов, ранжирования уровня критичности, четкие сроки реакции.

Подробнейшая эксплуатационная документация на русском языке, входящая в комплект поставки, и так далее.

Таким образом, лидерство в подобных обзорах действительно может

и должно являться одним из критериев выбора для технического специалиста, принимающего решение о закупке или замене антивирусной защиты для своей организации.

Отдельно хотелось бы сказать о бесплатных антивирусных программах. Зачастую они предназначены в основном для домашних пользователей. Причем не столько экономных, сколько достаточно опытных, чтобы обеспечить достойный уровень антивирусной защиты с помощью грубого технического инструмента. Иногда приходится слышать о том, что маленькие организации в качестве системы антивирусной защиты используют именно бесплатные антивирусы. Это имеет свой смысл только в том случае, если администрированием данного решения занимается грамотный технический специалист с большим опытом и высоким уровнем ответственности.

Технологические тесты

Если говорить о технологических тестированиях, проводимых, например, тем же Anti-Malware.ru, то на самом деле их результаты, с одной стороны, точны, понятны и аргументированы, а с другой – как бы не совсем показательны и однозначны.

Главная и почти единственная причина такого скепсиса – тот факт, что нередко тестированию подвергаются отдельные, конкретные технологические функции продуктов.

За наглядным примером далеко ходить не надо. Anti-Malware.ru в ноябрьском тестировании антивирусов на защиту от новейших (Zero-day) вредоносных программ проверяли комплексную эффективность антивирусов по противодействию новейшим образцам вредоносных программ, передаваемых пользователям наиболее распространенным сейчас способом – через зараженные веб-сайты. Как результат, абсолютным победителем признается специальная программа для проактивной защиты от новейших видов угроз класса HIPS (Hosted Intrusion Prevention System) DefenseWall HIPS 2.56. А на практике львиную долю всех используемых антивирусных решений составляют продукты, обеспечивающие комплексную защиту с применением различных методов детектирования и борьбы с вредоносными программами. И почти в любом из этих комплексных решений сегодня уже можно встретить и модуль класса HIPS.

При выборе антивируса специалисту совсем необязательно выбирать лучшее решение из тестируемых, достаточно выбрать из лучших, и именно в тех тестах, которые наиболее приближены к задачам, решаемым антивирусом в реальной практике.

Тенденции антивирусной защиты

У производителей антивирусных программ периодически возникает желание придумать какую-нибудь совершенно новую технологию, которая разом решит все проблемы с вредоносными программами – разработать этакую супертаблетку, которая защитит от всех компьютерных болезней раз и навсегда. Но вирусописатели тоже ребята не промах, и на каждую появившуюся на рынке технологию антивирусной защиты придумывают всё новые и новые виды вредоносного ПО, против которого она в итоге оказывается совершенно бессильной. И конца и края этому не видно, более того, судя по всему, подобное положение вещей обеспечивает работой и прибылью как одну, так и другую сторону, и в принципе всех устраивает. Тем не менее можно отметить несколько достаточно интересных явлений в области антивирусной защиты.

Вирусописатели — ребята не промах, на каждую появившуюся на рынке технологию антивирусной защиты придумывают всё новые и новые виды вредоносного ПО, против которого она в итоге оказывается совершенно бессильной

Во-первых, получает широкое распространение технология обеспечения антивирусной защиты на основе модели Software-as-a-Service (SaaS). За термином скрывается модель распространения программного обеспечения, при которой поставщик разрабатывает веб-приложение и самостоятельно управляет им, предоставляя своим заказчикам доступ к программному обеспечению через Интернет. Основное преимущество модели SaaS для конечного потребителя состоит в отсутствии затрат, связанных с установкой, обновлением и поддержкой работоспособности оборудования и программного обеспечения, работающего на нём. Сегодня многие из лидеров рынка антивирусной защиты имеют в своем портфеле решения по защите тех или иных каналов распространения вредоносного ПО на основе модели SaaS. Например, Symantec, TrendMicro, McAfee, «Лаборатория Касперского».

Во-вторых, в России начинает распространяться практика использования таких смежных к классическому антивирусу решений, как сканеры уязвимостей. Ведь по сути, чем меньше уязвимостей содержит используемое в организации ПО, тем меньше возможность успешного проникновения и распространения вредоносной программы. Более того, намечается тенденция по переходу от сканеров безопасности, как технических инструментов по сканированию уязвимостей в опытных руках специалистов, к построению в организации непрерывного процесса отслеживания свежих уязвимостей, их выявления и устранения.

В-третьих, совершенно чётко можно сказать, что практически стерлась грань между «чистым» антивирусом и сопутствующими решениями по обеспечению безопасности. В современном антивирусе помимо уже классического набора решений – персональный файрвол, HIPS, антируткит, антишпион, антиспам, контроль подключения внешних устройств и куча всего, как непосредственно в составе решения, так и в виде дополнительных утилит.

В-четвертых, есть отдельная тенденция, напрямую не связанная с техно-

логиями антивирусной защиты, но непосредственно касающаяся каждого антивируса, используемого на территории страны. Речь идет о пресловутом Федеральном законе № 152 ФЗ «О персональных данных» и подзаконных актах к нему. С 1 января 2011 года любая организация, занимающаяся обработкой персональных данных, обязана использовать средства информационной безопасности, которые прошли сертификацию ФСТЭК России, в том числе средства антивирусной защиты.

С государственным реестром сертифицированных средств защиты информации можно ознакомиться на официальном сайте ФСТЭК России http://www.fstec.ru/_razd/_serto.htm. Информация на сайте представлена в соответствии с действующими нормативными правовыми документами Российской Федерации. В государственном реестре можно увидеть, что далеко не все антивирусные решения имеют требуемый сертификат. В настоящий момент исключение составляют лишь продукты компании ESET и «Лаборатории Касперского».

Таким образом, требования российского законодательства в 2010 году будут как никогда актуальны, и, возможно, имеет смысл поинтересоваться у компании-разработчика антивируса, уже используемого в организации сейчас, о перспективах его сертификации по требованиям безопасности ФСТЭК в рамках соответствия ФЗ № 152 «О персональных данных».

Повторение – мать учения

В заключение я постараюсь перечислить несколько основных золотых правил, значительно повышающих уровень антивирусной защиты всей организации. Итак, известный факт – большинство вредоносных программ попадает в организацию извне по двум каналам: через переносные внешние устройства и через WEB.

Семь золотых правил

Первое правило. О чем необходимо позаботиться при настройке антивирусной защиты? Это определить правила обращения внешних устройств в организации. В современном офисе сотрудникам порой трудно обойтись без переносных внешних устройств хранения информации, таких как USB flash-накопители, поэтому тотальный запрет флешек зачастую только усложняет ситуацию. В идеале использование только определенных флешек разрешить можно, при этом необходимо документально закрепить каждое конкретное устройство за каждым сотрудником, политикой безопасности запретить автозапуск флешки и разрешать её запуск только после проведения сканирования на предмет наличия вредоносных программ.

Второе правило. Необходимо подумать о модернизации уже установленной системы антивирусной защиты специализированным решением по обеспечению дополнительной веб-безопасности, например, таким как Websense Web Security Suite или eSafe. Это позволит обеспечить дополнительную высококлассную защиту от шпионских программ (в том числе загружаемых «на лету»), программ-ботов и их трафика, вредоносного кода, фишинга и фарминга. Кроме того, решения позволяют блокировать каналы обратной связи шпионских программ и перехватчиков клавиатуры с их серверами-хозяевами.

Третье правило. Многовендорная защита – это аксиома. Чем больше рубежей обороны, чем они разнообразнее, тем выше вероятность выявить и предотвратить угрозу на ранней стадии. На рабочие станции нужно установить антивирус от одного производителя, на почтовые серверы – от другого, на интернет-шлюз – от третьего.

Четвертое правило. Необходимо установить отдельную политику безопасности для мобильных пользователей с ноутбуками. Антивирус на ноутбуке должен иметь персональный файервол, а также специальный режим мобильной политики безопасности – т.е. усиление защиты вне корпоративной сети. При этом ещё крайне желательно использовать в организации такую технологию, как NAC. На российском рынке данная технология представлена в основном в решениях Cisco Network Admission Control (Cisco NAC), Symantec Network Access Control (Symantec NAC), Microsoft Network Access Protection (Microsoft NAP) и Juniper Networks Unified Access Control.

Идея NAC проста. Мы хотим получить информацию о любом устройстве, которое подключается к сети (ноутбук, принтер, IP-телефон, МФУ и т.п.) из любой точки и любым способом (Ethernet, Wi-Fi, VPN и пр.), и в автоматическом режиме принять решение по данному устройству или пользователю: пустить/не пустить/пустить с ограничениями. На этом этапе возникают вопросы, связанные с проверкой устройства на соответствие корпоративным политикам доступа в сеть. Технология NAC позволит гибко настроить соответствующие правила и политики, обеспечив более высокий уровень защиты корпоративной сети.

Пятое правило. Документируй деятельность по антивирусной защите (хотя это распространяется и на другие средства). Необязательно и более того совершенно нет необходимости создавать ворохи ненужных документов по стандартам ISO 27001, но подготовить несколько очень важных кратких документов под силу каждому ИТ-специалисту. В случае с антивирусной защитой такими документами могут являться корпоративный стандарт антивирусной защиты, процедура включения нового узла/пользователя в контур антивирусной защиты. И надо обязательно прописать и постоянно модернизировать процедуру реагирования на различные вирусные инциденты: куда бежать, кому звонить, что выключать, что восстанавливать в первую очередь и как?

Шестое правило. Не нужно скупиться на расширенную поддержку от компании производителя. Как показывает практика, более сжатые сроки реакции на запросы, повышенный приоритет при решении проблем и прочие условия сильно облегчают жизнь при возникновении проблем с антивирусным ПО, а такие проблемы бывают у всех производителей.

Седьмое правило. По сути самое главное – наиболее уязвимым звеном в любой системе обеспечения безопасности являлся, является и будет являться человек. Особенно немотивированный, неосведомленный и неграмотный в вопросах обеспечения антивирусной безопасности. Как бы банально это ни звучало, но периодическое навязчивое, настойчивое и доходчивое напоминание обычным пользователям основ информационной безопасности дает гораздо больший эффект, чем внедрение самых строгих политик. EOF

Визитка

ВАЛЕРИЙ АНДРЕЕВ, заместитель директора по науке и развитию компании ИВК

Стоит только чихнуть…

Каждый сегодня может стать носителям вируса

Тема так сильно перегрета, что любое высказывание может ее едва ли не взорвать. Но дело не в эмоциях

Сегодня антивирусная защита затрагивает всех пользователей и все средства вычислительной техники (СВТ). Этот серьезный прорыв на все сегменты рынка без исключения заставляет задавать неудобные вопросы, глубже присматриваться к уже необходимой и достаточной системе защиты от несанкционированного проникновения. И ломать голову: что делать дальше?

Антивирусное ПО защищает не всех

Это связано прежде всего со временем реакции на неизвестную атаку. Первые пользователи, подвергшиеся новой атаке, страдают сильно. В это время поставщики антивирусов создают «противоядие», и после этого для тех, кто обновляет антивирус, риск гораздо меньше. Но и вирусописатели не теряют времени и успевают выпустить что-то новое…

В общем, статистическое большинство защищено прилично. Но тем, кто пострадал, от этого не легче. Разработчики вирусов дошли уже до того, что находят и используют уязвимости еще раньше, чем о них узнают разработчики ПО, антивирусные компании и ИТ-сообщество. Поэтому я сознательно использую слово «разработчики»: вирусы – это серьезный криминальный бизнес, и ресурсы, вкладываемые в поиск новых уязвимостей и разработку новых алгоритмов маскировки и проникновения, вероятно, очень значительны.

Особой угрозе подвергаются предприятия и организации, чья информация настолько ценна, что для проникновения в их сети разрабатываются специальные антивирусы, которые не распространяются вне атакуемой организации. Такие примеры уже есть. Соответственно разработчики антивируса в принципе не могут получить образцы вредоносного ПО. Конечно, должны помогать проактивные технологии, но никто точно не знает, насколько они эффективны в данном случае. К сожалению, отмеченные проблемы скорее всего могут перейти в разряд вечных, ибо решения, думается, не существует.

Малоприятные тенденции

Мы «на собственной шкуре» выявили новую тенденцию – композитные атаки. Они связаны с закладкой безобидного включения в виде обновления на диск СВТ. Атака активируется при помощи второй части вируса, заложенного в столь же безобидной корреспонденции, рассылаемой много позже с известной избирательностью. Мы все сегодня можем стать носителями вируса, для активизации которого нужен лишь «чих». Прямо как в жизни!

Еще одна неприятная тенденция – невозможность полностью удалить следы пребывания вируса на СВТ, что добавляет работы системным администраторам по переустановке общего ПО. Это очень серьезно, ведь часто теряются настройки, а то и данные. Да и время восстановления увеличивается очень сильно.

Что решаемо, что – нет

Первая проблема стара как мир. Уменьшение времени реакции на новую атаку и создание менее «тормозных» решений.

Вторая проблема новая. Это включение в антивирусные системы средств разграничения доступа (например, по портам и протоколам). Это не вполне свойственно именно антивирусным средствам и часто приводит к конфликтам с иными средствами контроля НСД, потому разобраться бывает совсем непросто.

Третья проблема пока кажется неразрешимой. Зачастую антивирусное ПО обновляется не через файлы, а требует прямого коннекта с сайтом производителя. Во многих организациях такая модель неприемлема в принципе, что ограничивает сферу применения такого ПО.

Что важно понимать сисадминам

Пожалуй, они должны понимать серьезность проблемы. Знать, что сама возможность существования вирусов связана с архитектурными просчетами при создании современных ИС. Поэтому решить вопрос без кардинального изменения архитектуры невозможно. Я вижу единственный путь – использование системообразующего ПО промежуточного слоя с обязательным набором функций. Именно такого подхода мы придерживаемся, создавая крупные автоматизированные ИС на базе нашего ПО ИВК «Юпитер». EOF

Безопасное завтра

Полной коллекции вирусов ни у кого нет

Проблемы антивирусного ПО обсуждают эксперты в области защиты от вирусов и вторжений

- **Как понять, насколько тот или иной антивирус надежен?**
- **Где их лучше тестировать?**
- **Кто должен больше думать об угрозах: пользователи или специалисты по информационной безопасности?**

КОНСТАНТИН АРХИПОВ, руководитель Panda Security в России

Не так тестируете!

Сегодня большое количество профильных компьютерных изданий проводят различного рода тестирования, исследования программных продуктов. Зачастую исследования СМИ приводят к совершенно противоположным результатам. С чем это связано? С тем, что различные издания используют абсолютно разные, подчас неверные методики исследования. Бывает так, что при тестировании к одному продукту применяют один метод, к конкурирующему продукту совершенно другой, в результате мы имеем неадекватный результат. Считаю, что для правильного адекватного тестирования должен применяться единый метод, вирусная база должна быть обновлена. Вот несколько правил, которые я хотел бы выделить особо.

1. Тест программного продукта должен проводиться на операционной системе, которая совместима с продуктом.

2. Избегайте тестирования программных решений на виртуальных машинах. Существенный процент вредоносных кодов невозможно обнаружить на виртуальном ПК, т.к. многие из них, попадая на виртуальную машину, меняют свою тактику поведения либо вообще никак себя не проявляют. При использовании реальных машин процент обнаружения будет значительно выше, что повлияет на результаты тестирования.

3. Рекомендую использовать только актуальные образцы вредоносных кодов, которые несут реальную опасность сегодня. Не очень хорошая идея использовать вредоносные образцы четырехлетней давности, которые сегодня не представляют угрозы для пользователей и не находятся в обращении. Кроме того, образцы, которые используются в тестах, должны быть правильно классифицированы.

4. Так как цель тестирования – проверить эффективность продукта, задача должна состоять в том, чтобы испытательная окружающая среда была максимально приближена к действительности. Компьютер должен быть подключен к Интернету. Человек, который проводит тестирование, должен делать те же действия, которые делает обычный пользователь, чтобы оценить эффективность защиты. Если антивирусный продукт не может обнаружить файл во время просмотра или по требованию, но препятствует тому, чтобы он был загружен, действительно ли справедливо сказать, что он не обнаружил его? Или, если антивирусное решение не может обнаружить опасный файл на любой из стадий, но может фактически обнаружить его аномальное поведение, которым кто-то управляет? В любом из этих случаев продукт защищает пользователя.

5. Вы должны быть связаны с Интернетом, чтобы правильно оценить продукт, который использует не только сигнатурные базы данных, но и базы онлайн. EOF

СЕРГЕЙ КОМАРОВ, руководитель отдела антивирусных разработок и исследований компании «Доктор Веб»

И не там, где надо!

Проблема заключается в антагонизме – с одной стороны, пользователи хотят ориентироваться на объективные данные, полученные из достоверных источников в оценке такого важного для безопасности компьютера продукта, как антивирус. С другой стороны, эти данные практически невозможно получить в лабораторных условиях.

История проблемы во многом кроется в привычке и стереотипах. Компьютеры вообще, а антивирусы тем более, отрасль относительно молодая, но и она уже обзавелась непререкаемыми авторитетами и мифами. С появлением

антивирусов возникла очень уважаемая организация Virus Bulletin, на чьи плечи легла организация тестирования антивирусов. В то время это было очень эффективное и крайне полезное предприятие. Вендоры «скидывались» вирусами, VB собирал коллекцию всех (и это была действительно почти полная коллекция) компьютерных вирусов и тестировала, как разные антивирусы справляются с этими угрозами. В таких условиях пользователь получал весьма объективную картину по интересующему его продукту.

Но времена изменились. Сейчас полной коллекции нет ни у кого, а если бы и была, она устарела бы за час. Но стереотип остался – нам, как пользователям, надо кому-то довериться, чтобы понять, насколько тот или иной антивирус надежен. А раз есть спрос, будет и предложение. Количество тестовых лабораторий постоянно растет, чего не скажешь про их объективность.

VB, поняв, что индустрию не устраивают результаты его тестов, стал реформироваться и реформируется по сей день. Помимо того, что изменилась вирусная обстановка, очень сильно изменились и сами антивирусы – теперь это большие, сложные комплексы. «Тестеры» пытаются тестировать теперь то одну, то другую функциональность антивируса, потому что протестировать весь комплекс целиком слишком накладно и по времени, и по ресурсам. А это, согласитесь, сродни оценке костюма по крепости пришитых пуговиц.

На наш взгляд, практически невозможно в лабораторных условиях протестировать антивирус и предложить такую систему критериев и оценок, чтобы пользователь мог заключить, устроил бы его этот продукт или нет. И страдает в этом случае именно пользователь, потому что он теряется. На сайте антивируса гордо висит награда VB100, а его компьютер, защищенный этим антивирусом, инфицируется в первую же неделю работы.

Нам кажется, что антивирусной индустрии стоит радикально пересмотреть свое отношение и к тестам, и к информированию пользователей – люди просто не захотят, чтобы их обманывали, причем так долго. И авторитет уважаемых тестеров будет полностью нивелирован до «инструмента маркетинга». EOF

СЕРГЕЙ НИКИТИН, маркетинговый аналитик «Лаборатории Касперского»

Пусть безопасность обеспечивают профессионалы

Технический прогресс был бы возможен и без хакеров: всегда найдутся люди, которые будут исследовать, изобретать и развивать по зову сердца, а не ради денег. И кто знает, каких высот достигла бы современная наука, если бы у далеко не худших умов высвободились силы и время, которые сегодня используются на совершение взломов и соответственно противостояние им! Тем не менее в реальной жизни киберпреступники существуют.

К сожалению, как это часто случается, преступники опережают тех, кто им противостоит: если даже правоохранительным органам удается накрыть в своей стране часть ботнета, состоящего из десятков тысяч ПК на разных континентах, то справиться с оставшимися частями и, что гораздо важнее, с его создателями не удается, так как они находятся в других странах.

Вывод из всего вышесказанного тривиален: спасение утопающих является делом самих утопающих. Применительно к киберугрозам это означает, что если человек не хочет пострадать от действий хакеров и спокойно пользоваться Интернетом, электронной почтой и так далее, он должен установить на свой компьютер защитное решение. Причем не только на свой стационарный ПК, но и на ноутбук, нетбук, коммуникатор и т.д. Несмотря на кажущуюся сложность установки защитного решения, это элементарные операции, которые не требуют больших затрат времени и денег: большую часть работы за пользователя сделали компании-разработчики защитных продуктов.

В частности, в решениях «Лаборатории Касперского» реализован автоматический режим работы, при котором пользователь может вообще забыть, что на его компьютере установлен антивирус, программа сделает все сама. Аналогично тому, как сегодня никто не возражает против того, что в каждой квартире есть дверь с парой замков, автомобили оснащены сигнализациями с брелоками, а кредитные карты имеют пин-код, в самом скором времени такое же отношение будет и к защитным решениям для компьютеров и смартфонов. Более того, на человека, который ими не пользуется, будут подозрительно коситься.

«Тестеры» пытаются тестировать теперь то одну, то другую функциональность антивируса. А это, согласитесь, сродни оценке костюма по крепости пришитых пуговиц

Защита корпоративных сетей и информационных ресурсов требует немалых усилий, но они оправдываются сторицей. Компьютер удобнее пишущей машинки, электронная почта быстрее и функциональнее обычной, а с помощью виртуальных сетей сотрудники офисов, находящихся в разных городах, работаю так, будто сидят на одном этаже. Естественно, если вся система построена правильно, то КПД сотрудников компании растет. Однако если эта инфраструктура в результате атаки киберперступников «встанет» или «рухнет» и произойдет утечка персональных данных сотрудников или клиентов, то мало не покажется не только ИТ-отделу, но и всей компании. Чтобы избежать этого, нужно потратить время и средства на защиту серверов и рабочих станций, разработку стратегии ИБ и так далее. В таком случае киберпреступники останутся за границами защитного периметра, тогда как внутри будет кипеть продуктивная работа. Именно так «Лаборатория Касперского» видит современную ситуацию с информационной безопасностью. EOF

Какие требования вы предъявляете к антивирусному ПО?

На вопрос «СА» отвечают ИТ-специалисты

Владислав Котусов, директор по информационным технологиям, компания Softline

Высокое качество, приемлемая цена

Главное в нем – надежность, масштабируемость, производительность, мультизадачность, приемлемая цена, простота работы для пользователей и администрирования.

Мнение с форума журнала

Без шаманства и глюков

1. Чтобы не мешало работать.
2. Чтобы имело нормальный интерфейс.
3. Чтобы легко настраивалось на различные источники обновлений.
4. Чтобы не глючило.
5. Чтобы устанавливалось и удалялось без шаманств и рысканий по реестру.

Павел Наседкин, программист «1С», компания «БАЗЭЛ недвижимость»

Не люблю «тормозных» версий

Самое важное, чтобы антивирус защищал нормально, работал быстро, не было ничего лишнего. Лидирует ESET NOD32 – быстро работает, база большая, нет ничего лишнего типа Internet Security. «Антивирус Касперского» – большая база, есть версия без лишних «примочек», но очень медленно работает. К тому же наблюдались ошибки при работе с «1С». Мне «Антивирус Касперского» всегда больше всех нравился, но после версии 3.0 он стал очень уж «тормозным», так что я на NOD 32 окончательно перехожу.

Мнение с форума журнала

Антивирус не должен быть наглым

1. Он может иметь свои настройки уведомлений по умолчанию, но должен давать возможность настроить и его уведомления, и поведение.
2. Должен устанавливаться также из безопасного режима (если говорим о популярных ОС).
3. Самое важное – не должен тормозить работу (особенно если у кого-то не очень современный ноут – на нём иногда поставить антивирус просто невозможно, так как работать после этого будет нельзя).
4. Если у пользователя нет файервола – желательно, чтобы антивирусное ПО давало минимальные возможности для контроля автозапуска и доступа приложения в сеть.

Алексей Сечной, ИТ-инженер, компания «РТС-Инжиниринг»

Защита – важнее быстродействия

Чтобы ловил все вирусы и как можно меньше при этом нагружал систему. Я сторонник «Антивируса Касперского». Он меня полностью устраивает. Использую на работе и домой куплена лицензия. Я считаю, что защита главнее, чем быстродействие. Версия 4 да, была медленной. Но с тех пор компания Касперского достигла значительных успехов в снижении нагрузки на систему.

Мнение с форума журнала

Чтобы было понятно без слов

Антивирусное ПО должно отвечать следующим требованиям:

1. Защита от вирусов на достойном уровне, актуальность и быстрая реакция на новые вирусы.
2. Высокая скорость работы, невысокие системные требования.
3. Гибкость настроек, в том числе приемлемые настройки по умолчанию.
4. Интуитивно понятный интерфейс.

Юрий Кишалов, ИТ-консультант

Антивирус должен быть незаметным

Наиболее важным параметром антивируса для меня является качество защиты информации (операционной системы, ПО, пользовательских данных), т.е. определение и обезвреживание по возможности всех известных вирусов и оперативное обновление антивирусных баз. Вторым параметром по важности назову «незаметность для пользователя» – сюда входят простота настройки, скорость работы и минимальное количество вопросов к пользователю. Антивирус Dr.Web более всего удовлетворяет моим требованиям. На домашнем компьютере использую его. И, конечно же, рекомендую всем друзьям и клиентам.

Мнение с форума журнала

Гибкость, скорость, простота

Мои требования – быстрая работа, гибкие настройки и пресеты, простой и понятный интерфейс, небольшой трафик обновления.

Юрий Годына, руководитель департамента биллинга, компания «Мастертел»

Пользователь не должен испытывать дискомфорта

Безусловно, на первый план выступают функциональные возможности и качество выполнения необходимых задач. При этом важно, чтобы ПО не было слишком перегружено – зачем платить за опции, которыми не пользуешься?

Вторым фактором по степени важности идет надёжность. Особенно это касается серверного ПО, обеспечивающего круглосуточную работу сервисов.

Третьим важнейшим свойством для ПО является его эргономичность и простота в использовании и администрировании. Как негативный фактор в работе ПО можно обозначить длительность обучения и сертификацию пользователей ПО, поскольку обладание особыми познаниями в области ИТ не должно становиться необходимостью и пользователь не должен испытывать дискомфорта при использовании ПО.

Ещё одно важное качество – ПО должно быть хорошо документировано, в том числе иметь развитую систему контекстной помощи и разного рода всплывающих подсказок. И наконец, ПО должно быть интегрируемым и стандартизованным.

Мнение с форума журнала

Ценю за неподкупность

Требования к антивирусу – работа на уровне ядра системы. Отсутствие «пользовательских» настроек – только от лица пользователя, созданного при установке. Простой и понятный интерфейс управления – как у Dr.Web для Macintosh. Настройки – в plain-text. Настраиваемые обновления, то есть чтобы была возможность менять источник обновлений. Надёжность, работа в изолированном системном пространстве. «Неподкупность» – то есть отсутствие в базах зарегистрированных малварей и прочих супостатов.

Артем Сычев, начальник управления информационной безопасности, ОАО «Россельхозбанк»

Лекарство от заразы полезно всем

Во-первых, в случае с антивирусными средствами просто необходим принцип эшелонированной обороны, так как периметр сети, почтовые и файловые сервера, рабочие станции и мобильные клиенты подвержены разного рода угрозам. Во-вторых, представляется актуальным строить защиту от вредоносного программного кода на многовендорной основе. Практика показывает, что нередко вредоносные программы, с легкостью определявшиеся и нейтрализованные антивирусным средством одного производителя, были также с легкостью не замечены антивирусным средством другого вендора. Кроме того, имеет место разрыв по скорости выпуска антивирусных сигнатур у разных вендоров. В этом случае противодействовать массированной атаке возможно, только если в вашем арсенале есть несколько взаимодополняющих антивирусных средств. В-третьих, надо признать, что антивирусное программное обеспечение – это уже лекарство от «полученной заразы». Задача же службы ИБ минимизировать источники ее проникновения. И далеко не всегда это достигается установкой антивирусных средств. Редко обновляющиеся антивирусные базы на рабочей станции, с неконтролируемыми портами, с разрешенной автозагрузкой со съемных носителей информации – могут на долгое время вывести из строя всю сеть. В-четвертых, немаловажным условием успешного внедрения и функционирования антивирусных средств в крупной многофилиальной организации является наличие механизма централизованного управления и анализа. Отсутствие таких механизмов ставит антивирусную защиту в зависимость от аккуратности выполнения специалистами своих обязанностей. В-пятых, мобильность. Причем как мобильность сотрудников предприятия (ноутбуки, удаленные рабочие места и т.д.), так и мобильность клиентов. Таким образом, за пределами «контролируемой зоны» оказывается большое количество потенциально уязвимых рабочих мест. Поэтому, оценивая инвестиции в обеспечение информационной безопасности, надо не забывать о повышении осведомленности персонала и клиентов в вопросах антивирусной защиты.

Мнение с форума журнала

Только без ручек!

1. Важно, чтобы был грамотный файервол, настраиваемый ручками (никаких автоматических режимов и режима обучения, ибо всё это некорректно работает). По умолчанию должно блокироваться всё! Также должны быть импорт/экспорт настроек и применение их через групповые политики. А еще лучше, чтоб был единый центр, устанавливаемый на сервер, с которым бы происходила периодическая синхронизация правил доступа. В таком случае было бы неплохо иметь возможность задавать различные правила для разных клиентов.
2. Чтоб обновление версии антивирусной программы происходило автоматически при выходе новой версии. А не как сейчас у всех есть (Kaspersky, NOD32, Dr.Web): ручками закачиваешь новую версию и ручками же (различными способами) обновляешь на всех компьютерах.
3. Уже был упомянут безопасный режим в связи с установкой антивирусника. Так вот было бы просто замечательно, если бы он еще и мог сканировать/лечить в этом самом режиме!
4. И хотелось бы более быстродействующий антивирус с наименьшим потреблением ресурсов компьютера (RAM, CPU, HDD). Но это уже из области фантастики!

Визитка

МИХАИЛ ДАНЬШИН, эксперт в области ИТ. Специализируется на Exchange и смежных технологиях. Ведет блог (http://danshin.ms), выступает на конференциях и в MCP-клубе.Награжден премией Microsoft MVP

После катастрофы

Теория и практика восстановления Exchnage Server

Почта в организации является одним из основных рабочих инструментов. Что будет, если вы его лишитесь? Готовы ли вы к такому повороту событий?

Данной публикацией я хочу начать цикл статей, посвященных восстановлению Exchange Server. Вы узнаете о том, что нужно сделать, чтобы сбой сервера не застал вас врасплох, как правильно разработать план аварийного восстановления Microsoft Exchange Server и о чем следует позаботиться задолго до того, как произойдет авария. Я расскажу о «подводных камнях», которые встретятся у вас на пути. Вы также узнаете, какие действия нужно предпринять после восстановления, и сможете ответить на следующие вопросы:

- Какое оборудование и ПО нужно использовать, чтобы иметь возможность восстановления почтовой системы после сбоя?
- Сколько времени займет восстановление почтовой системы после отказа почтового сервера?
- Смогут ли пользователи продолжить работу с почтой после отказа почтовой системы?
- Что произойдет с почтой, которая была отправлена на ваш адрес, в то время когда ваша почтовая система находится в нерабочем состоянии?
- Что нужно будет предпринять пользователям после восстановления почтовой системы?
- Сохранятся ли правила обработки почтовых сообщений на сервере и клиенте после восстановления?
- Стоит ли использовать отказоустойчивые серверы и серверы «горячей» замены или можно обойтись восстановлением после аварии?

Удобство использования электронных сообщений не нуждается в описании. А вот задуматься, что произойдет, если по какой-то причине мы не сможем обеспечить данный сервис, стоит. Менеджер вовремя не отправит клиенту договор – крупная сделка может сорваться. Директор не получит отчет – поздно отреагирует на динамично меняющуюся ситуацию в бизнесе. Вы не оповестите своих клиентов и не сообщите важную информацию, над которой маркетологи работали многие месяцы, и можете остаться незамеченными, конкуренты опередят вас.

Оценить ущерб от остановки почтовой системы очень сложно. Но даже если вы не знаете точных цифр, уверяю вас, это парализует деятельность большинства современных организаций и нанесет финансовый вред компании.

Как вы думаете, где большинство менеджеров хранит важные документы, которые когда-то получили в виде вложений, по электронной почте? Конечно же, в своем почтовом ящике! А вам никогда не приходилось искать старое сообщение, чтобы скопировать оттуда часть информации? Думаю, приходилось. Несложно представить, что будет, если мы вдруг лишимся всего этого.

Сценарий

За основу для статьи возьмем следующую схему. Назовем нашу вымышленную организацию CONTOSO. В ее главном офисе два сервера – CON-HQ-DC-01 и CON-HQ-EX-01. На первом сервере установлены следующие роли: контроллер домена, DNS, DHCP, подключено лентозаписывающее устройство для хранения резервных копий, а на втором – почтовый сервер Exchange. Все наши серверы выпущены известными фирмами-производителями, правильно настроены и не имеют проблем с оборудованием. Везде используется операционная система Windows Server 2003 SP2. Резервное копирование всех почтовых хранилищ осуществляется ежедневно при помощи программы NTBackup. За правильностью создания резервных копий ведется постоянное наблюдение (см. рисунок).

Все остальные серверы и роли, такие как файловый сервер или веб-сервер, не оказывают прямого влияния на рассматриваемую проблему, поэтому мы не будем их учитывать.

Описанная схема организации не является оптимальной, но для простоты изложения и понимания материала я решил опустить тонкости планирования ИТ-инфраструктуры предприятия. В будущих статьях я надеюсь уделить внимание проблеме правильной организации ИТ-инфраструктуры с использованием Exchange Server.

План восстановления

Прежде чем приступить к описанию действий, которые нужно предпринять после аварии, необходимо рассказать, о чем нужно позаботиться заранее, до происшествия.

Что нужно сделать, чтобы сбой сервера не застал вас врасплох, как правильно разработать план аварийного восстановления?

Прежде всего нужно позаботиться о резервном копировании наших данных, чтобы было что восстанавливать.

Также нам понадобится резервное оборудование, которое мы будем использовать взамен вышедшего из строя. И конечно же, нужен чёткий план действий.

Но это не все… Давайте опишем все подробно.

Существует такое понятие, как Disaster Recovery (восстановление после катастрофы). Под катастрофой понимается ситуация, при которой часть или все серверы нашей организации полностью выведены из строя или уничтожены. Например, серверную затопило, сломался кондиционер и возник пожар, ураган, землетрясение – все что угодно. В наших широтах ураган и землетрясение – явления нечастые, а вот «затопило» или «пожар» случаются каждый день.

Но катастрофой могут быть не только масштабные разрушения. Иногда это потеря важного документа или письма. На подобные случаи есть резервное копирование, VSS и другие подобные механизмы. Раз уж мы рассуждаем категориями «сервер», «база», то данное упоминание здесь, попросту говоря, лишнее.

Хорошо иметь план восстановления после катастрофы или Disaster Recovery Plan (DRP). В описываемом мною сценарии, а именно потери почтового сервера, катастрофа по своим масштабам выглядит не так ужасно, но по значимости она равна потере всех серверов компании, т.к. в некоторых случаях способна полностью парализовать ее деятельность. Посудите сами, зачем вам нужны будут остальные сервисы, такие как AD, DNS, DHCP, если почтовая система вышла из строя? Конечно, возможно, будут доступны такие ресурсы, как файловый сервер и веб-сервер, но для нас они не столь критичны, как почта.

Подробный план восстановления поможет сократить время, которое мы затратим на «оживление» нашей почтовой системы после сбоя. А время, как известно, деньги. Кроме того, имея на руках подробный план восстановления и детальное описание тех действий, которые необходимо предпринять, вы сможете сэкономить не только время, но и нервы. Попробуйте заняться чем-то важным, когда телефон разрывается от звонков, гневные пользователи требуют восстановить работу, а начальник стоит над душой. Особенно, если вы точно не знаете, что делать.

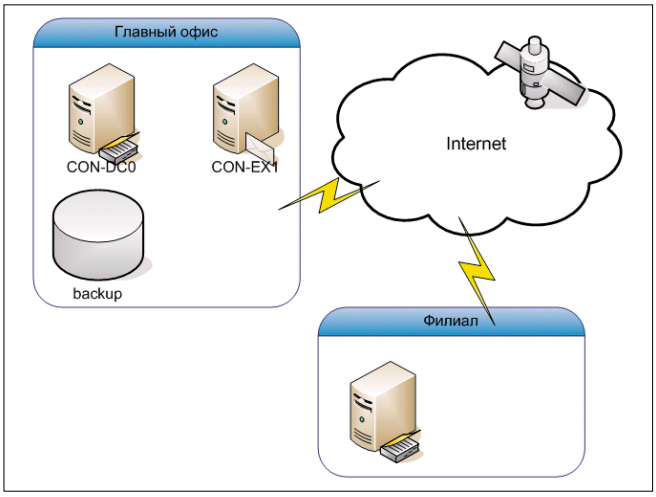
Как же должен выглядеть план восстановления?

Что включить в план восстановления?

План восстановления обязательно должен включать в себя следующее:

1. Подробное описание всех серверов, которые планируется восстанавливать. Частой ошибкой системных администраторов является недостаточное документирование серверов. В самый критический момент они не могут вспомнить, как были размечены диски, какой фирмы использовались жесткие диски, какие обновления были установлены, какой IP, шлюз и DNS были в сетевых настройках сервера и т.д. В нашем случае отсутствие этих данных может стоить нам очень дорого, так как без этой информации восстановить Exchange Server невозможно! Поэтому позаботьтесь о том, чтобы в плане восстановления была следующая информация:

Схема нашей вымышленной организации



- конфигурация RAID-контроллера;
- конфигурация логических дисков и разметка разделов;
- сведения обо всем оборудовании – его модель и наименование производителя;
- конфигурация всех сетевых интерфейсов, включая таблицы маршрутизации;
- информация обо всем установленном ПО и обновлениях;
- пути хранения и названия файлов резервных копий.

Примечание: план восстановления может не содержать подробного описания всей ИТ-инфраструктуры компании, но при наличии такого документа лучше упомянуть о том, что он есть и где его взять.

Говоря об описании серверов, хочется упомянуть о хорошей практике составления так называемых паспортов устройств. Паспорта устройств составляются не только на серверы, но и на все конфигурационные элементы вашей ИТ-инфраструктуры, такие как пользовательские рабочие станции, принтеры. Паспорта устройств могут включать в себя не только информацию о конфигурации, как описано выше, но и все, что вы сами захотите включить в него. Например, вы можете указать, в какой комнате расположено устройство, кто является ответственным за контент и кого необходимо оповестить в случае возникновения проблем.

2. Подробные инструкции по восстановлению. Снабдите ваш план подробными инструкциями, описывающими весь процесс восстановления по шагам. Лучше, если план будет содержать снимки экранов. Имея такой план восстановления, новый сотрудник, который только начинает знакомиться с вашей ИТ-инфраструктурой, гораздо быстрее сможет войти в работу. А подробные действия, описанные в плане, минимизируют риск ошибки.

3. Пароли. Если вам знакома ситуация, когда в самый неподходящий момент оказывается, что пароль от сервисной учетной записи потерян, то вы меня поймете. Проверьте себя – помните ли вы пароль восстановления Active Directory? Если нет, то, возможно, предыдущий системный администратор уволился, не оставив пароль, или вы просто его забыли или не знали. Мне кажется, что вам не помешает следующая информация.

Со временем ИТ-инфраструктура обрастает так называемыми служебными учетными записями. Например, учетная запись, от имени которой делается резервное копирование. Или учетка для запуска какой-то службы. Пароли от таких записей обычно не держат в голове, а хранят в «паролехранилке». Это может быть зашифрованный файл или, на худой конец, простая электронная таблица. Каждый для себя решает сам. Лишним будет объяснять, что пароли необходимо хранить в безопасном месте.

Но чем безопаснее место, тем оно более труднодоступно. Необходим компромисс. Я рекомендую поступать следующим образом. Храните ваши пароли на бумажке! Бумажку с паролем запечатайте в конверт, а конверт снабдите печатями и подписями. Запечатанный конверт можно прикрепить к серверу, а копию хранить в сейфе руководителя, чтобы пароли при пожаре не сгорели вместе с сервером.

Такой способ хранения паролей удобен еще в нескольких ситуациях. Например, когда вас нет в офисе. Ну какие там пароли, когда вы занимаетесь сёрфингом где-то на Карибах, а вашему заместителю он вдруг понадобился? Это удобно еще и в случае, когда вам нужно сообщить пароль лицу, временно допущенному к администрированию. Например, для обновления такого ПО, как «Консультант+» или «1С». После того как конверт будет вскрыт и пароль станет известен кому-то, кроме вас, вы сможете сменить его, заново распечатать бумажку и опять сохранить в надёжном месте.

4. Изменения. Помните, что как бы ни была хороша ваша ИТ-инфраструктура, рано или поздно вы сделаете какие-нибудь изменения. Установите новое оборудование или программное обеспечение. Измените имя сервера или установите новый. Очень важно отслеживать подобные изменения и своевременно вносить их в план восстановления. Когда у вас будет правильно составленный план восстановления, чтобы не распечатывать его каждый раз при внесении изменений, вы можете использовать так называемый лист изменений. В нем можно отражать, кто, когда и какое именно изменение проделал. А подшить один листок гораздо проще, чем заново напечатать весь план.

5. Список лиц для оповещения. Также обязательно нужно включить список ответственных лиц и их телефонов, по которым необходимо звонить в случае ЧП. Это особенно актуально, когда в организации работает больше одного администратора и функции по обслуживанию оборудования, сети и ПО распределены.

Безусловно, список оповещения сократит время решения инцидента. Так как оператор точно будет знать, кому и в каких случаях звонить, информацию об инциденте скорее смогут получить ответственные люди.

6. Сценарии. Я рекомендую включать в план восстановления сценарии, которые могут произойти во время эксплуатации системы. Например, у вас могут быть заранее заготовлены планы на следующие случаи:

- потеря почтового сообщения;
- потеря почтовой базы данных;
- потеря группы хранения;
- потеря хранилища логов;
- потеря почтового сервера;
- выход из строя кластера;
- потеря контроллера домена Active Directory и/или сервера – обладателя роли глобального каталога;
- выход из строя всех почтовых серверов;
- выход из строя всех серверов компании в каком-либо филиале.

Список можно продолжать долго. Польза от подобных сценариев очевидна. Продумывая сценарии заранее, составляя план восстановления, вы прорабатываете в деталях каждую мелочь. Кроме того, чем масштабнее разрушения, тем дороже обойдутся меры по восстановлению. Рассмотрев заранее различные сценарии, вы сможете предметно объяснить руководству, сколько будет стоить восстановление в том или ином случае. А уж оно пусть само решает, насколько высока вероятность возникновения подобной ситуации и насколько это критично для бизнеса. Но не забудьте попросить ваших руководителей рассчитать, сколько стоит час простоя вашей организации. Иногда цифры говорят сами за себя.

И еще... Мало однажды определить все возможные сценарии. Мало разработать и составить детальное описание.

Необходимо регулярно эти сценарии проигрывать. Например, можно на постоянной основе моделировать ситуацию с выходом почтового сервера из строя. Это позволит быть уверенным в том, что вы:

- все предусмотрели;
- ваших навыков достаточно для быстрого восстановления;
- изменения, происходящие в вашей ИТ-инфраструктуре, не привели к тому, что ваш план стал непригоден.

Однажды мне рассказывали, как одна крупная организация отрабатывала сценарий «Выход из строя всех серверов компании в филиале». Трудно поверить, но сотрудники компании выезжали в новый офис, разворачивали резервные серверы, проводили необходимые настройки и тестировали удалённое подключение сотрудников компании к серверам. Здорово, правда?

Тем более, с развитием виртуальных технологий вам совсем не обязательно рушить «боевые» серверы. Вы вполне можете иметь у себя виртуальную лабораторию, в которой смоделируете вашу реальную инфраструктуру, и делайте все, что захотите. Останавливайте серверы, разрушайте хранилища – и пробуйте все вернуть обратно.

7. Дистрибутивы и устройства чтения. Необходимо заранее позаботиться о дистрибутивах, которые будут использоваться для восстановления сервера. Сделайте резервные копии ваших установочных дисков и храните их вместе с планом восстановления. Не забудьте о драйверах устройств. Тогда вы не попадете в ситуацию, когда у вас под рукой не окажется нужного вам диска или он будет нечитаемым.

Что же касается устройств чтения, то я часто оказывался в ситуации, когда устройства чтения, будь то CD-ROM или FDD, были забиты пылью и не только не читали информацию с носителей, но еще и портили их.

8. Буфферный склад. Готовясь к восстановлению, позаботьтесь о том, чтобы у вас был запасной, подменный фонд. Например, это могут быть отдельные запчасти или сервер целиком. Главное, чтобы в критический момент не оказалось, что вам необходимо заказывать вышедшее из строя оборудование со склада производителя.

С кем согласовать?

После составления плана его необходимо согласовать во всех инстанциях. Согласования плана с руководством компании позволит вам разделить ответственность в случае допущения ошибки и обеспечит дополнительную проверку документа на случай, если вы что-то упустили.

Обязательно нужно согласовать план с администраторами сети и операторами архивов.

Как и где хранить?

План восстановления обязательно нужно хранить в печатном виде, т.к. в критический момент вы можете не вспомнить, куда же запихнули этот файл или, что еще хуже, сам файл может оказаться на том самом сервере, который вышел из строя.

Распечатайте, подпишите у всех, с кем необходимо согласовать план, и поместите его в несгораемый шкаф. Главное, чтобы это было доступное для всего ИТ-персонала место.

Электронную копию также нужно хранить в надежном месте. На тот случай, чтобы можно было внести изменения.

На этом теоретическая часть окончена. Я надеюсь, что статья заставила вас задуматься над тем, насколько хорошо документирована ваша ИТ-инфраструктура. Насколько вы готовы к неприятным неожиданностям в виде выхода серверов из строя.

***

В следующих статьях мы поговорим о практике восстановления Microsoft Exchange Server. Я наглядно продемонстрирую процесс восстановления различных версий – от 2003 до 2010. Также я планирую подробно описать те сценарии восстановления, о которых упоминал в данной статье. От восстановления письма до восстановления кластера серверов. И обязательно уделю особое внимание восстановлению и обслуживанию базы данных. Именно база данных, в которой хранится вся почта организации (а зачастую не только почта, но и важные документы), доставляет больше всего хлопот системным администраторам. Поэтому очень важно знать, как проводить ремонт базы данных. Как делать дефрагментацию. Какие утилиты для этого использовать и т.д.

Несмотря на то, что я делаю акцент на восстановлении именно Exchange Server, используя ту теоретическую базу, которая дана в этой статье, вы сможете применить ее для того, чтобы составить план восстановления любых серверов, которые используются в вашей организации. EOF

Визитка

АЛЕКСЕЙ БЕРЕЖНОЙ, системный администратор. Главные направления деятельности: виртуализация и гетерогенные сети. Еще одно увлечение помимо написания статей – популяризация бесплатного ПО

Расширение возможностей

при работе с сетевыми хранилищами NETGEAR ReadyNAS

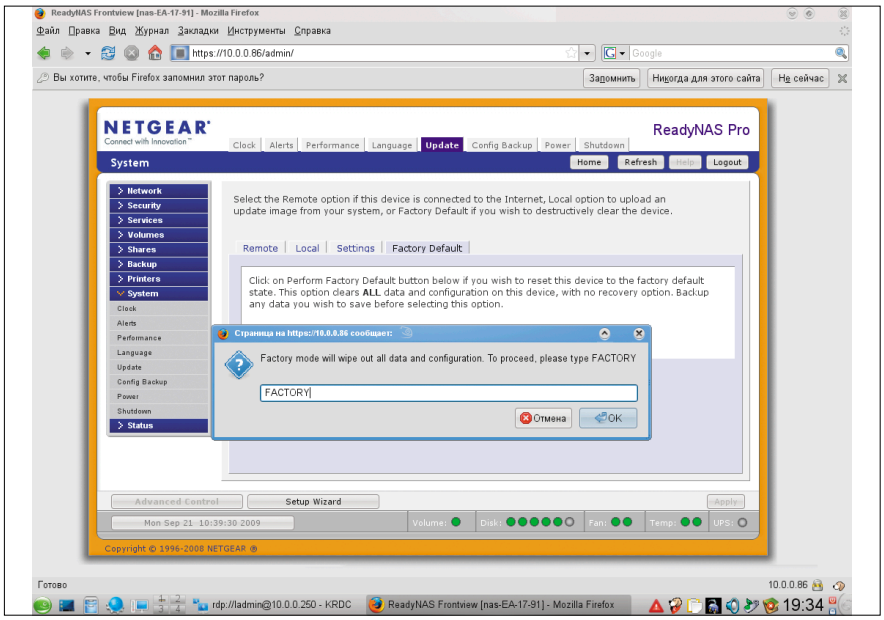
Не только приобрести и подключить сетевое хранилище, а попытаться добиться от него максимальной функциональности — вот профессиональный подход

В предыдущих статьях [1, 2] было рассказано о первоначальной установке и настройке дискового хранилища, а также об использовании резервного копирования информации как с рабочих станций и других серверов, так и непосредственно на самом дисковом хранилище. Методы организации работы, описанные в этих статьях, основывались в первую очередь на настройках по умолчанию и на техфункциях, которые предоставляются веб-интерфейсом и, что называется, «плавают на поверхности».

В этот раз мы поговорим о более тонких сущностях, таких как недокументированные возможности, тюнинг работы дискового хранилища, интересные нюансы и т.д.

Перед тем как начать работу, установите ваше хранилище на устойчивую горизонтальную поверхность в помещении с соответствующим температурным режимом. Температурный диапазон для NETGEAR ReadyNas Pro – от 0 до 40 градусов по Цельсию. Желательно, чтобы у вас был свободный доступ к хранилищу, чтобы можно было прочесть сообщения на индикаторе лицевой панели.

Рисунок 1. Окно подтверждения запуска процедуры возврата к заводским настройкам



Важное замечание! Во время выполнения описываемых ниже функций, таких как обновление системы, установка обновлений или возврат к заводским настройкам, ни в коем случае нельзя выключать или перезагружать сетевое хранилище, кроме тех случаев, когда этого требует система. В противном случае вы можете получить полностью неработоспособное устройство. Поэтому запаситесь надежным блоком бесперебойного питания (UPS) и терпением в ожидании нормального завершения операции.

Возврат к заводским настройкам (Perform Factory Default)

Возможность ознакомиться на практике с данной функцией выпадает довольно редко. Причиной тому являются два фактора.

Во-первых, сетевое хранилище ReadyNAS Pro – хорошо продуманная система для хранения данных, практически не требующая «доработки напильником». Если в только что купленном хранилище уже установлено некоторое количество дисков, сразу же при старте системы запустится процесс создания X-RAID-массива с дисками установленного объема. По окончании данного этапа хранилище в принципе готово к работе с настройками по умолчанию.

Во-вторых, при возврате к заводским настройкам удаляется вся (!) информация как с дискового массива, так и касающаяся пользовательских настроек, например, пароля администра-

тора, учетных записей пользователей, вручную присвоенных IP-адресов и т.д.

Применение данной функции имеет смысл в следующих случаях:

- выполняется первоначальная настройка сетевого хранилища, что называется, с нуля;
- в связи с ошибочно проведенными настройками и невозможностью их отмены обычным путем требуется вернуть хранилище к рабочему состоянию пусть даже ценой потери информации и всех настроек;
- возникла необходимость быстро уничтожить всю информацию на дисковом массиве;
- производится смена типа RAID-массива.

В качестве подготовительного этапа необходимо хотя бы на одном компьютере сети установить программу RAIDar для обнаружения, мониторинга и управления сетевыми хранилищами ReadyNAS. И крайне желательно иметь в сети поднятый хотя бы на время сервер DHCP. Следует заметить, что программа RAIDar в Linux-редакции устанавливается далеко не на все дистрибутивы Linux. Например, мне так и не удалось установить ее на свой любимый Linux openSUSE 11.1 Поэтому очень рекомендую на этом этапе иметь в сети хотя бы один компьютер с Microsoft Windows XP, чтобы установить программу RAIDar.

Несмотря на все грозные предупреждения, проходит данный процесс довольно оперативно. Для его запуска необходимо запустить веб-интерфейс, введя в браузере строку https://_IP_ или_hostname/admin выбрать режим Advanced Control, раскрыть пункт меню System → Update и перейти на вкладку Factory Default, далее необходимо просто нажать кнопку Perform Factory Default. После активации указанной кнопки система выдаст окно с сообщением: Factory mode will wipe out all date and configuration. To proceed, please type FACTORY (переход к заводским настройкам уничтожит все данные и конфигурацию, для продолжения напечатайте FACTORY) (см. рис 1).

Чтобы подтвердить всю серьезность своих намерений, набираем указанное слово в поле ввода и нажимаем OK. Далее сетевое хранилище попросит перезагрузки.

Вот тут начинается самое интересное. При первом старте после возврата к заводским установкам сетевое хранилище в течении 10 минут сканирует сеть с целью найти запущенную программу RAIDar. Для взаимного обнаружения дисковых хранилищ и RAIDar используется специальный протокол, разработанный компанией NETGEAR. Сам механизм обнаружения основан на широковещательных посылках и имеет некоторые ограничения. Например, при использовании VLAN необходимо, чтобы компьютер с запущенным RAIDar был в том же VLAN, что и искомое сетевое хранилище. После ее удачного обнаружения можно воспользоваться кнопкой Setup программы RAIDar. Чтобы вызвать диалог, сетевое хранилище сообщает свои данные RAIDar, и на экране компьютера с запущенной программой появляется окно создания RAID-массива: Welcome to the ReadyNAS Volume Setup (см. рис. 2).

Рисунок 2. Окно выбора разновидности RAID-массива

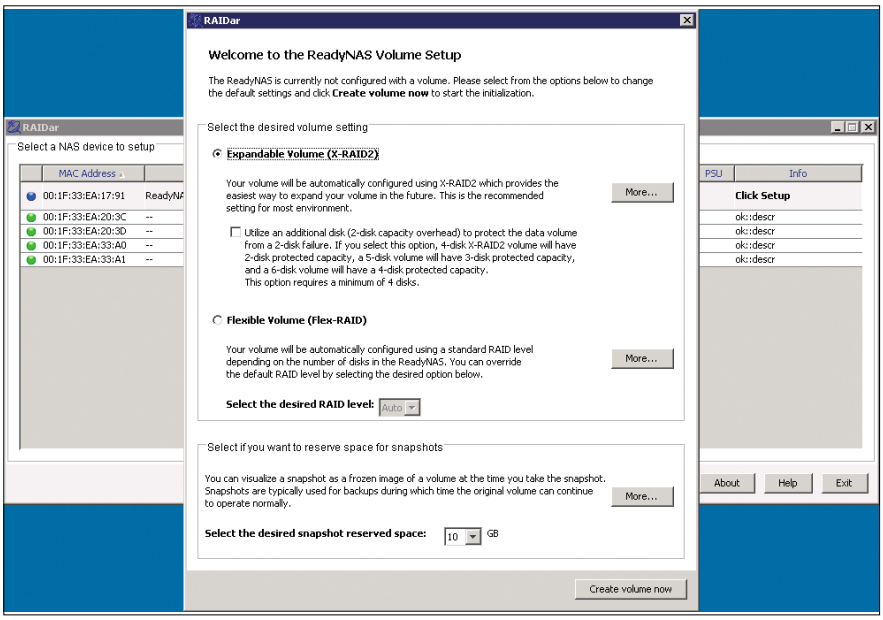
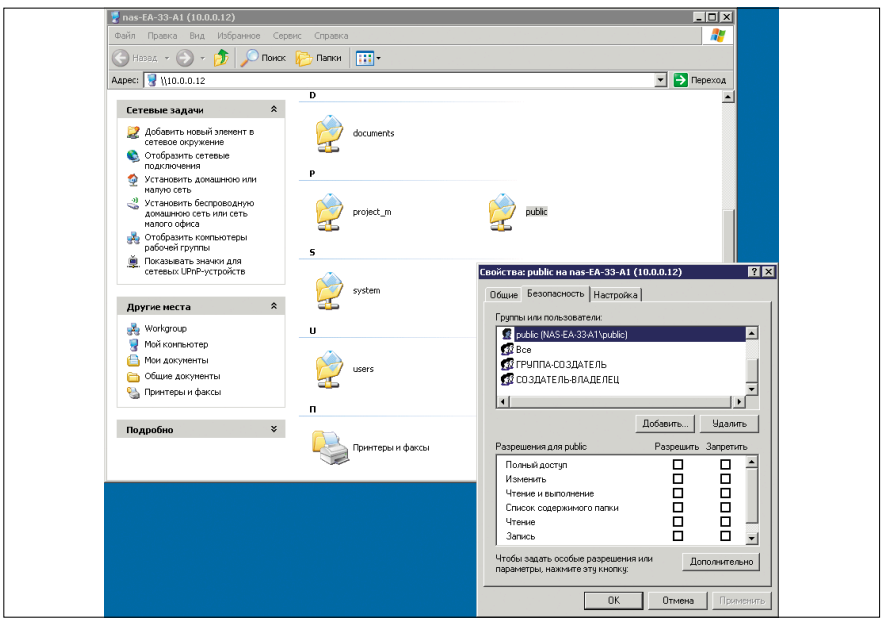


Рисунок 3. Расширенное назначение прав на сетевом каталоге, используя Windows Explorer



В сетевых хранилищах NETGEAR ReadyNAS Pro предлагается два типа RAID-массивов.

Expandable Volume (X-RAID2) – позволяет на лету увеличивать объем дискового пространства, подключая новые диски к сетевому хранилищу, а также по очереди заменив жесткие диски на другие, большего объема.

Flexible Volume (Flex-RAID) – основан на «традиционных» технологиях RAID со всеми их плюсами и минусами. Можно собрать RAID-0, RAID-1, RAID-5, RAID-6 или даже несколько массивов (не более 4) с условием, что на одном физическом диске будет не больше 2 разных массивов (томов).

Более подробную информацию по этому вопросу можно найти в [1].

Лично я предпочитаю X-RAID2. Flex-RAID хорошо подходит для непродолжительных операций чтения – записи, но для изменения объема данного типа массива требуется полная перестройка. Думаю, со мной согласится большинство системных администраторов, которые стремятся к установке сервера по принципу «раз – и надолго». (Не говоря уже о случаях, когда требуется работоспособная система в режиме 24x7.)

После нажатия кнопки Create Volume now начнется создание RAID-массива. В принципе ничего сверхъестественного. Для 5 дисков по 1 Гб RAID-массив процесс занимает около 10 часов.

Обновление системы

Компания NETGEAR время от времени выпускает обновления своих продуктов, в том числе и для ReadyNAS Pro. Наверное, нет особой нужды объяснять, что обновление позволяет повысить отказоустойчивость и безопасность системы.

Для выполнения обновлений используется меню System → Update.

Сама операция крайне проста. Можно запустить обновление в автоматическом режиме (вкладка Remote), тогда прошивка будет скачана автоматически. Минусом в этом случае является необходимость открыть на файерволе порт 80 для исходящих соединений. К сожалению, автоматическое обновление через Интернет не предполагает использование прокси-сервера. В качестве альтернативного варианта можно заранее скачать свежую версию прошивки, она называется RAIDiator (на момент написания статьи была актуальна версия 4.2.5 выполнить обновление.

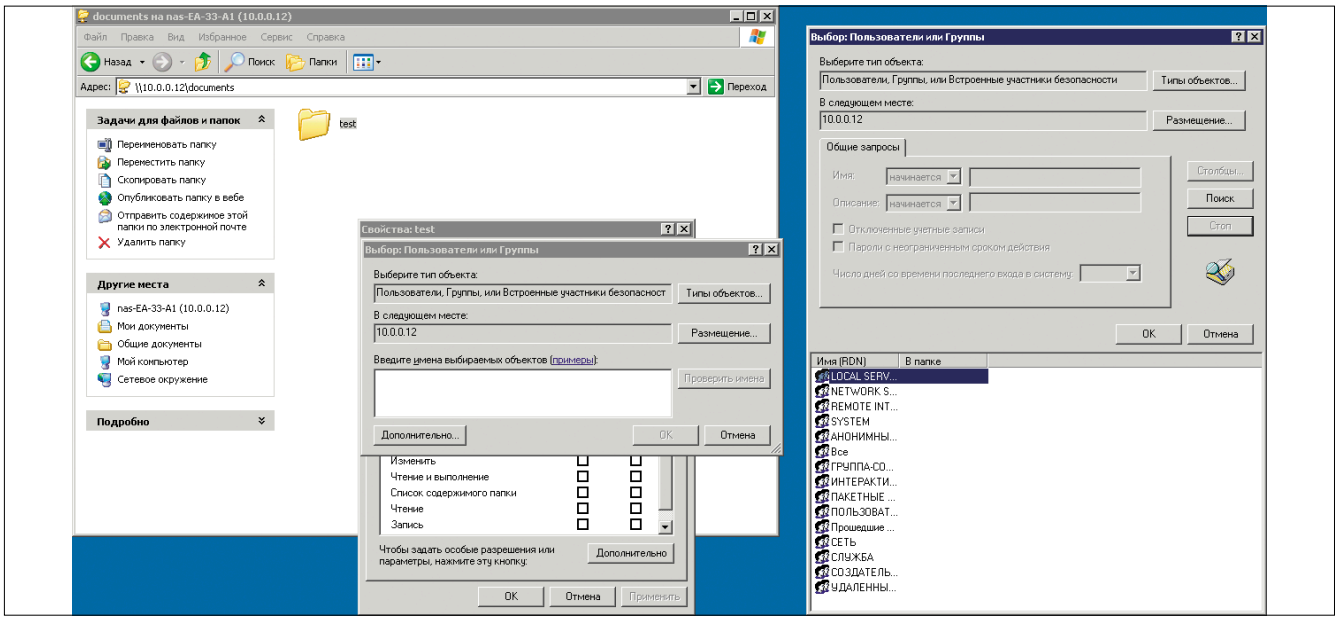
Я лентяй, и поэтому просто открыв все порты на шлюзе для исходящих соединений, «выпустил» на короткое время свое сетевое хранилище в Интернет и запустил обновление в автоматическом режиме. Обновление проходит на ура, поэтому особых поводов для беспокойства нет, правда, не стоит забывать о необходимости обеспечить режим бесперебойного питания. В конце обновления вас попросят перезагрузить хранилище. Ну вот, собственно, и все. После завершения операции настройки файервола можно вернуть в первоначальное состояние.

Чтобы не отставать от жизни, в этом же меню существует вкладка Settings, где вы можете определить как возможность периодической проверки новых обновлений, так и автоматической загрузки обновлений в хранилище. (Лично я не планирую использовать вторую функцию, все равно при установке обновленной версии системы требуется перезагрузка.)

Эксклюзивное назначение прав доступа

Посредством FrontView (встроенный веб-интерфейс управления хранилищем) можно в большинстве случаев полностью решить вопрос назначения прав доступа к файлам. Сложность заключается в том, что установленные атрибуты распространяются на весь общий раздел, так как задействован механизм наследования разрешений. А как быть в том случае, если необходимо раздать эксклюзивные права на каждый каталог? Например, если сетевое хранилище служит для размещения личных каталогов пользователей?

Рисунок 4. «Пропадание» пользователей и групп из списка доступных объектов



На наше счастье NETGEAR ReadyNAS Pro поддерживает возможность ACL (Access Control List) при организации доступа к файлам, аналогично NTFS. Секрет использования ACL заключается в том, что для использования этого метода права необходимо назначать не посредством FrontView, а прямо из проводника Windows, подключившись к сетевому ресурсу (см. рис. 3).

Казалось бы, все просто. Но существует один нюанс, который следует учитывать. Хранилище работает на nix-платформе, поэтому полноценной поддержки ACL «как у Windows» нет. И это, кстати, синдром большинства NAS'ов не только от NETGEAR.

Если вы захотите отказаться от наследования прав и передать выбранному неголовному каталогу права, отличные от каталога верхнего уровня, возникает ситуация пропадания из списка доступных объектов пользователей и групп, созданных на сетевом хранилище. То есть вы будете видеть только группы, присутствующие по умолчанию в любой Windows системе (см. рис. 4).

Обойти данную проблему можно, если «подняться» на самый верхний уровень сетевого каталога и назначить там права для избранных пользователей и групп (например, только на чтение) и затем, спускаясь по цепочке вниз, снимать наследование и убирать ненужные объекты из списка разрешений. Решение неудобное с точки зрения повседневной эксплуатации, зато работает.

Несколько рекомендаций для облегчения жизни системного администратора в случае, если NETGEAR ReadyNAS Pro используется в качестве файлового сервера.

Создавайте общие ресурсы под конкретные задачи. Допустим, необходимо создать раздел для хранения личных папок, раздел для организации с документами с определенными правами и некий общий публичный раздел с доступом для всех в режиме чтение – запись. В этом случае имеет смысл использовать минимум три общих раздела: один (Users) для личных каталогов, второй (Documents) для документов работы с документами, на который будут розданы особые права с учетом групп пользователей, и третий каталог (Public) для публичного доступа. В этом случае, если меняются права у подкаталога в разделе Documents, не придется менять права для папок, находящихся в Users и Public (см. рис. 5).

Создавайте специальные группы для каждого случая. Допустим, необходимо ограничить доступ к подкаталогу Folder_1 на общем ресурсе Documents. В этом случае имеет смысл создать две группы: только для чтения и для чтения – записи соответственно. Теперь если необходимо какому-либо пользователю открыть доступ к Folder_1, его нужно просто добавить в соответствующую группу, не занимаясь постоянным переназначением прав сверху вниз, начиная с корневого каталога.

Иногда вместо того чтобы раздавать сложные права на файлы и каталоги, гораздо проще переместить их в соответствующий раздел. Например, потребовалось для файла, находящегося в каталоге Documents, дать права на чтение – запись для всех сотрудников. Не легче ли просто переместить его в Public?

В принципе эти простые приемы годятся для использования не только в случае с сетевыми хранилищами от NETGEAR, но и для любого другого файл-сервера. Надеюсь, они помогут.

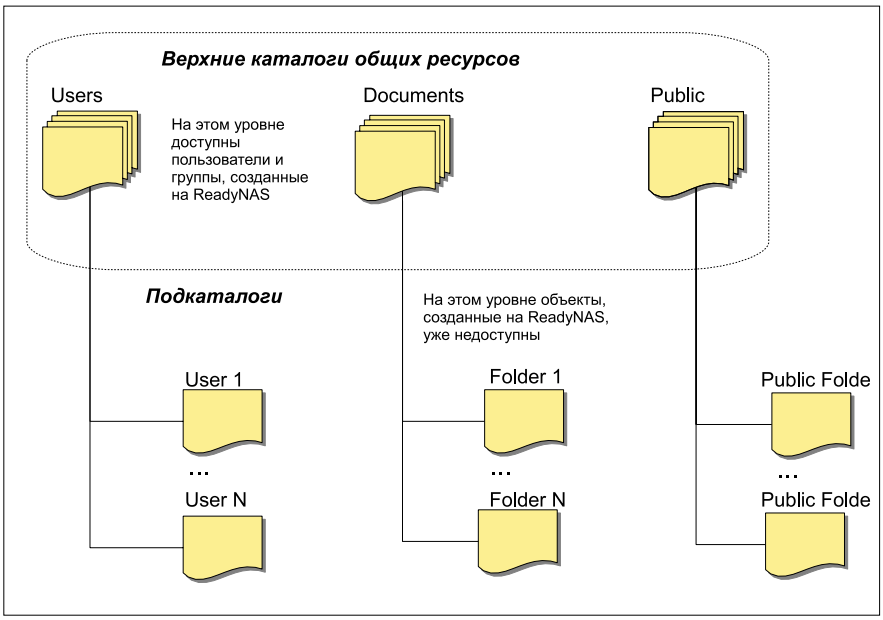
***

Использование специализированных сетевых хранилищ в качестве файл-серверов не только упрощает работу: не нужно устанавливать операционную систему и драйвера, ПО для мониторинга системы и защищать операционную систему файл-сервера от вирусной атаки, но и накладывает некоторые особенности при работе. Но инженерный подход и грамотное планирование способны значительно минимизировать трудовые затраты на обслуживание.

В следующей статье мы коснемся установки расширений (add-on) и дополнительных программ, таких как антивирус. EOF

1. Бережной А. Альтернатива файловому серверу. Это – дисковое хранилище NETGEAR ReadyNAS. //Системный администратор, №7, 2009 г. – С. 14-20.
2. Бережной А. NTI Shadow for ReadyNAS: проводим резервное копирование данных. //Системный администратор, №8, 2009 г. – С. 52-54.
3. Сайт на русском языке, посвященный дисковым хранилищам NETGEAR ReadyNAS – http://www.readynas.su.
4. Официальный сайт ReadyNAS Community – http://www.netgear.com.
5. Сайт международного сообщества ReadyNAS – http://www.readynas.com.
6. Официальный международный сайт NETGEAR – http://www.netgear.com.

Рисунок 5. Пример создания разделенных общих сетевых ресурсов для конкретных задач



Визитка

АНТОН ПИЩУЛИН, главный специалист ОАО «ОТП Банк», занимается разработкой и интеграцией корпоративных информационных систем

Технология Oracle Streams

Настраиваем потоки данных, экономим время и деньги

Режим DownStream технологии Oracle Streams помогает повысить эффективность использования ресурсов в высоконагруженных информационных системах

При эксплуатации автоматических банковских систем (АБС) в условиях высокой нагрузки, а также параллельно существующих подотчетных систем (в том числе и хранилищ данных), которые зависимы от АБС и должны, например, раз в сутки забирать данные из нее, в период пиковой нагрузки может не хватить ресурсов для своевременного выполнения всех бизнес-задач. Причина понятна – на первом месте всегда стоит задача сдачи отчетности, от которой зависит напрямую деятельность банка. Помимо этой проблемы существует задача аудита данных, для которой также нужны ресурсы. Решить задачи помогает технология Oracle Streams. При помощи этой технологии поток данных с центрального сервера АБС оказывается вполне допустимым и не нагружающим систему. Однако всегда присутствует желание снизить эту нагрузку еще.

Дело в том, что классическая архитектура технологии Oracle Streams состоит из трех процессов – захвата изменений (capture), распространения изменений (propagation) и применения изменений (apply). Два первых процесса в этом случае работают на системе-источнике, то есть на стороне АБС, что не всегда допустимо не только из-за ограниченности ресурсов, но и из соображений отказоустойчивости и безопасности. Выходом из положения является режим захвата изменений на стороне-приемнике, режим DownStream. При этом на АБС настраивается только дополнительное логирование для системы-приемника. Процессы захвата и применения изменений работают на системе-приемнике.

Между системой-источником и системой-приемником настраивается лог-транспорт. Таким образом, снимается нагрузка с АБС и повышается эффективность работы всех задач банка.

Итак, для начала работы (демонстрации) нам необходимы два экземпляра Oracle с учебной схемой данных пользователя HR (входит в учебный дистрибутив Oracle по OTN лицензии). В общем случае потоки данных могут быть между базами данных, схемами, таблицами и т.д. На практике чаще всего требуется создать поток из одной схемы в другую или из одного набора таблиц в другой. Рассмотрим поток из одной таблицы в другую. Масштабируя данный подход, можно получить набор возможностей для решения достаточно большого объема задач.

Перед началом настройки данные на сервере-источнике (site1) и сервере-приемнике (site2) должны быть синхронизированы. Вся инсталляция программного обеспечения Oracle была выполнена в каталоги, которые предлагает инсталлятор по умолчанию (на платформе Windows – исключительно для целей обучения). Все скрипты написаны в общем виде на PLSQL так, чтобы их легко можно было масштабировать.

Часть 1. Сервер-источник (site1)

В этой части будем настраивать сервер, на котором будет формироваться поток изменений.

Настройка параметров инициализации экземпляра Oracle

> Настраиваем параметр global_names. Этот параметр накладывает стандарт на название dblink, а именно названия должны совпадать с именем базы данных (или имя базы.домен).

```
global_names = true
```

> Режим archivelog должен быть включен.

> Возможность создания логов на удаленный сервер-приемник должна быть включена:

```
remote_archive_enable = true
```

> Настраиваем лог-транспорт на удаленный сервер-приемник:

```
log_archive_dest_2 = 'SERVICE=site2 noregister ↵
        reopen=300  template=C:\oracle\streams\ ↵
        site1_%t_%s_%r.arc'
```

Папку C:\oracle\streams я создал вручную.

> На всякий случай проверяем параметр (необходим для лог-транспорта):

При помощи Oracle Streams поток данных с центрального сервера АБС оказывается вполне допустимым и не нагружающим систему

```
log_archive_dest_state_2 = 'enable'
```

> Создаем пользователя – администратора streams и даем ему роль dba:

```
create user strmadm identified by strmadm;
grant dba to strmadm;

BEGIN
  DBMS_STREAMS_AUTH.grant_admin_privilege(
    grantee          => 'strmadm',
    grant_privileges => true);
    commit;
END;
```

Настройка реплицируемых объектов (далее все от пользователя strmadm)

Будем настраивать поток из базы site1 объекта hr.employees в базу данных site2.

Все объекты, участвующие в репликации, должны быть специальным образом настроены. На сервере-источнике для идентификации записей, для процессов streams, в журнальные логи должна попадать дополнительная информация (в формате sys.anydata) об операциях над объектами. Эта дополнительная информация будет считана процессом захвата изменений (capture) на сервере-приемнике.

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(20);
  tb tp := tp(
  --Определяем схему и имя реплицируемого объекта
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP
  --Включаем дополнительное логирование на таблицу
  --на основе ключей
  DBMS_CAPTURE_ADM.prepare_table_instantiation(
    table_name          => tb(i),
    supplemental_logging => 'keys');

  END LOOP;

END;
```

Часть 2. Сервер-приемник (site2)

В этой части будем настраивать сервер, который будет принимать поток изменений с сервера-источника.

Настройка параметров инициализации экземпляра Oracle

> Настраиваем параметр global_names. Этот параметр накладывает стандарт на название dblink, а именно названия должны совпадать с именем базы данных (или имя базы. домен).

```
global_names = true
```

> Режим archivelog должен быть включен.

> Формат архивных журнальных логов log_archive_format сделаем таким же, как на системе-источнике с точностью до sid, то есть site2_%t_%s_%r.arc.

> Создаем администратора streams и даем ему роль dba.

```
create user strmadm identified by strmadm;
grant dba to strmadm;

BEGIN
  DBMS_STREAMS_AUTH.grant_admin_privilege(
    grantee          => 'strmadm',
    grant_privileges => true);
    commit;
END;
```

Настройка реплицируемых объектов (далее все от пользователя strmadm)

> Создаем dblink на сервер-источник для автоматизации настройки процесса захвата изменений (capture):

```
CREATE DATABASE LINK site1 CONNECT TO strmadm ↵
       IDENTIFIED BY strmadm
   USING '(DESCRIPTION =
    (ADDRESS_LIST =
     (ADDRESS = (PROTOCOL = TCP)(HOST = localhost) ↵
       (PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = site1)
    )
  )';
```

> Включаем логирование на необходимые таблицы (по аналогии с сервером-источником):

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(20);
  tb tp := tp(
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP
  --Включаем дополнительное логирование на таблицу
  --на основе ключей
  DBMS_CAPTURE_ADM.prepare_table_instantiation(
    table_name          => tb(i),
    supplemental_logging => 'keys');

  END LOOP;

END;
```

> Создаем очереди сообщений. Архитектура Oracle Streams в режиме DownStream устроена следующим образом: процесс захвата изменений (capture) формирует сообщения из журнальных логов, которые порождает сервер-источник (site1), и помещает их в очередь типа sys.anydata, затем процесс применения (apply) сканирует очередь сообщений и применяет изменения на сервере-приемнике (site2).

```
BEGIN
DBMS_STREAMS_ADM.set_up_queue(
   --Таблица очереди сообщений
   queue_table => 'site1_queue_table',
  --Название очереди сообщений
   queue_name  => 'site1_queue'
   );
END;
```

> Создаем процесс захвата изменений (capture):

```
BEGIN
    DBMS_CAPTUR_ADM.create_capture(
   --Название очереди сообщений
    queue_name => 'strmadm.site1_queue',
    --Название процесса захвата
    capture_name => 'capture_site1',
    --Использовать dblink для настройки
    use_database_link => true,
    --База-источник
    source_database => 'site1'
    );
END;
```

> Создаем правила для процесса захвата изменений. Дело в том, что архитектурой Oracle Streams предусмотрено, что каждый процесс должен иметь хотя бы одно правило (условие), при выполнении которого сообщение пропускается дальше или нет. Правила бывают положительные и отрицательные, то есть при срабатывании положительного правила сообщение пропускается, при срабатывании отрицательного – нет. В данной статье используются самые простые положительные правила.

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(20);
  tb tp := tp(
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP

--Создание положительного правила для таблицы
 DBMS_STREAMS_ADM.add_table_rules(
  table_name => tb(i),
  --Тип процесса streams
  streams_type => 'capture',
  --Название процесса streams
  streams_name => 'capture_site1',
  --Очередь сообщений streams
  queue_name => 'strmadm.site1_queue',
  --Включить dml операции
  include_dml => true,
  --Включить ddl операции
  include_ddl => true,
  --База-источник
  source_database => 'site1',
  --Положительное правило
  inclusion_rule => true
  );

 END LOOP;

END;
```

> Добавляем сервисные поля в реплицируемые объекты. В начале статьи упоминалось о задаче аудита данных. Здесь предлагается на уровне записи отслеживать, когда запись была вставлена, обновлена или удалена. Для этого добавляются два сервисных поля – одно типа date (дата транзакции над записью), другое – varchar2 (для отслеживания типа транзакции над записью).

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(20);
  tb tp := tp(
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP

  execute immediate 'alter table '||tb(i)||' ↵
       add status_rec varchar2(1)';
  execute immediate 'alter table '||tb(i)||' ↵
       add status_dt date';

 END LOOP;

END;
```

> Создаем функцию трансформации для процесса применения изменений. Она необходима для заполнения сервисных полей на системе-приемнике, созданных в предыдущем пункте. Функция трансформации в об-

щем случае привязывается к любому правилу, соответственно к любому процессу Streams и является расширением простых правил. В данной статье функция трансформации привязывается к процессу применения изменений (apply), как это будет видно ниже.

```
create or replace function site1_trans ↵
        (evt in sys.anydata)

--Получаем сообщение и отдаем его дальше после
--трансформации
return sys.anydata is
  lcr       sys.lcr$_row_record;
  lcr_val   SYS.LCR$_ROW_LIST;
  obj_name  varchar2(30);
  rc        number;
  cmd_tp    varchar2(10);
  st        varchar2(1);
  dt        date;
begin

  --Если сообщение dml
  if evt.GetTypeName='SYS.LCR$_ROW_RECORD' then
     rc := evt.GetObject(lcr);
     obj_name := lcr.get_object_name();

        --Определяем тип транзакции
        cmd_tp := lcr.get_command_type();
        st := upper(substr(cmd_tp,1,1));

        --Если удаление, то трансформируем в обновление
        if st = 'D' then

          lcr_val := lcr.get_values('OLD');
          lcr.set_command_type('UPDATE');
          lcr.set_values('OLD',lcr_val);

          dt := lcr.get_source_time();
          --Заполняем сервисные поля
          lcr.ADD_COLUMN('NEW','status_dt', ↵
                 sys.anydata.convertdate(dt));
          lcr.ADD_COLUMN('NEW','status_rec', ↵
                 sys.anydata.ConvertVarchar2(st));

          return sys.anydata.convertobject(lcr);
        end if;
        --Заполняем сервисные поля
        lcr.ADD_COLUMN('NEW','status_dt', ↵
                 sys.anydata.convertdate(sysdate));
        lcr.ADD_COLUMN('NEW','status_rec', ↵
                 sys.anydata.ConvertVarchar2(st));


     return sys.anydata.convertobject(lcr);
       end if;

  return evt;
end site1_trans;
```

> Создаем процесс применения изменения (apply) сразу вместе с простыми положительными правилами.

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(50);
  tb tp := tp(
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP

 DBMS_STREAMS_ADM.add_table_rules(
  table_name => tb(i),
  --Тип процесса streams
  streams_type => 'apply',
  --Название процесса streams
  streams_name => 'apply_site1',
  --Очередь сообщений streams
  queue_name => 'strmadm.site1_queue',
  --Включить dml операции
  include_dml => TRUE,
  --Включить ddl операции
  include_ddl => true,
  --База-источник
  source_database => 'site1',
  --Положительное правило
  inclusion_rule => true
  );

 END LOOP;

END;
```

Ниже приведен код, устанавливающий вручную scn (снапшот), с которого процесс захвата (capture) начинает отслеживать изменения на сервере-источнике. Данный код не обязателен, так как данная операция производится автоматически при выполнении кода выше. Однако на практике иногда возникает необходимость выполнить это вручную.

```
DECLARE

TYPE tp IS TABLE OF VARCHAR2(50);
  tb tp := tp(
  'hr.employees'
  );

BEGIN

  FOR i IN tb.FIRST .. tb.LAST
  LOOP

    DECLARE
     iscn  NUMBER;          --scn
    BEGIN
     --Получаем scn базы сервера-источника
     iscn := ↵
        DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER@site1;

     --Устанавливаем scn на таблицу сервера-приемника
     DBMS_APPLY_ADM.set_table_instantiation_scn(
       source_object_name    => tb(i),
       source_database_name  => 'site1',
       instantiation_scn     => iscn);
    END;

  END LOOP;

END;
```

> Настраиваем функции трансформации. Привязываем функции трансформации к правилам процесса применения изменений (apply).

```
DECLARE

--Осуществляем выборку искомых правил
CURSOR c1 (s1 varchar2) IS
select c.rule_name from dba_streams_rules c where ↵
       c.streams_type='APPLY'
and c.rule_type = 'DML';

ss1 varchar2(200);

BEGIN

  OPEN c1(ss1);
  LOOP
     begin
     FETCH c1 INTO ss1;
     EXIT WHEN c1%NOTFOUND;
     --Привязываем существующую функцию трансформации
     --к правилам
     DBMS_STREAMS_ADM.set_rule_transform_function( ↵
        ss1,'site1_trans');

  exception when others then null;
     end;
  END LOOP;

END;
```

Часть 3. Тестирование Oracle Streams

Сейчас мы готовы протестировать всю проделанную нами работу. Для начала необходимо стартовать созданные нами процессы.

```
--Старт процессов захвата и применения изменений
BEGIN
  dbms_apply_adm.start_apply('apply_site1');
  dbms_capture_adm.start_capture('capture_site1');
END;
```

> На сервере-источнике (site1) выполним следующую транзакцию типа update.

```
update hr.employees t set t.first_name='anton' where ↵
       t.employee_id=198;
commit;
--Попереключаем несколько раз журнальные логи,
--так как мы быстрее хотим дождаться результата теста
alter system switch logfile;
```

Через некоторое время мы должны увидеть на сервере-приемнике следующее.

```
select t.first_name,t.status_rec,t.status_dt from
hr.employees t where t.employee_id=198;

anton           U        01.11.2009 20:16:25
(в вашем случае дата будет другая)
```

> На сервере-источнике (site1) выполним следующую транзакцию типа delete.

```
delete from hr.employees t where t.employee_id=199;
commit;
--Попереключаем несколько раз журнальные логи
alter system switch logfile;
```

Через некоторое время мы должны увидеть на сервере-приемнике следующее.

```
select t.first_name,t.status_rec,t.status_dt from
hr.employees t where t.employee_id=199;

Douglas D        02.11.2009 8:33:23
(в вашем случае дата будет другая)
```

> На сервере-источнике (site1) выполним следующую транзакцию типа insert.

```
insert into hr.employees
select 207, e.first_name, e.last_name, ↵
       'letanton@mail.ru', e.phone_number, ↵
       e.hire_date, e.job_id, e.salary, ↵
       e.commission_pct, e.manager_id, e.department_id
from hr.employees e where e.employee_id=198;
commit;
--Попереключаем несколько раз журнальные логи
alter system switch logfile;
```

Через некоторое время мы должны увидеть на сервере-приемнике следующее.

```
select t.first_name,t.status_rec,t.email,t.status_dt
from hr.employees t where t.employee_id=207;

anton   I    letanton@mail.ru   02.11.2009 8:43:5
(в вашем случае дата будет другая)
```

Вот и все, теперь еще пара слов об основных системных представлениях, необходимых для мониторинга нашего потока:

```
--Просмотр состояния процесса захвата изменений
Select * from dba_capture
--Просмотр состояния процесса применения изменений
select * from dba_apply;
--Детализация ошибок процесса применения
select * from dba_apply_error;

--Мониторинг работы процесса захвата изменений
   SELECT c.CAPTURE_NAME,
       SUBSTR(s.PROGRAM,INSTR(s.PROGRAM,'(')+1,4) ↵
              PROCESS_NAME,
       c.SID,
       c.SERIAL#,
       c.STATE,
       c.TOTAL_MESSAGES_CAPTURED,
       c.TOTAL_MESSAGES_ENQUEUED
  FROM V$STREAMS_CAPTURE c, V$SESSION s
  WHERE c.SID = s.SID AND
        c.SERIAL# = s.SERIAL#;

--Мониторинг работы процесса применения изменений
SELECT r.APPLY_NAME,
       DECODE(ap.APPLY_CAPTURED,
              'YES','Captured LCRS',
              'NO','User-enqueued messages','UNKNOWN') ↵
                     APPLY_CAPTURED,
       SUBSTR(s.PROGRAM,INSTR(s.PROGRAM,'(')+1,4) ↵
              PROCESS_NAME,
       r.STATE,
       r.TOTAL_MESSAGES_DEQUEUED,
       R.TOTAL_MESSAGES_SPILLED
       FROM V$STREAMS_APPLY_READER r, V$SESSION s, ↵
              DBA_APPLY ap
       WHERE r.SID = s.SID AND
             r.SERIAL# = s.SERIAL# AND
             r.APPLY_NAME = ap.APPLY_NAME;
```

За более подробной информацией необходимо обратиться к оригинальной документации Oracle [1]. EOF

1. http://download.oracle.com/docs/cd/B19306_01/server.102/b14228/mon_rep.htm#i1012369.

Решите проблемы **лицензирования ПО** с помощью профессионалов!

Операционная система GNU/Linux и свободное программное обеспечение помогут вам с минимальными затратами решить проблему лицензирования программного обеспечения, повысить безопасность и надежность вашей компьютерной сети.

Компания ГНУ/Линуксцентр предлагает вам внедрение ОС GNU/Linux и свободного программного обеспечения, реализацию и техническую поддержку сложных технических решений на базе свободного ПО, обучение ваших сотрудников — как пользователей, так и технических специалистов.

С НАШЕЙ ПОМОЩЬЮ ВЫ СМОЖЕТЕ:

- оптимизировать затраты на лицензирование ПО за счет максимально возможного использования свободного ПО;
- существенно сократить время системных администраторов, затрачиваемое на устранение последствий деятельности вирусов и сбоев в программном обеспечении.

ТИПОВЫЕ ПРОЕКТЫ:

- миграция рабочих станций и серверов с Microsoft Windows на GNU/Linux;
- установка 1С на серверах и рабочих станциях под управлением GNU/Linux;
- миграция с Microsoft Windows Active Directory на Mandriva Directory Server;
- миграция с Microsoft Exchange на Zimbra;
- внедрение интернет-телефонии на базе Asterisk;
- внедрение свободной CRM-системы SugarCRM;
- создание кластеров высокой доступности;
- реализация терминальных решений;
- создание порталов любой сложности на базе свободных CMS-систем — Joomla, Drupal, Plone;
- внедрение защищенных систем на основе сертифицированного ФСТЭК ПО.

СРЕДИ НАШИХ КЛИЕНТОВ:

- Правительство Московской области;
- Правительство Нижегородской области;
- администрация Черниговского района Приморского края;
- Министерство финансов республики Саха (Якутия);
- Владивостокский государственный университет экономики и сервиса;
- группа компаний «ИМАГ»;
- компания «Азбука мебели»;
- компания «Бестли — выставочные материалы»

и другие организации различного профиля.

Наш опыт внедрения свободного программного обеспечения в компаниях различного профиля поможет выбрать оптимальное сочетание свободного и коммерческого программного обеспечения, подходящее именно для вашей организации, а также поможет избежать техничеких и организационных проблем при внедрении свободного ПО.

Linux center www.linuxcenter.ru

Департамент внедрений компании ГНУ/Линуксцентр
Телефон в Москве: (499) 271-49-54,
в Санкт-Петербурге: (812) 309-06-86

ЗВОНИТЕ СЕЙЧАС!

Реклама

Визитка

АНДРЕЙ БИРЮКОВ, специалист по информационной безопасности. Работает в крупном системном интеграторе. Занимается внедрением решений по защите корпоративных ресурсов

WDS поможет

Установка операционных систем. Часть 1

Разворачивание операционных систем является одной из задач любого системного администратора. Посмотрим, какие средства для этих целей предлагает Windows Server 2008

Различные системы клонирования

Развертывание операционных систем на большое количество рабочих станций представляет собой долгий, утомительный и однообразный процесс. Ведь необходимо установить множество идентичных операционных систем на рабочие станции. Кроме того, в крупных компаниях рабочие станции, как правило, размещаются в нескольких офисах. Территориальное размещение добавляет трудностей при развертывании операционных систем на рабочие станции. Так как администратор не может произвести установку ОС на рабочую станцию удаленно, и ему необходимо производить установку с консоли. Существует много различных систем клонирования, призванных максимально автоматизировать этот процесс. Системы клонирования или создания образов могут быть полезными при развертывании серии идентичных настольных компьютеров или серверов. Программное обеспечение на «чистый» компьютер можно клонировать с помощью различных средств. Наиболее распространенным из них, на мой взгляд, является Acronis True Image, за ним следует Systems Management Server. Однако тут сразу следует отметить, что это отдельные программные продукты, требующие дополнительных финансовых затрат на покупку лицензий, развертывание и обучение персонала. Также данные продукты требуют использования дополнительных аппаратных мощностей, так как им необходимы отдельные сервера и большое количество свободного дискового пространства.

Между тем серверная операционная система Windows Server, начиная с версии 2000, обладает собственными средствами клонирования образов операционных систем. В Windows Server 2008 для этого используется служба Windows Deployment Services (WDS). В первой части своей статьи я рассмотрю установку WDS и создание загрузочных и установочных образов. Следует отметить, что WDS входящая в состав Windows Server не требует дополнительного лицензирования, и работает непосредственно на сервере. Так что этой службе не нужен дополнительный сервер, а что касается дискового пространства, то оно требуется только для хранения образов разворачиваемых систем.

WDS поддерживает клонирование и создание образов серверов и рабочих станций, только если для генерирования идентификаторов защиты (Security Identifiers, SID) нового компьютера используется утилита Sysprep. Для развертывания «чистых» инсталляционных образов, а также для специально настроенных или захваченных инсталляционных образов на серверах и настольных компьютерах, работающих под управлением Windows, может использоваться служба Windows Deployment Services, входящая в состав Windows Server 2008.

WDS и ее предшественники

Служба Windows Deployment Services является новым средством развертывания операционных систем семейства Windows, пришедшая на смену службе RIS (Remote Installation Services). Служба удаленной установки RIS была выпущена вместе с операционной системой Windows 2000 Server и являлась первой успешной службой развертывания операционной системы, которая могла работать в сети. Служба в Windows 2000 Server не поддерживала развертывание серверной операционной системы и имела много ограничений, хотя и представляла собой очень функциональное и ценное средство. Для развертывания серверных операционных систем была разработана служба автономного развертывания Automated Deployment Services, которая представляла собой дополнение к операционной системе Windows Server 2003 Enterprise Edition и была предназначена для помощи в быстром развертывании только серверных операционных систем Windows 2000/2003. Если организации требовалось развертывание как серверных, так и клиентских операционных систем, приходилось использовать и Remote Installation, и Automated Deployment Services.

С выходом операционной системы Windows Server 2003 Service Pack 2 администраторы могли модифицировать свои системы Windows Server 2003 RIS до Windows Server 2003 Windows Deployment Services. После нескольких простых действий по конфигурированию существующие образы RIS могли успешно использоваться WDS.

Новый WDS в Windows Server 2008

В новой операционной системе Windows Server 2008 служба WDS предлагает много тех же функций, что и комбинация RIS, Automated Deployment Services и Windows Server 2003 SP2. Кроме того, Windows 2008 WDS предлагает дополнительные функции, которых не было ни в одной из ее предшественниц.

Рассмотрим более подробно эти нововведения:

- Поддержка образов операционных систем Windows Vista и Windows 2008.
- Возможность развертывать образы с помощью широковещательной передачи, когда клиент посылает широковещательный запрос, а сервер WDS на него отвечает.
- Возможность развертывать образы на клиентах с помощью автономного сервера, на котором работает отдельный компонент WDS, называемый службой роли Transport.
- Возможность использовать образы загрузки и установки, содержащиеся на инсталляционных носителях Windows Vista и Windows Server 2008, с использованием расширения .wim. Их можно копировать с носителя прямо на сервер Windows, обеспечивая доступ к базовым образам этих операционных систем за считанные минуты, без какой-либо настройки.
- Поддержка развертывания операционных систем на 32- и 64-разрядных платформах.

Таким образом, Windows Server 2008 WDS представляет собой роль сервера, который использует в своей работе службу каталога Active Directory Domain Services при развертывании Windows.

Предварительные требования

Как уже упоминалось выше, для полноценного функционирования Windows Deployment Services организации необходимо предварительно развернуть Active Directory Domain Services. Также из-за особенностей среды Preboot Execution Environment (PXE, предзагрузочная среда выполнения) сервер WDS требует использования протокола DHCP для динамической выдачи IP-адресов.

Кроме того, не стоит забывать, что для нормального развертывания образов компьютер, на который производится установка, должен поддерживать загрузку PXE и иметь как минимум 512 Мб оперативной памяти, поскольку это минимальный объем памяти, необходимый для инсталляции операционных систем Windows Server 2008 и Windows Vista.

Служба WDS содержится в редакциях Standard и Enterprise Windows Server 2008.

Типы образов WDS

Windows 2008 WDS работает с несколькими типами образов:

Загрузочный образ (boot image) – содержит клиент Windows Deployment Services и среду Windows PE (Windows Preinstallation Environment – предустановочная среда Windows), которая, по сути, является операционной системой в миниатюре, предназначенной для соединения системы с сервером WDS и предоставления средств для выбора и установки инсталляционного образа WDS. Загрузочный образ, имеющийся на инсталляционном носителе Windows 2008, имеет соответствующее ему имя boot.wim и может использоваться для загрузки систем, которые будут устанавливать образы Windows Vista, Windows 2008. Загрузочный образ Windows 2008 может применяться также для инсталляции образов с помощью широковещательных передач.

Инсталляционный образ (installation image) – является установочным носителем Windows, который упакован в одном файле. В зависимости от самого носителя он может содержать множество различных установочных образов. Например, если ваша организация получила лицензионный DVD-диск Microsoft Windows Server 2008, то он может содержать установочные образы операционных систем Standard, Enterprise, Standard Core и Enterprise Core Edition. На сервере WDS обычно требуется только один загрузочный образ, однако он может содержать множество различных установочных образов.

Образ обнаружения (discovery image) – создается из загрузочного образа и используется для запуска системы и загрузки среды Windows Preinstallation Environment, а также для нахождения сервера WDS и соединения с ним. Образ обнаружения обычно используется, если сеть или система не поддерживает загрузку PXE. Образы обнаружения могут храниться на переносимых носителях, таких как компакт-диски или карты памяти USB.

Образ захвата (capture image) – тоже создается из загрузочного образа, который вместо выполнения настройки или запуска клиента WDS запускает утилиту захвата WDS. Утилита WDS используется для связи с системой, готовой для создания образа или клонирования, и создает новый установочный образ, который можно загрузить на сервер WDS. Прежде чем использовать образ захвата, производится настройка системы с ОС посредством добавления других приложений, специальных конфигураций и других изменений в системе, необходимых для определенной организации. Когда система будет готова к созданию образа, она конфигурируется с помощью утилиты Sysprep. С помощью этой утилиты очищается идентификатор безопасности SID компьютера и конфигураций операционной системы, являющихся специфическими для системы, на основе которой будет создан образ.

Установка WDS

На этом закончим с теорией и перейдем к практике, а именно к установке службы Windows Deployment Service. WDS 2008 можно инсталлировать несколькими различными способами: с помощью консоли диспетчера Server Manager, мастера Initial Configuration Wizard (мастер первоначальной конфигурации), утилиты командной строки Servermanagercmd.exe. Мы будем устанавливать WDS с помощью консоли диспетчера Server Manager.

Перед началом установки необходимо проверить, все ли диски на сервере имеют формат NTFS. Также необходимо использовать отдельный диск для хранения образов WDS.

Приступим к установке. Так как Windows Server 2008 для многих является пока еще незнакомым продуктом, я рассмотрю процесс установки подробно.

Для развертывания WDS вам необходимо выполнять все описанные ниже действия под учетной записью доменного администратора.

- В меню Start выберите пункт All Programs → Administrative Tools → Server Manager.
- В окне диспетчера Server Manager в панели с древовидным представлением выберите узел Roles.
- В панели задач щелкните на ссылке Add Roles.
- На открывшейся странице Before You Begin щелкните по кнопке Next.
- На странице с выбором ролей сервера Select Server Roles отметьте флажок рядом с ролью Windows Deployment Services и щелкните на кнопке Next (см. рис. 1).
- В открывшемся окне снова нажмите кнопку Next.
- На странице Select Role Services проверьте, отмечены ли флажки Deployment Server (сервер развертывания) и Transport Server (сервер транспорта) и щелкните по кнопке Next (см. рис. 2). Здесь следует отметить, что при необходимости сервер транспорта может быть развернут отдельно на другом компьютере, при условии, что в сети доступен хотя бы один сервер развертывания.
- На странице Confirm Installation Selections просмотрите выбранный вариант и щелкните на кнопке Install.
- По окончании установки на странице Installation Results нажмите Close, чтобы завершить процесс установки.

Настройка

После успешной установки перейдем к настройке службы WDS. Настройку службы Windows Deployment Services можно осуществлять одним из следующих способов:

- с помощью графического интерфейса ОС Windows;
- с помощью инструмента командной строки WDSUTIL.

Графический интерфейс более удобен при интерактивной работе с WDS, так как нет необходимости использовать команды со сложным синтаксисом. Командная строка позволяет использовать командные файлы и сценарии для развертывания.

Настроим службу Windows Deployment Services с помощью графического интерфейса. Для этого нам необходимо выполнить следующие действия:

- В меню Start раскройте папку Administrative Tools и щелкните значок Windows Deployment Services.
- В дереве консоли оснастки Windows Deployment Services раскройте список серверов Servers.
- Как видно на рис. 3, при первом запуске сервер WDS в панели с древовидным представлением имеет значок восклицательного знака. Это означает, что служба требует настройки. Правой кнопкой мыши щелкните на имени сервера и в контекстном меню выберите команду Configure Server.
- На странице Remote Installation Folder Location (местонахождение папки удаленной установки) укажите путь для инсталляции по умолчанию для образов WDS. Например, в статье будет использован отдельный привод и путь e:\RemoteInstall. В случае, если сервер имеет один диск и этот диск используется для папки инсталляции, на экран будет выведено предупреждение с рекомендацией создать диск для инсталляции на другом томе (см. рис. 4).
- На странице PXE Server Install Settings (первичные настройки сервера PXE) имеются следующие опции для настроек загрузки PXE. Рассмотрим каждую из этих опций подробнее.

  Do Not Respond to Any Client Computer – эта опция запрещает какое-либо использование загрузки PXE.

  Respond Only to Known Client Computers (отвечать только известным клиентским компьютерам) – эта опция требует, чтобы для каждой машины, на которой будет развернут или захвачен образ, имелась существующая учетная запись компьютера в Active Directory с заранее определенным GUID.

  Respond to All (Known and Unknown) Client Computers – эта опция позволяет любому компьютеру, на котором может быть произведена загрузка PXE, соединиться с сервером WDS и получать загрузочный образ.

  For Unknown Clients, Notify Administrator and

Рисунок 1. Установка WDS

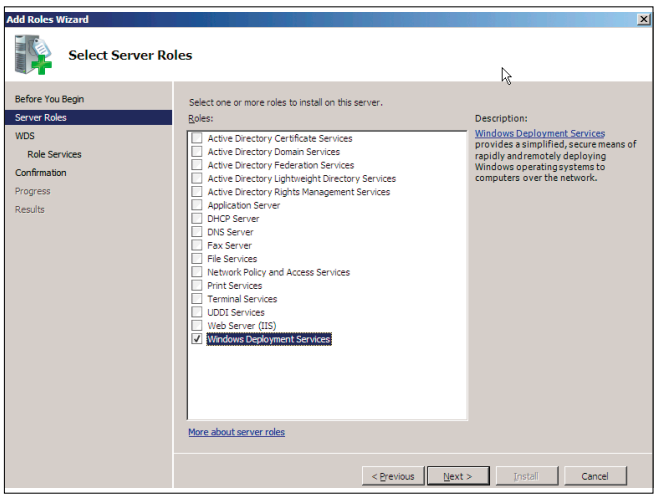
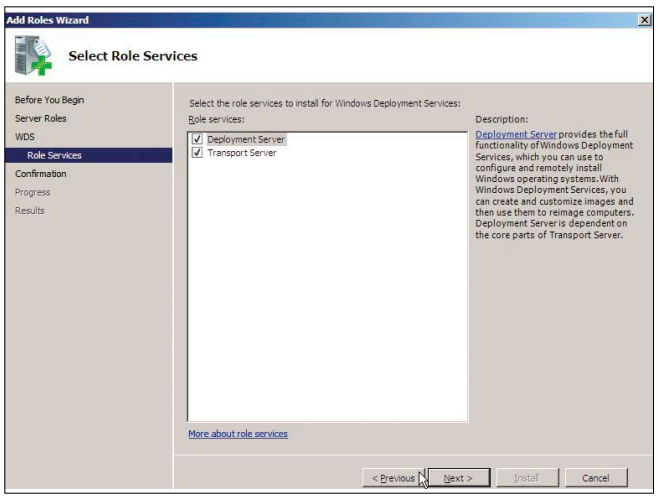


Рисунок 2. Роли сервера WDS



Respond After Approval (для неизвестных клиентов уведомлять администратора и отвечать после его разрешения) – благодаря этой дополнительной защите снимается необходимость в коллекционировании GUID-идентификаторов системы до того, как система сможет соединиться с WDS, и добавляется необходимая степень защиты, которая позволяет администратору управлять образами WDS-развертывания.

- Для простоты инсталляции WDS выберите опцию Respond to All (Known and Unknown) Client Computers (см. рис. 5).
- По завершении установки снимите флажок Add images to Windows Deployment Services now и нажмите кнопку Finish.

После того как работа мастера будет завершена, на экране появится консоль Windows Deployment Services. Теперь в консоли отображается ряд узлов:

- WDS Install Images (установочные образы);
- Boot Images (загрузочные образы);
- Legacy Images (унаследованные образы);
- Pending Devices Multicast Transmissions (широковещательные передачи).

Настройка из командной строки с помощью WDSUTIL

Если по каким-либо причинам вам удобнее устанавливать WDS с помощью командной строки, то для настройки служб Windows Deployment Services выполните следующие действия:

- В меню Start щелкните правой кнопкой мыши на значок Command Prompt и в контекстном меню выберите пункт Run as administrator (если вы не используете административные привилегии).
- Выполните следующую команду:

```
WDSUTIL /initialize-server ↵
        /reminst:"<буква_диска>\<имя_папки>"
```

где <буква_диска> – буква тома, отформатированного в файловой системе NTFS, а <имя_каталога> – имя ка-

Рисунок 3. Первый запуск WDS

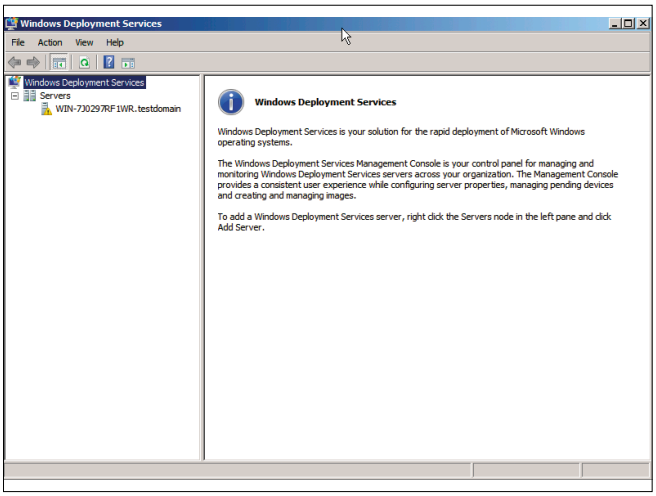


Рисунок 4. Предупреждение системы

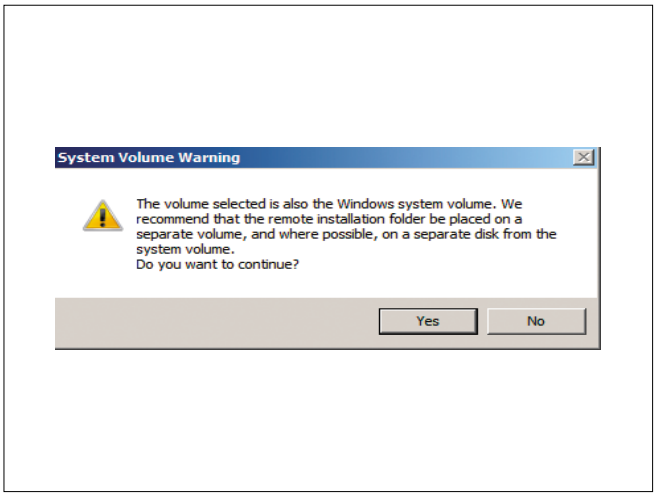


Рисунок 5. Опции PXE

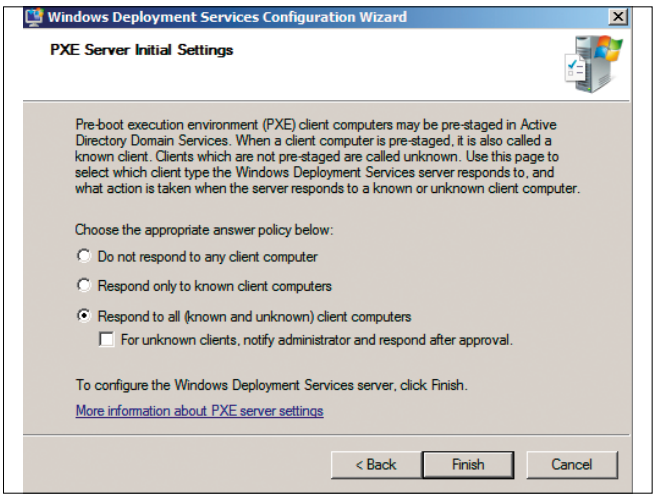
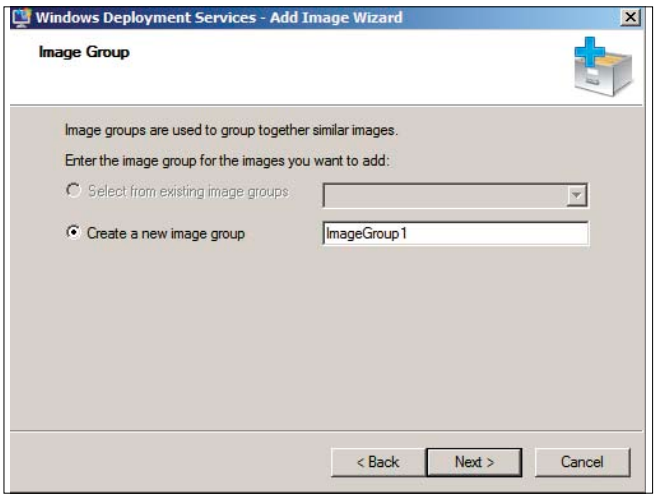


Рисунок 6. Имя загрузочного образа



талога, в котором вы хотите организовать общую папку.

> Чтобы настроить политику ответов для всех клиентов, выполните следующую команду:

```
WDSUTIL /Set-Server /AnswerClients:all
```

> Если вы устанавливаете службы Windows Deployment Services на компьютер, на котором установлен DHCP–сервер Microsoft, выполните команду:

```
WDSUTIL /Set-Server /UseDHCPPorts:no /DHCPoption60:yes
```

> Чтобы добавить загрузочный образ, выполните следующую команду:

```
WDSUTIL /Add-Image /ImageFile:<загрузочный_образ> ↵
        /ImageType:boot
```

где <загрузочный_образ> – полный путь к загрузочному образу.

> Чтобы добавить установочный образ, выполните следующую команду:

```
WDSUTIL /add-image /ImageFile:<установочный_образ> ↵
        /ImageType:install /ImageGroup:<имя_группы>
```

где <установочный_образ> – полный путь к установочному образу, а <имя_группы> – имя создаваемый группы образов.

Вы можете добавить параметр /SingleImage:<имя_образа>, чтобы добавить только один из образов, включенный в файл install.wim.

В случае если вам необходима дополнительная информация, воспользуйтесь справкой по использованию инструмента WDSUTIL, набрав в командной строке:

```
WDSUTIL /?
```

Добавляем загрузочный образ

Для развертывания образов WDS у вас должен быть рабочий сервер DHCP с активной областью действия в сети. Он используется для того, чтобы предоставить клиентскому компьютеру PXE-адрес в формате Ipv4 во время инсталляции образа. Когда происходит установка WDS, сервер PXE автоматически регистрирует его в Active Directory, позволив всем клиентам PXE находить сервер WDS, не прибегая к изменению каких-либо опций DHCP.

В случае если службы WDS и DHCP расположены на одном сервере, то в свойствах WDS необходимо открыть вкладку DHCP, где установить опции: Do not listen on port 67 и Configure DHCP option 60 to ‘PXEClient’. Это необходимо для правильного функционирования служб WDS и DHCP на одном сервере.

Теперь необходимо добавить загрузочный образ на сервер WDS. Так как мы предполагаем, что все Windows-серверы, которые мы будем в дальнейшем разворачивать в нашей сети, должны работать под управлением Windows Server 2008, думаю, в первом примере мы создадим образ именно этой операционной системы. Если система загружается с помощью PXE и соединяется с сервером WDS, то загрузочный образ используется для подготовки клиентской системы к инсталляции Windows. Загрузочный образ содержит клиент WDS и Windows PE. В устаревших системах создания образов применялись драйверы сети на основе DOS для загрузки системы и соединения ее с сервером образов. Теперь же, когда имеется загрузочный образ по умолчанию Windows 2008 (boot.wim), Windows PE содержит большой список драйверов сети, поэтому большинство систем можно загружать в WDS и успешно инсталлировать образ. Загрузочные образы используются также для создания образов захвата и образов обнаружения.

Поэтому начнем с добавления образа, который находится на установочном DVD-диске ОС Windows Server 2008. Для этого выполните следующие действия:

> Раскройте узел с именем сервера, на который вы хотите добавить загрузочный образ. Если локальный сервер не будет показан в узле Servers, его нужно добавить.

> Правой кнопкой мыши щелкните на узле Boot Images и в контекстном меню выберите команду Add Boot Image.

> Когда будет открыто окно мастера Add Image Wizard, на первой странице будет показано местонахождение файла загрузочного образа.

Рисунок 7. Список доступных образов

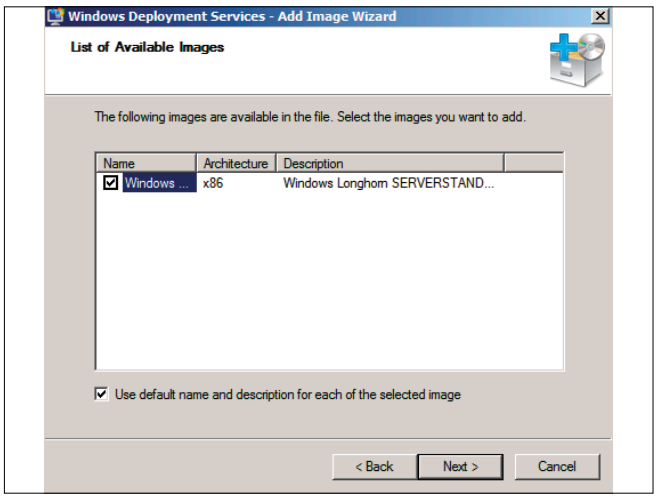
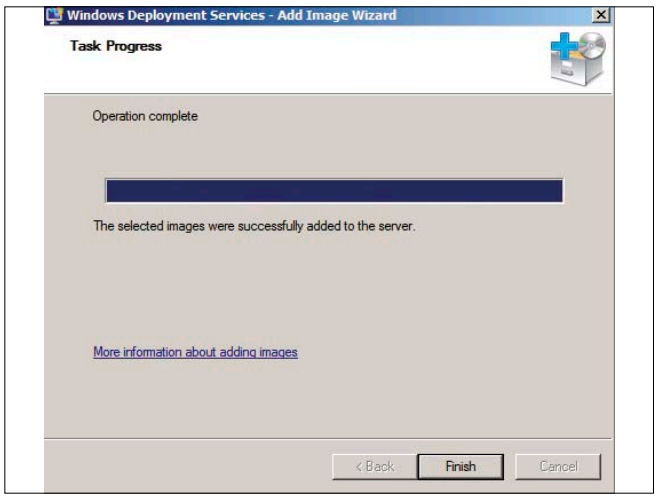


Рисунок 8. Добавление установочного образа



- Нажмите Browse и укажите расположение инсталляционного носителя. Далее откройте папку Sources в корне инсталляционного носителя Windows 2008, выберите файл boot.wim и щелкните по кнопке Open, чтобы добавить его.
- В окне мастера Add Image Wizard щелкните на кнопке Next.
- На странице Metadata вам необходимо указать имя загрузочного образа и нажать Next (см. рис. 6).
- Просмотрите итоговую страницу с результатами настроек и, если все правильно, нажмите Next.
- По завершении процесса создания нажмите Finish.
- В консоли Windows Deployment Services выберите узел Boot Images и проверьте, добавлен ли новый загрузочный образ.

Добавляем установочный образ

После того, как на сервер WDS был добавлен загрузочный образ, можно приступать к добавлению инсталляционных образов. На установочных носителях Windows 2008 и Windows Vista имеется совместимый файл формата Windows Image (WIM). Эти WIM файлы могут быть напрямую добавлены на сервер WDS в качестве установочных образов.

Для добавления одного или нескольких установочных образов по умолчанию, расположенных на установочном DVD-диске Windows, выполните следующие действия:

- Раскройте узел с именем сервера, на который вы хотите добавить образ.
- Правой кнопкой мыши щелкните на узле Install Images и в контекстном меню выберите команду Add Install Image.
- Задайте имя для группы образов и нажмите кнопку Next.
- Выберите установочный образ по умолчанию (install.wim), расположенный на установочном диске Windows Vista или Windows Server 2008 в папке Sources, нажмите кнопку Open, а затем нажмите кнопку Next (см. рис. 7).
- Для добавления образов, включенных в файл install.wim, установите флажки напротив тех из них, которые вы хотите добавить на сервер. Следует добавлять только те образы, для которых у вас имеются лицензии и лицензионные ключи.
- На странице Summary просмотрите список инсталляционных образов, которые будут загружены на сервер WDS, и нажмите Next.
- По завершении нажмите Finish (см. рис. 8).

***

Итак, мы рассмотрели установку сервера WDS, а также создание загрузочных и установочных образов для Windows Deployment Services. Во второй части статьи поговорим о развертывании установочного образа, а также о создании специальных образов захвата, позволяющих использовать образы систем с более гибкими настройками. EOF

1. Microsoft Windows Server 2008. Полное руководство.

Визитка

СЕРГЕЙ ЯРЕМЧУК, фрилансер. Автор более 800 статей и 4 книг. С «СА» с первого номера. Интересы: сетевые технологии, защита информации, свободные ОС

Собираем свой дистрибутив
с Calculate Linux Scratch

Практически каждый пользователь Linux хоть раз да собирал свой дистрибутив. Разработчики Calculate Linux предлагают свой вариант

Необходимость иметь свой вариант системы для системного администратора, да и обычного пользователя может быть продиктована многими факторами. Среди главных – удобство развертывания, когда в устанавливаемой системе присутствуют все необходимые приложения, последние версии ядра и системных библиотек, модули локализации, драйвера и так далее.

В Microsoft, кстати, это тоже отлично понимают, и корпорация предлагает средства пересборки системы – пакет автоматической установки Windows (Windows Automated Installation Kit) [1] и Microsoft Deployment Toolkit.

Конечно же, Linux развивается несколько иным путем, дистрибутивы выходят гораздо чаще Windows и собираются по другому принципу. Хотя не все релизы считаются стабильными и рекомендуются разработчиками к промышленному применению. А при массовом развертывании на предприятии используются стабильные релизы, рекомендуемые разработчиками, например Ubuntu LTS (Long Term Support).

Очевидно, это одна из причин, по которой в Linux большее распространение получили системы автоматической инсталляции, например, такие как – Kickstart. При помощи подготовленного Kickstart-файла можно установить дистрибутив с заданными параметрами и набором приложений. Изначально Kickstart появился в RedHat, и сегодня используется его многочисленными клонами и некоторыми другими дистрибутивами вроде Ubuntu. Пересборка системы производится в основном энтузиастами и для собственных нужд. А в инструментарии большинства дистрибутивов мы не найдем удобных средств для этого.

Здесь можно отметить Ubuntu Customization Kit (UCK) [2] и rBuilder, средство сборки дистрибутива rPath [3]. Конечно, есть Linux From Scratch, который хотя и пользуется некоторой популярностью, но вряд ли может послужить средством массового применения.

Основой Calculate Linux [4], о котором уже говорилось на страницах журнала [5], послужил Gentoo. До недавнего времени развивались две основные ветки – серверная CDS (Calculate Directory Server) и для настольных систем CLD (Calculate Linux Desktop). Последняя была представлена в двух вариантах с разными рабочими столами: CLD это KDE 4.x и CLDX – XFCE 4.6.x.

Недавно в семействе Calculate Linux появился совершенно новый вариант Calculate Linux Scratch (CLS) – представляющий собой LiveCD, предназначенный для самостоятельной сборки системы под любые задачи. Идея, в общем, проста – вместо готовых сборок, в которых часто нет необходимого конкретному пользователю софта, дать удобный инструмент позволяющий создать нужное решение самостоятельно.

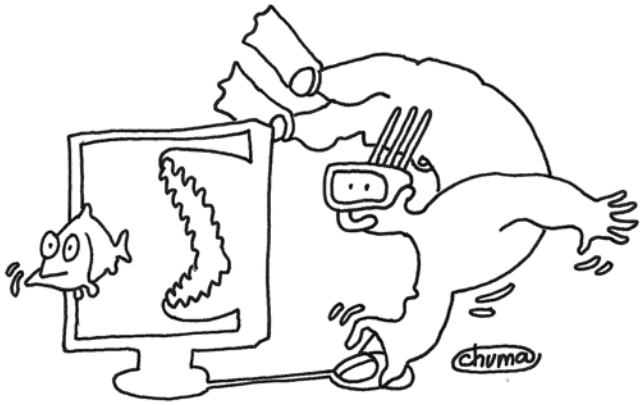
История появления CLS проста. Вначале была собрана информация о том, что не устраивает пользователей Gentoo в других бинарных дистрибутивах, построенных на Gentoo, в частности Calculate и Sabayon. Выяснились две причины: отсутствие нужного софта и отсутствие сборки Calculate Linux с рабочим столом GNOME.

В итоге оптимальный дистрибутив, устраивающий всех, должен был содержать только те программы и библиотеки, которые будут установлены в большинстве случаев – XOrg, Firefox, библиотеки, системные утилиты, драйвера. Так, собственно, и выглядит CLS.

Первая CLS-версия под номером 9.8 вышла в августе 2009 года. Главная особенность – появление режима интерактивной сборки системы, позволяющего внести изменения в состав дистрибутива. Причем сразу стало доступно два варианта: CLS и CLSG.

Версия CLSG содержит облегченную среду Gnome (gnome-base/gnome-light), менеджер сеансов GDM и Wicd для графической настройки проводных и Wi-Fi-сетей (всего приблизительно на 80 пакетов больше CLS). Дистрибутив выпущен под две архитектуры i686 и x86_64.

Первоначально CLS ориентировался именно на стороннего пользователя, сами же разработчики не планировали использовать его для сборки Calculate Linux. Но результат заставил пересмотреть это мнение, и теперь CLS является основной для всех будущих версий Calculate Linux, как серверных, так и настольных. Ведь CLS может выступать при сборке в роли stage3, содержит portage и базовый



Идея проста — вместо готовых сборок дать удобный инструмент, позволяющий создать нужное решение самостоятельно

софт, а все что отличается в USE-флагах (например, в KDE-версии) при необходимости пересобирается, но таких пакетов достаточно мало.

Так, если при стандартной сборке Gentoo (и CLS) команда «emerge -e system» выполняется дважды, то далее уже в этом нет необходимости, а значит, экономятся время и ресурсы. Кроме того, учитывая достаточно небольшой состав приложений, для самостоятельной сборки CLS не требуется больших ресурсов. Продолжая ежемесячный цикл выпуска новых версий, последние два релиза дистрибутива (август и сентябрь) были именно CLS/CLSG.

В середине сентября появился следующий релиз CLS 9.9, в котором:

- перешли на файловую систему Aufs2;
- сохранена 100-процентная совместимость с Gentoo (по сути, Calculate уже практически является частью Gentoo, оверлей помещен в layman, ebuild calculate2 планируется перенести в портежи);
- используется ядро gentoo-sources;
- поддерживается установка на USB-Flash и на HDD/USB-HDD (с файловой системой: ext4, ext3, ext2, reiserfs, xfs и jfs);
- появилась новая утилита cl-unmask, предназначенная для размаскировки пакетов;
- обновлены X.Org 7.4, Kernel 2.6.30.5, Firefox 3.5.3, Gnome 2.24.1 (в CLSG).

Размер образа:

- CLS 9.9: i686 – 573 Мб, x86_64 – 629 Мб;
- CLSG 9.9: i686 – 651 Мб, x86_64 – 709 Мб.

Собираем дистрибутив в интерактивном режиме

Чтобы воспользоваться режимом интерактивной сборки, следует установить дистрибутив на жесткий диск с параметром --build либо выбрать режим загрузки Builder в загрузочном меню LiveCD. В этом случае файловая система после загрузки системы будет состоять из трех слоев aufs2 (Advanced Multi Layered Unification File System, представляет собой усовершенствованную реализацию файловой системы Unionfs 2):

calculate – первый слой, являющийся образом системы, загружаемой с носителя (livecd.squashfs), и смонтированный в режиме «только для чтения». Это основа будущего дистрибутива, которую можно менять;
delta – слой, в котором сохраняются все изменения, произведенные во время сборки;
workspace – рабочий слой, в котором производятся все манипуляции по изменению исходной системы.

Все указанные слои являются подкаталогами в /mnt/scratch и автоматически образуются после загрузки системы.

```
# mount | grep /mnt/scratch
tmpfs on /mnt/scratch/delta type tmpfs (rw,relatime)
tmpfs on /mnt/scratch/workspace type tmpfs (rw,relatime)
/dev/loop0 on /mnt/scratch/calculate type squashfs
(ro,relatime)
```

Собственно, по наличию каталога /mnt/scratch и определяется текущий режим работы системы (в нашем случае Builder).

Переходим в режим интерактивной сборки, введя в консоли cl-builder, после выполнения команды приглашение изменит цвет. Вся дальнейшая работа в текущей виртуальной консоли будет производиться в chroot-окружении /mnt/builder. Теперь можно работать как обычно в Gentoo – обновить портежи, устанавливать, обновлять программы. Здесь необходимо сделать важное замечание. В обычном режиме слои монтируются в таком порядке: calculate + delta + workspace, в интерактивном – calculate + delta.

После установки каждого пакета (завершения выполнения emerge) скрипт cl-builder выполняет команду «mount -o remount», перемонтируя заново слой delta. В итоге все изменения в delta сразу же становятся доступными в основной системе (например, появляются ярлыки приложений) и их можно протестировать перед сборкой нового образа. Запуск приложения в chroot-окружении приводит к созданию нежелательных временных файлов, которые затем попадают в образ. Именно поэтому во избежание конфликтов в работе устанавливаемых программ установку и удаление пакетов следует производить только в chroot-окружении че-

рез cl-builder, а тестировать в обычном. Обновляем утилиту calculate.

```
# layman -S; emerge calculate
```

Обновляем портежи (в принципе релизы CLS выходят раз в месяц и этот шаг можно пропустить):

```
# eix-sync
```

И устанавливаем нужное приложение:

```
# emerge weechat
```

По окончании сборки программа будет сразу же доступна в рабочей системе. В слое delta появятся новые файлы и библиотеки. Аналогичным образом добавляем и остальные приложения. По окончании работы выходим из режима сборки командой exit.

Теперь чтобы собрать загрузочный ISO-образ, достаточно смонтировать раздел жесткого диска (если объема ОЗУ достаточно, этот шаг можно пропустить):

```
# mount /dev/sda3 /usr/calculate/share/linux
```

И затем собираем образ командой:

```
# calculate --iso
```

Через некоторое время в каталоге /usr/calculate/share/linux появится ISO-образ с новой системой.

```
# ls /usr/calculate/share/linux/
cls-9.9-i686.iso cls-9.9-i686.iso.DIGESTS
```

Еще важный момент. Если CLS для сборки установлен на флешку (calculate -d /dev/sdX), можно пересобрать оригинальный файл livecd.squashfs, в котором, собственно, и находится система, сохранив изменения. Для этого достаточно ввести команду:

```
# calculate --rebuild
```

Новый файл будет иметь другой порядковый номер сборки. При следующей загрузке будет уже использован новый образ со всеми изменениями.

Следует заметить, что полученный в интерактивном режиме дистрибутив подходит для массового пользователя, но считается не совместимым с Calculate Directory Server (CDS). Чтобы собрать совместимый с CDS вариант, необходимо использовать обычный режим сборки.

Обычный режим сборки

Чтобы использовать обычный режим сборки системы, следует установить Calculate Linux (любую версию CLS/CLSG, CLD, CLDX или CDS) на жесткий диск обычным образом. Затем копируем в каталог /usr/calculate/share/linux установленной системы ISO-образ CLS оригинальный (если нужна оригинальная версия системы, предложенная разработчиками) или созданный нами в интерактивном режиме.

Вначале следует приготовить систему для сборки командой calcalute, запустив ее с параметром:

```
# calculate -c/--configure
```

По умолчанию будет производиться сборка текущей версии системы, то есть если работаем в CLS, то и собираться будет CLS. Чтобы указать другой вариант дистрибутива, добавляем параметр –s/--os=[CLDICLDXICLSICLSGICDS]. И готовим систему к сборке, например CLDX.

```
# layman -S; emerge calculate
# calculate -c -s CLDX

Building Calculate Linux Desktop
Collecting system information
  Language: en_US
  Keymap: en_US
  Timezone: UTC
  Computer name: calculate
  Network devices: eth0 (DHCP)
Hardware
  Machine hardware name: i686
  Quantity processors: 1
  Create directory stages...  done.
…
```

В текущей версии скрипта calculate (1.2.6) не убраны проверки наличия архивов stage3 и portage, которые требовались ранее для сборки системы. Так, если в процессе появится сообщение:

Рисунок 1. В загрузочном меню Calculate Linux Scratch выбираем Build

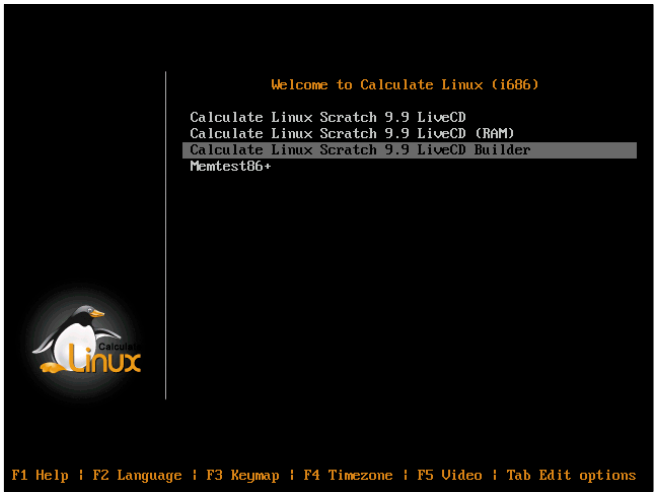
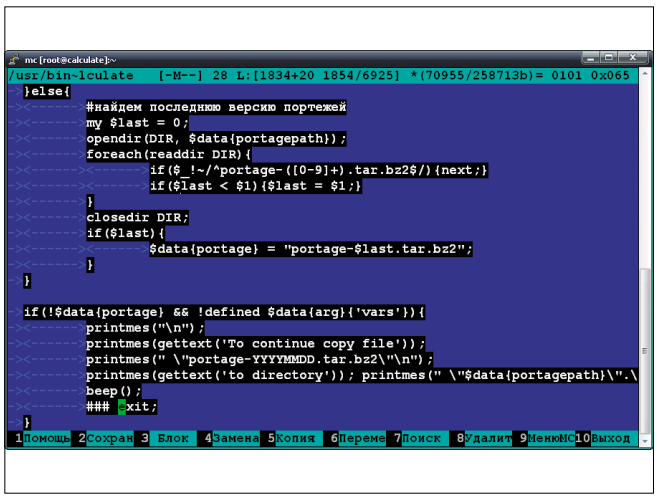


Рисунок 2. В Calculate 1.2.6 необходимо закомментировать строчку



```
To continue copy file "stage3-i686-YYYYMMDD.tar.bz2"
to directory "/usr/calculate/share/stages".
```

Значит в /usr/calculate/share/linux недоступен ISO-образ. Проверяем:

```
# ls /usr/calculate/share/linux/
```

```
cls-9.9-i686.iso
```

Второе возможное сообщение требует архив с portage.

```
To continue copy file "portage-YYYYMMDD.tar.bz2"
to directory "/usr/calculate/share/snapshots".
```

Для всех систем, кроме CDS и CLS, в stage3 и portage нет необходимости, их заменяет сам CLS. Для этих двух систем уже понадобятся последние версии архивов с stage3 и portage. В будущей версии 1.2.8 проверку наличия portage для настольных версий систем обещают убрать (в GIT уже исправления присутствуют). Если это так, можно обновить вручную оверлей Calculate командой eix-sync (в ходе выполнения configure обновляются оверлей и утилита calculate).

Как вариант лучше изменить сам скрипт, закомментировав одну строку в разделе «Определим наличие stage, portage» (строка 1828):

```
# "Определим наличие stage, portage"

if(!$data{portage} && !defined $data{arg}{'vars'}){
printmes("\n");
printmes(gettext('To continue copy file'));
printmes(" \"portage-YYYYMMDD.tar.bz2\"\n");

 printmes(gettext('to directory')); ↵
        printmes(" \"$data{portagepath}\".\n");
beep();
# Комментируем строку с exit
#                       exit;
}
```

После этого подготовка системы к сборке должна пройти без проблем. Если ввести команду «calculate -c» в режиме Builder, получим сообщение:

```
rmdir: failed to remove `/mnt/builder': Device or resource busy
```

Смотрим:

```
# mount | grep /mnt/builder
```

```
none on /mnt/builder type aufs (rw,relatime,si=150eca98)
/usr/calculate/share on /mnt/builder/usr/calculate/share
type none (rw,bind)
none on /mnt/builder/proc type proc (rw)
/dev on /mnt/builder/dev type none (rw,bind)
/dev/pts on /mnt/builder/dev/pts type none (rw,bind)
```

При конфигурировании ISO-образ CLS будет распакован в свободный дисковый раздел (специально, чтобы было легко чистить временные файлы), который автоматически форматируется в ReiserFS и монтируется в /mnt/builder.

```
# mount | grep /mnt/builder
```

```
/dev/sda3 on /mnt/builder type reiserfs (rw)
```

Если скрипт не сможет определить раздел самостоятельно, его следует указать при помощи параметра -d.

В процессе работы команды будет изменен файл /mnt/builder/etc/make.conf в соответствии с настройками выбранной системы, в частности USE-флаги, языковые настройки.

Новое в Calculate Linux

С момента выхода первой статьи в Calculate Linux появилась поддержка «из коробки» DNS- и DHCP-серверов, в качестве реализации выбраны BIND и dhcpd. Соответственно, добавлены и новые утилиты:

Для управления DNS – cl-dns-recadd, cl-dns-recdel, cl-dns-recmod, cl-dns-zoneadd, cl-dns-zonedel, cl-dns-zonemod.

Для управления DHCP – cl-dhcp-hostadd, cl-dhcp-hostdel, cl-dhcp-hostmod, cl-dhcp-netadd, cl-dhcp-netdel, cl-dhcp-netmod.

Установка и управление заданными сервисами осуществляется также просто. Например, чтобы установить DNS, используем команду:

```
cl-setup dns
```

Во время установки DHCP сразу задаются параметры работы при помощи такой команды:

```
cl-setup --router <ip шлюза> --dnames <имена доменов> ↵
         --range <диапазон ip> --net <ip сети с маской /24> ↵
         --dnsip <ip DNS сервера> dhcp
```

Плюс полезная утилита cl-unmask, предназначенная для размаскировки пакетов, которая является альтернативой autoumnask. Формат вывода прост:

```
# cl-unmask package1 package2
```

Файл размаскировки создается для всех версий включая последнюю.

Сравним оригинальный make.conf с CLS и образовавшийся после ввода «calculate -c -s CLDX»:

```
# diff /etc/make.conf /mnt/builder/etc/make.conf
```

```
7c7
< CFLAGS="-O2 -march=i686 -pipe"
---
> CFLAGS="-Os -march=i686 -pipe"
28c28
<   gtk
---
>   gtk -eds -gnome -kde
40c40
< LINGUAS="en ru"
---
> LINGUAS="en de es fr it pl pt_BR ru uk"
42c42
< PKGDIR="/usr/calculate/share/packages/CLS/9.9/i686"
---
> PKGDIR="/usr/calculate/share/packages/CLDX/9.9/i686"
```

После обновления портежей и выполнения команды «emerge sys-apps/portage» скрипт закончит этап подготовки. Состав пакетов, которые будут установлены, можно просмотреть при помощи:

```
# calculate -l
```

Правим при необходимости /mnt/builder/etc/make.conf (man make.conf) и переходим к шагу проверки и размаскировки зависимостей, для чего используем ключ -D/--dependence.

```
# calculate -D
```

По окончании – самый долгий этап – сборка системы при помощи ключа -m/--make:

```
# calculate -m
```

Все новые пакеты пересобираются с флагом --newuse, то, что есть, пропускается. По окончании упаковываем дистрибутив в архив (7z) командой:

```
# calculate -p
```

И создаем ISO-образ:

```
# calculate --iso -s cld
```

Результат сборки системы можно увидеть в каталоге /usr/calculate/share/linux/.

***

В итоге сборка системы при помощи Calculate Linux Scratch выглядит на порядок проще и происходит быстрее, чем сборка того же Gentoo. Путем довольно нехитрых манипуляций можно получить на выходе готовый дистрибутив под свои задачи. EOF

1. Яремчук С. Основные изменения в WAIK для Windows Server 2008 R2/7. //Системный администратор, №4, 2009 г. – С. 16-19.
2. Ubuntu Customization Kit – http://uck.sourceforge.net.
3. rPath rBuilder – http://www.rpath.org.
4. Сайт проекта Calculate Linux – http://www.calculate-linux.ru.
5. Яремчук С. Строим сеть на Calculate Directory Server. //Системный администратор, №8, 2009 г. – С. 16-22.

Рисунок 3. Подготавливаем систему к сборке

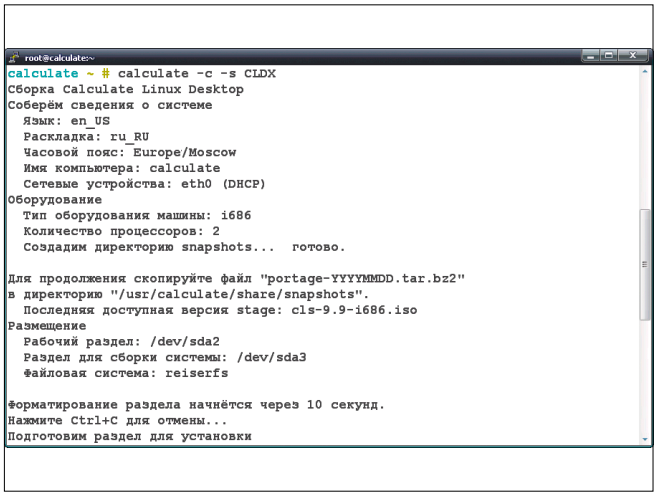
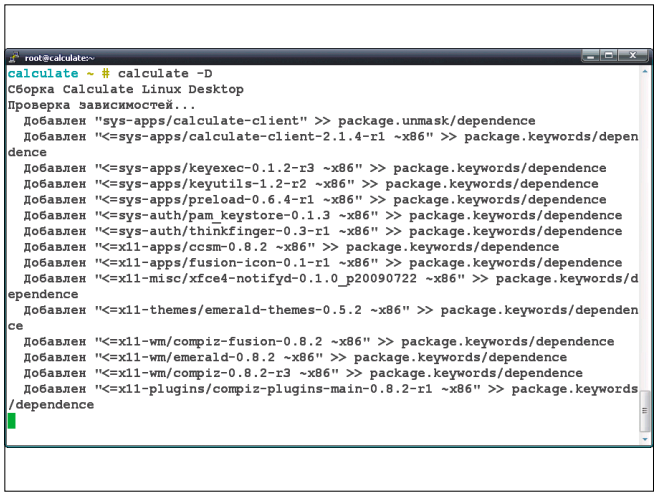


Рисунок 4. Проверка зависимостей



Визитка

АЛЕКСЕЙ БЕРЕЖНОЙ, системный администратор. Главные направления деятельности: виртуализация и гетерогенные сети. Еще одно увлечение помимо написания статей – популяризация бесплатного ПО

Sun VirtualBox

Как персональная система виртуализации

Виртуальная машина на компьютере системного администратора — давно не экзотика. А новый VirtualBox от компании SUN — отличный выбор для создания собственной системы виртуальных машин

Современные решения по виртуализации, несмотря на свое недавнее появление, уже довольно часто используются при решении многих задач. Промышленные системы виртуализации, такие как VMware ESX/ESXi, Citrix XenServer/Essentials, установленные на мощных серверах, способны заменить целый парк серверного оборудования для крупной компании. Подобные системы относятся к типу bare-metal solutions, то есть устанавливаются как, говорится, «на голое железо» и обеспечивают хорошее быстродействие.

В то же время довольно часто необходима доступная система виртуализации под рукой, без приобретения дополнительного оборудования. Например, на ноутбуке системного администратора или на домашнем компьютере. Думаю, не стоит лишний раз рассказывать о преимуществах виртуализации в работе современных ИТ-специалистов. Протестировать то или иное решение, познакомиться с новым программным продуктом, сделать работу над ошибками, чтобы понять причину проблемы и найти способ ее устранения – все это также нашло свое применение в виртуальной среде. Данную систему виртуальных машин в дальнейшем будем называть «персональной системой виртуализации».

Критерии выбора персональной системы виртуализации

Систем виртуализации персонального уровня существует довольно много. Поклонники UNIX-систем, в частности Linux, вспомнят о Xen и OpenVZ, любителям продукции Microsoft придет на память MS Virtual PC, также довольно давно существуют платный продукт VMware Workstation, и бесплатный – VMware Server. Более-менее подробный список систем виртуализации можно найти на Wikipedia (http://ru.wikipedia.org/wiki), набрав в строке поиска: «сравнение_виртуальных_машин».

В то же время персональная система виртуализации должна отвечать определенным критериям, таким как:

Кроссплатформенность. У системы должны быть модификации для наиболее популярных платформ: Windows, Linux, MacOS и Solaris (Open Solaris). Виртуальная машина, созданная в host-системе на одной платформе, должна легко переноситься на host-систему, использующую другую платформу. Сам я использую Linux и иногда Windows XP, поэтому для меня в первую очередь актуальна поддержка именно этих платформ.

Поддержка разнообразных гостевых операционных систем. В отличие от промышленных систем виртуализации, на которых зачастую устанавливается ограниченный перечень операционных систем (например, только MS Windows 2003 и RedHat Enterprise), тестовые системы виртуальных машин нередко используются для ознакомления специалистов с другими платформами.

Возможность переноса гостевых машин с одной host-системы на другую. Должна присутствовать либо возможность прямого переноса файлов виртуальных машин, например, как у VMware Server, либо внятная система экспорта/импорта виртуальных машин, позволяющая переносить виртуальные машины между разными host-системами и, при необходимости, на другую платформу виртуализации.

Простота установки и управления. В данном случае система виртуализации должна требовать минимум усилий для ее изучения. Основные силы и время необходимо отдать задачам, ради которых эта система и устанавливалась: тестированию решений, знакомству с другими платформами, но никак не изучению сложных нюансов самой системы виртуализации.

Поддержка различных методов управления системой виртуализации. Кому-то удобней использовать графический интерфейс, кому-то – интерфейс командной строки. В хорошей системе виртуализации должны быть представлены оба метода. Часто это требование диктуется внешними факторами, например, особенностями организации удаленного доступа: подключение по протоколу RDP в графическом режиме или по telnet/SSH из командной строки.

Наличие хорошей документации. Требование очевидно само по себе.

Быстрота выполнения гостевых операционных систем. Если гостевые операционные системы, развернутые на данной системе, будут безбожно «тормозить», смысл всех вышеперечисленных факторов теряется.

Данная система должна быть бесплатной. Дело даже не в том, что придется заплатить некую сумму денег за программный продукт, пусть даже очень хороший, но который нужен исключительно «по работе». Системные администраторы – люди зачастую весьма занятые. Поэтому лучше потратить время на тестирование новой системы, чем на поиск способа перевести требуемую сумму через какую-либо систему платежа. Например, посредством WebMoney или ехать в банк, чтобы оформлять перевод, а потом еще ждать некоторое количество времени, пока деньги поступят на счет компании-дистрибьютора и в ответ придет лицензионный ключ или коробка с дистрибутивом.

Преимущества VirtualBox перед другими системами

В свое время для экспериментов я использовал VMware Server версии 1.x. Данный продукт устраивал практически по всем вышеперечисленным параметрам. Наличие версии для Windows XP и Linux CentOS в качестве host-систем, простота установки, удобная консоль управления, не очень широкий, но вполне приемлемый перечень поддерживаемых гостевых операционных систем: большинство версий Windows, популярные Linux-дистрибутивы, FreeBSD и DOS. Минусом использования данной версии было присутствие в системе дополнительных процессов, необходимых для работы самой VMware. Для серверного решения это как раз то, что нужно, но для персональной системы виртуализации, когда виртуальная машина запускается время от времени, постоянное наличие дополнительных «пожирателей ресурсов» совсем ни к чему.

Что касается дальнейшего развития продукта, то с выходом версии VMware Server 2.x ситуация изменилась далеко не в лучшую сторону. Во-первых, исчезла поддержка host-систем на базе Windows XP (Только Windows 2003 Server). Во-вторых, удобная быстрая консоль была заменена на довольно неповоротливый веб-интерфейс. Словом, VMware Server 2.x является именно серверной системой виртуализации со всеми вытекающими особенностями. Можно, конечно, продолжать использовать VMware Server версии 1.x. Но все-таки хочется идти в ногу с прогрессом и использовать современные решения по виртуализации.

Во время очередного интернет-серфинга на тему виртуализации мне попалось на глаза упоминание о системе VirtualBox (http://www.virtualbox.org) производства компании Sun Microsystems. Продукция компании Sun всегда отличалась хорошим качеством и продуманностью решений. Именно так обстоит дело с VirtualBox.

Host-системы, поддерживаемые VirtualBox

VirtualBox может быть установлен на большое число операционных систем. Ниже представлен список поддерживаемых ОС в качестве host-системы:

- MS Windows:
  - Windows Server 2003 (32 бит);
  - Windows XP все сервис-паки (32 бит);
  - Windows Vista (32 бит и 64 бит);
  - Windows Server 2008 (32 бит и 64 бит);
  - Windows 7 beta (32 бит и 64 бит).
- Apple Mac OS X hosts:
  - все версии MacOS X на Intel-платформе.
- Linux (32-bit и 64-bit):
  - Debian GNU/Linux 3.1 («Sarge»), 4.0 («Etch») и 5.0 («Lenny»);
  - Fedora Core 4 to 11;
  - Gentoo Linux;
  - Redhat Enterprise Linux 4 и 5;
  - SUSE Linux 9 и 10, OpenSUSE 10.3, 11.0 и 11.1;
  - Ubuntu 6.06 («Dapper Drake»), 6.10 («Edgy Eft»), 7.04 («Feisty Fawn»), 7.10 («Gutsy Gibbon»), 8.04 («Hardy Heron»), 8.10 («Intrepid Ibex»), 9.04 («Jaunty Jackalope»);
  - Mandriva 2007.1, 2008.0 и 2009.1.
- Solaris hosts (32 бит и 64 бит):
  - OpenSolaris (2008.05 и выше, «Nevada» релиз 86 и выше);
  - Solaris 10 (u5 и выше).

Рисунок 1. Главное окно программы VirtualBox

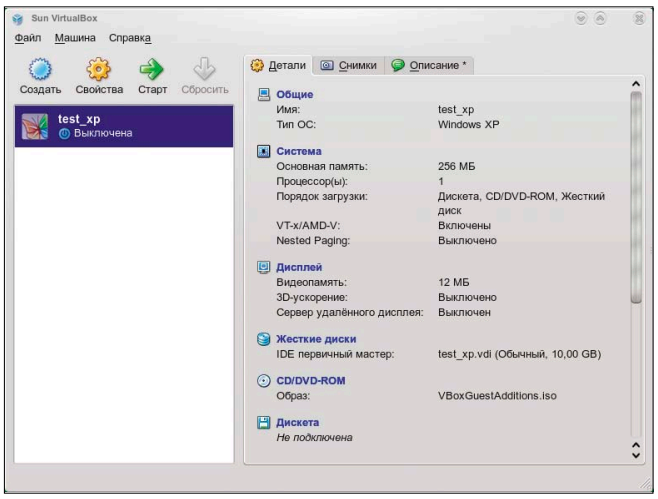
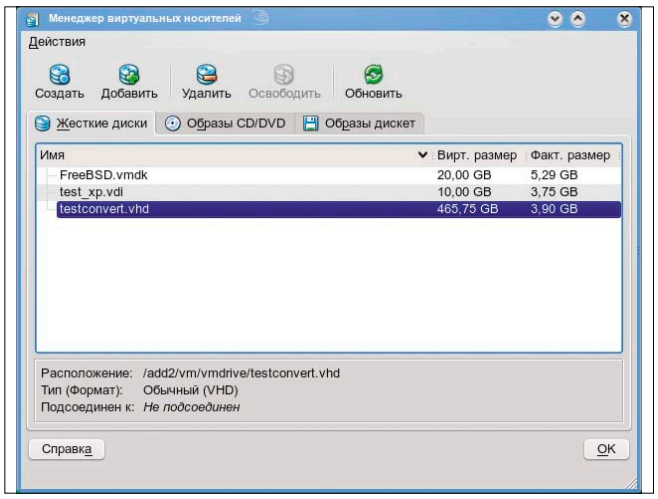


Рисунок 2. Окно менеджера виртуальных носителей



Гостевые операционные системы (Guest Systems)

Что касается списка гостевых операционных систем, поддерживаемых VirtualBox, то он поистине впечатляет:

- MS Windows:
  - Windows 95/98/ME (нет поддержки Guest Additions);
  - Windows NT 4.0 (все версии и все сервис-паки, рекомендуется устанавливать service pack 6a);
  - Windows 2000/XP/Server 2003/Vista/Server 2008 все версии и сервис-паки, включая 64-битные версии;
  - Windows 7 beta;
  - DOS и Windows 3.x.
- Linux. Поддерживаемые версии ядра:
  - Linux 2.4;
  - Linux 2.6 (настоятельно рекомендуется использовать ядро версии 2.6.13 и выше для лучшей производительности).

Из популярных Linux-дистрибутивов поддерживаются: Arch Linux, Debian, OpenSUSE, Fedora, Gentoo, Mandriva, Red Hat, Turbolinux, Ubuntu, Xandros.

- Sun Solaris:
  - Solaris 10, OpenSolaris (32 бит и 64 бит).
- BSD-системы:
  - FreeBSD;
  - OpenBSD версия 3.7 и позднее;
  - NetBSD.
- OS/2:
  - OS/2 Warp от 3.0 и выше;
  - eComStation.

Дополнительно поддерживаются:

- Novell Netware;
- L4;
- QNX;
- и чудесный пункт Other/Unkhow, куда, видимо, должны входить все остальные операционные системы, когда-либо порожденные человечеством.

Как видим, диапазон поддерживаемых систем довольно широк (как минимум, число названия гостевых систем гораздо больше, чем у VMware Server 1.x).

Быстродействие виртуальных машин и всей системы виртуализации

Что касается быстродействия систем, то в Sun Microsystems также постарались на славу.

В последней VirtualBox версии 3.0.2 имеются следующие механизмы улучшения быстродействия:

- Поддержка технологии Hardware Virtualization VT-x для компьютеров на базе процессоров Intel и AMD-V для компьютеров на базе AMD.
- Поддержка функции 3D acceleration.
- Дополнения гостевой операционной системы (Guest Additions) – набор драйверов для виртуальной машины, позволяющий улучшить производительность работы и функции «захвата» мыши и клавиатуры, когда курсор находится в окне виртуальной машины. Данные дополнения доступны для систем:
  - Microsoft Windows NT 4.0;
  - Microsoft Windows 2000;
  - Microsoft Windows XP;
  - Microsoft Windows Server 2003 (все сервис-паки);
  - Microsoft Windows Vista (все редакции);
  - Microsoft Windows 7 Beta;
  - Fedora Core 4, 5, 6, 7, 8, 9 and 11;
  - Redhat Enterprise Linux 3, 4 and 5;
  - SUSE and OpenSUSE Linux 9, 10.0, 10.1, 10.2, 10.3, 11.0 and 11.1;
  - Ubuntu 5.10, 6.06, 7.04, 7.10, 8.04, 8.10 and 9.04;
  - OpenSolaris Nevada (Build 82 и выше; включая OpenSolaris 2008.05, 2008.11 и 2009.06);
  - OpenSolaris Indiana (Developer Preview 2 и выше);
  - Solaris 10 (u5 и выше).

Общее впечатление от быстродействия гостевых систем и всей системы в целом осталось положительным. Старенький компьютер с Pentium IV 2 Ггц, 1,5 Гб RAM и одним жестким диском SATA (не SATA II) весьма шустро поддерживал виртуальную машину Microsoft Windows Server 2003 R2 32 бит с виртуальным объемом RAM 512 Мб, одновременно с открытым текстовым процессором OpenOffice Writer из пакета OpenOffice 3.1 (в котором писалась эта статья), графическим редактором Paint, интернет-браузером FireFox 3.5.2. Замедления были заметны в момент высокой дисковой нагрузки, например, когда загружалась виртуальная машина.

Совет – по возможности используйте различные физические жесткие диски для размещения виртуальных машин и приложений гостевой OS.

Управление VirtualBox

VirtualBox, как и положено грамотно спроектированной системе виртуализации, управляется двумя способами: через консоль управления, вызываемую путем запуска программы VirtualBox или из командной строки посредством программы VBoxManage.

Например, запустить виртуальную машину в host-системе на Linux OpenSUSE с установленным оконным менеджером KDE 4.x можно из окна программы в графическом режиме (GUI). Необходимо перейти в меню «Приложение → Система» и вызвать приложение Virtual Machine. Далее в этом приложении выбрать нужную виртуальную машину (в данном случае это test_xp) и нажать экранную кнопочку Start (см. рис. 1).

А можно воспользоваться интерфейсом командной строки (GLI) и просто выполнить команду:

```
/usr/bin/VBoxManage startvm "test_xp"
```

При удачном запуске в ответ система выдаст сообщение:

```
VirtualBox Command Line Management Interface Version 3.0.4
(C) 2005-2009 Sun Microsystems, Inc.
All rights reserved.

Waiting for the remote session to open...
Remote session has been successfully opened.
```

Аналогичный дуализм присутствует в host-системах на Windows-платформе.

Чтобы запустить виртуальную машину под названием test_convert32 из командной строки, необходимо перейти в каталог C:\Program Files\Sun\xVM VirtualBox:

```
cd "c:\Program Files\Sun\xVM VirtualBox"
```

и выполнить команду:

```
VBoxManage.exe startvm "test_convert32"
```

Или можно вызвать графическое окно программы из меню Programs → Sun xVM VirtualBox и запустить виртуальную машину из него.

И тот и другой методы отлично документированы как в User Manual, так и во встроенной справке, вызываемой по нажатию клавиши <F1>.

Возможность переноса виртуальных машин на другую host-систему

Данный аспект также неплохо проработан разработчиками VirtualBox. Во-первых, есть процедура экспорта-импорта, вызываемая из главного окна VirtualBox «Файл → Импорт конфигурации» или «Файл → Экспорт конфигурации». При этом для данных операций используется Open Virtualization Format (ovf), позволяющий переносить диски на другую платформу виртуализации. Во-вторых, можно без труда переписать соответствующий файл виртуального диска *.vmi на другую host-систему и на его основе поднять аналогичную виртуальную машину.

Возможность переноса виртуальных машин на другую систему виртуализации

Как я уже говорил ранее, существует возможность экспорта-импорта с использованием Open Virtualization Format.

Кроме того, VirtualBox поддерживает виртуальные диски формата *.vhd и *.vmdk, созданные в виртуальных средах Microsoft Virtual PC и VMware Server, VMware Workstation соответственно. Также файлы формата *.vhd используются в Citrix XenServer в качестве посредника при переносе или конвертации виртуальных машин. Эта поистине удивительная всеядность позволяет перейти на VirtualBox с других программных продуктов для виртуализации (см. рис. 2).

При этом использование виртуальных жестких дисков других платформ виртуализации не требует конвертации из одного формата в другой. Вы просто можете подключить, к примеру, файл *.vmdk, работавший ранее под управлением VMware Server, к host-системе с VirtualBox. Поработав с ним какое-то время, можно точно так же вернуть его обратно на VMware Server. В некоторых случаях вам придется настроить драйвера системы. Словом, все как если бы вы переносили один физический винчестер между разными компьютерами.

Лицензирование VirtualBox

Sun Microsystems распространяет свой продукт под двумя лицензиями. Откомпилированные, готовые к работе бинарники, например, инсталляционный файл для Windows – VirtualBox-3.x.x-xxxxx-Win.exe, распространяются под лицензией Personal Use and Evaluation License (PUEL). В рамках данной лицензии вы можете устанавливать этот продукт на свой домашний компьютер без каких-либо ограничений. Вы также можете установить этот продукт VirtualBox на свой рабочий компьютер и использовать его как персональную систему виртуализации.

Вот что говорится по этому поводу в Licensing FAQ (http://www.virtualbox.org/wiki/Licensing_FAQ):

«Also, if you install it on your work PC at some large company, this is still personal use» (кроме того, если вы устанавливаете это на своем рабочем компьютере в некоторой крупной компании, это все еще личное использование).

В то же время для случаев промышленного развертывания системы (например, предоставления большого числа виртуальных машин в коммерческое использование) компания Sun Microsystems настоятельно рекомендует приобрести коммерческую лицензию с поддержкой.

Есть еще VVirtualBox Open Source Edition (OSE) – это исходные тексты программы, распространяемые под лицензией GPL2. Соответственно, в рамках этой лицензии вы можете распространять и изменять исходный код, соблюдая условия GPL2. Правда, такую свободу вы получаете не без потерь. В VirtualBox Open Source Edition отсутствуют некоторые интересные, а подчас и очень важные функции виртуальной host-системы, которые есть в закрытой системе под лицензией PUEL. В частности это:

Remote Display Protocol (RDP) Server – предоставляет возможность соединяться с виртуальной машиной по RDP при помощи стандартного клиента, например, Remote Desktop Connections в MS Windows или rdesktop в UNIX-системах. При этом на самой виртуальной машине не требуется иметь службы сервера RDP.

Рисунок 3. Окно выбора пользовательских установок (Custom Setup) при инсталляции VirtualBox в Windows

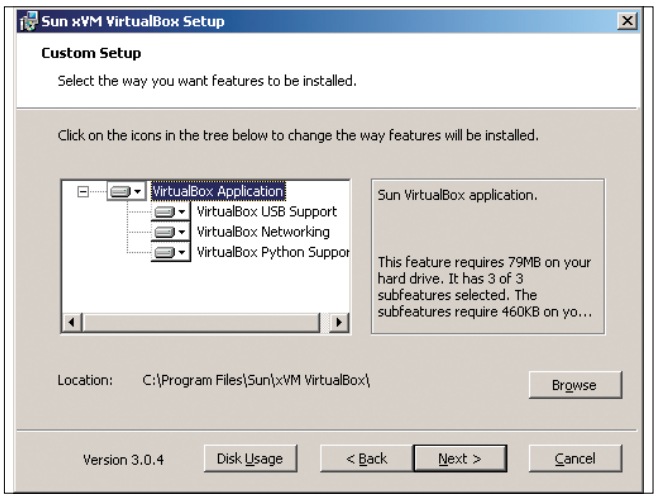


Рисунок 4. Окно с предупреждением о сбросе сетевых подключений во время установки VirtualBox

USB support – поддержка USB-носителей в виртуальных машинах.

USB over RDP – позволяет осуществлять доступ к USB-носителям на виртуальных машинах посредством RDP-протокола.

Serial ATA controller – поддержка виртуальных SATA-дисков. Как и в реальной жизни, виртуальные SATA работают быстрее, и на виртуальной машине их можно иметь больше трех штук.

Регистрация программы

После инсталляции программы предлагается зарегистрировать копию VirtualBox. Это необходимо в первую очередь компании-разработчику VirtualBox для сбора статистики о популярности продукта. Имеет смысл поддержать компанию Sun Microsystems в ее желании знать о том, насколько ее детище пользуется спросом.

Особенности установки VirtualBox

Рассмотрим установку VirtualBox в некоторых операционных системах.

Установка в Linux OpenSUSE 11.1

Для установки VirtualBox в Linux необходимо, чтобы в системе были установлены следующие пакеты.

- Qt 4.3.0 или выше;
- SDL 1.2.7 или выше (эту графическую библиотеку часто называют libsdl или similar).

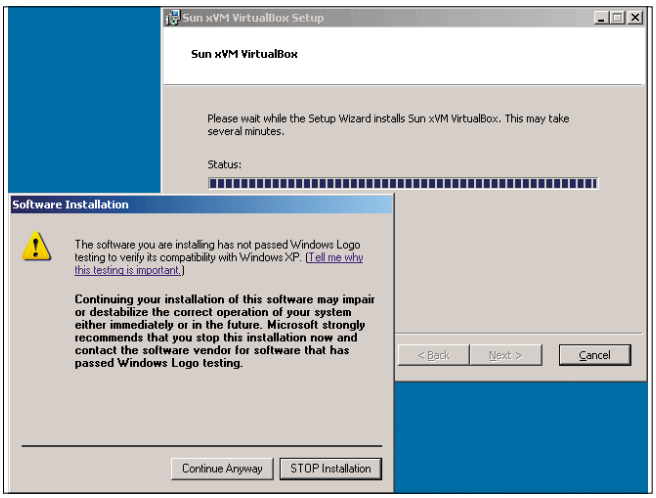
К этому времени у меня уже были установлены соответствующие пакеты. Поэтому скачаем по ссылке http://download.virtualbox.org/virtualbox/3.0.4/VirtualBox-3.0.4_50677_openSUSE111-1.i586.rpm необходимый для установки пакет (я использую 32-битную версию OpenSUSE).

Далее регистрируемся в системе как root, переходим в каталог с сохраненным RPM-пакетом и выполняем команду:

```
rpm -Uvh VirtualBox-3.0.4_50677_openSUSE111-1.i586.rpm
```

После этого у меня в меню появился пункт Virtual Machine с комментарием Sun VirtualBox.

Рисунок 5. Окно запроса разрешения установки дополнительных виртуальных устройств при инсталляции VirtualBox в Windows



Нажав на данный пункт меню, можно запустить главное окно программы (см. рис. 1).

Установка в Windows XP

Установка в Windows проходит обычным образом. Скачиваем и запускаем файл инсталляционный *.exe-файл. Сейчас доступна версия VirtualBox 3.1.0 (http://download.virtualbox.org/virtualbox/3.1.0/VirtualBox-3.1.0-55467-Win.exe).

Установка программы проходит самым обычным способом. Вслед за окном приветствия (Welcome to the Sun xVM VirtualBox Setup Wizard) после нажатия кнопки Next появляется окно выбора пользовательских установок (Custom Setup). В этом окне будет предложено выбрать устанавливаемые компоненты программы, включая поддержку USB (VirtualBox USB support), поддержку сети (VirtualBox Networking) и поддержку расширений на языке Python (VirtualBox Python Support). Как обычно, предоставляется возможность указать местоположение файлов программы (Location) с ручным указанием пути по кнопке Browse. Дополнительно по кнопке Disk Usage можно получить информацию о наличии дискового пространства (см. рис. 3).

В следующем окне в лучших традициях MS Windows будет задан вопрос, создавать ли ярлыки на рабочем столе (Create a shortcut on the Desktop) и в панели быстрого запуска (Create a shortcut in the Quick Launch Bar). После нажатия кнопки Next появляется окно с предупреждением о временном сбросе установленных сетевых соединений и временном отключении сети в связи с инсталляцией VirtualBox Networking (Warning: Network Interfaces. Installing the Sun xVM VirtualBox Networking feature will reset your network connection and temporary disconnect you from the network) (см. рис. 4).

Все, подготовка к инсталляции закончена, о чем становится понятно из появившегося окна. (Ready to install). После нажатия соответствующей клавиши начнется процедура установки программы. Во время инсталляции в системе будет создано несколько новых устройств для поддержки виртуальных режимов, таких как VirtualBox Networking, и система выдаст соответствующее окно с запросом о разрешении или отклонении данного действия (см. рис. 5).

По завершении процесса появится окно, сообщающее о завершении процесса инсталляции (Sun xVM VirtualBox installation complete), с предложением сразу же запустить окно управления VirtualBox (Start Sun xVM VirtualBox after installation).

Все, система виртуальных машин установлена, можно приступать к работе.

***

Мы рассмотрели основные вопросы использования VirtualBox в качестве системы персональной виртуализации. В следующей статье будут рассмотрены вопросы создания и сетевого подключения виртуальных машин, а также улучшения производительности, технологии миграции с других платформ. EOF

1. Домашняя страничка проекта – http://www.virtualbox.org.
2. Информация о проекте на официальном сайте компании Sun Microsystems – http://www.sun.com/software/products/virtualbox.
3. Страничка GNU General Public License, version 2 – http://www.gnu.org/licenses/old-licenses/gpl-2.0.html.

Визитка

РАШИД АЧИЛОВ, поклонник FreeBSD с 14-летним опытом использования ее в совмещенных с Windows сетях и сторонник Open Source. Администратор сетей и средств защиты крупной торговой сети

Один UPS на двоих

FreeBSD в домене Windows

Ситуация, когда к одному UPS подключено несколько серверов, — не исключение, а скорее правило. Как вовремя отключить сервер, если программа для работы с данным UPS существует только для Windows?

И вольт 220 – и то пополам...

На сегодняшний день существует широкий диапазон устройств, способных обеспечивать резервное питание серверов в той мере, в которой это необходимо. Далеко не ко всем устройствам (а точнее говоря только к UPS производства APC [1]) существуют клиенты под FreeBSD, позволяющие принимать по сети сообщения от сетевых UPS и соответственно обрабатывать их. Как поступить в том случае, если имеется только клиент под Windows, а необходимо запрограммировать реакцию на сообщение UPS о том, что необходимо прервать работу в связи с пропаданием питания?

Моделировать ситуацию будем в следующих условиях – имеется UPS Powerware 9120 [2], к которому подключены компьютер под управлением Windows и компьютер под управлением FreeBSD. Управление UPS осуществляется через поставляемый вместе с UPS кабель RS-232 с компьютера с ОС Windows, на котором установлено программное обеспечение LanSafe. LanSafe можно бесплатно скачать непосредственно с сайта производителя [2]. Требуется – при долговременном пропадании питания вместе с корректным завершением работы сервера под управлением ОС Windows корректно завершать работу сервера под управлением FreeBSD.

Задача эта на самом деле решается достаточно просто в том случае, когда программное обеспечение для управления UPS имеет возможность по наступлению некоторого события выполнять определенную внешнюю команду. Уже упомянутый выше LanSafe имеет возможность на каждое событие (а их достаточно много) настроить три вида реакции – оповещение по почте, оповещение широковещательным сообщением и выполнение внешней команды, причем все виды реакции можно настраивать независимо друг от друга.

Таким образом, обработка события Power failure (пропадание напряжения питания), которое возникает в том случае, если исчезает напряжение питания, будет происходить согласно следующей схеме (см. рис. 1).

Для реализации данной схемы нам понадобится, кроме самого LanSafe:

Для Windows – программа удаленного выполнения команд по протоколу SSH plink.exe и программа генерации ключей SSH (входят в состав бесплатного SSH-клиента PuTTY, скачать можно с [3]).

Для FreeBSD – программа выполнения команды от имени другого пользователя sudo (ports/security/sudo) и SSH-сервер для приема и обработки команды. Можно использовать стандартный OpenSSH, но я предпочитаю сервер от SSH Communication Inc. (ports/security/ssh2-nox11).

Итак, в общем случае задача может быть сформулирована так: «Как из одной программы запустить другую на удаленном сервере». Сформулировав же задачу в общем виде, переходим к частностям.

Частности

Подробности реализации мы начнем рассматривать с последнего звена цепочки – с сервера FreeBSD. Чтобы отправить на него команду, необходимо, чтобы кто-то эту команду принял. Этот «кто-то» будет у нас псевдопользователем (то есть пользователем, не имеющим пароля, но имеющим допустимый шелл) upsadmin.

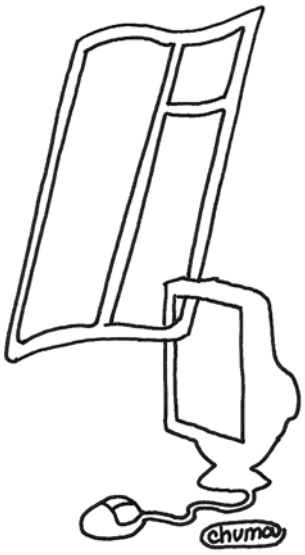
Создаем учетную запись пользователя:

```
# pw useradd upsadmin -d /usr/home/upsadmin ↵
        -m -s /bin/sh -c "UPS administrator" -h -
```

В случае успешного завершения команда pw не выводит ничего. Переключаемся на учетную запись этого пользователя (то, что все это выполняется от пользователя root, я думаю, упоминать излишне) и настраиваем SSH для использования беспарольной авторизации по публичному ключу:

```
# su -l upsadmin
# ssh-keygen2 -t dsa
```

```
Generating 2048-bit dsa key pair
   3 o.oOo..oOo.o
Key generated.
2048-bit dsa, upsadmin@citycat.shelton.net, Wed Oct 28 2009
00:37:39 +0600
Passphrase :
Again      :
Key is stored with NULL passphrase.
```



Это набор инструментов, который один раз настраиваешь, и скрипты будут выполняться, серверы вовремя выключаться…

```
(You can ignore the following warning if you are
generating hostkeys.)
This is not recommended.
Don't do this unless you know what you're doing.
If file system protections fail (someone can access
the keyfile),
or if the super-user is malicious, your key can be used
without the deciphering effort.
Private key saved to /usr/home/upsadmin/.ssh2/id_dsa_2048_a
Public key saved to /usr/home/upsadmin/.ssh2/id_dsa_2048_a.pub
```

Предупреждение о возможном нарушении политики безопасности, при которой каждый личный ключ из пары ключей должен быть защищен паролем (а иначе злоумышленник, получивший доступ к нему, сможет им воспользоваться), игнорируем – мы не будем пользоваться собственными ключами пользователя upsadmin.

Переходим в каталог .ssh2 (он будет создан автоматически) и создаем два пустых файла – identification и authorization.

В файл identification прописываем строку:

```
IdKey id_dsa_2048_a
```

Файл authorization пока не трогаем, он нам понадобится потом.

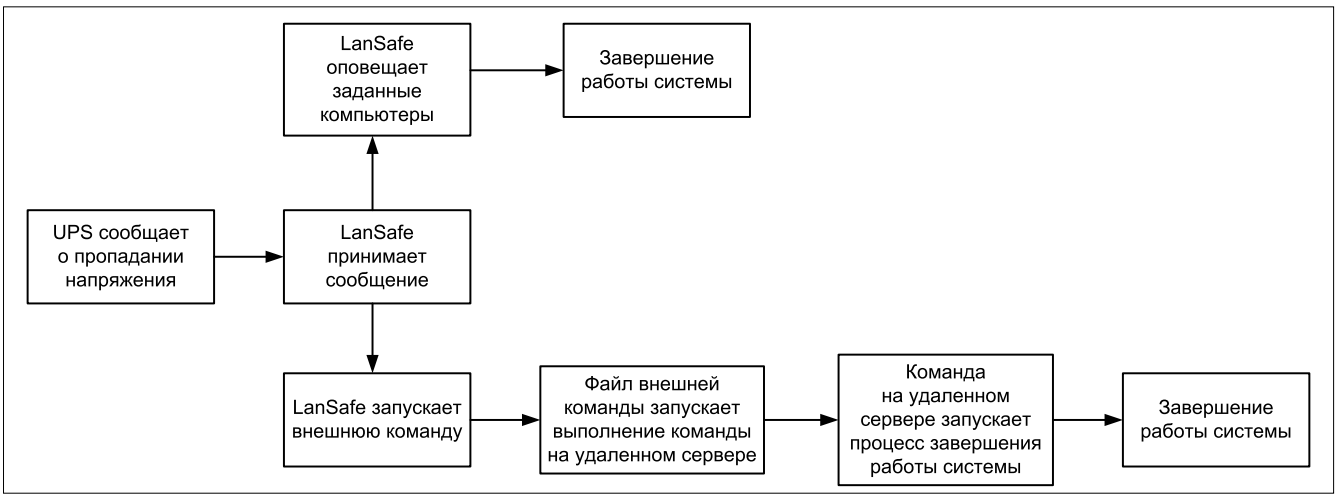
Далее наделяем пользователя upsadmin правами на выключение и перезагрузку компьютера. Устанавливаем пакет sudo, если он не установлен, и прописываем в файле /usr/local/etc/sudoers (он будет автоматически создан и заполнен примерами) следующие строки:

```
# Cmnd alias specification
Cmnd_Alias      SHUTDOWN = /sbin/shutdown
Cmnd_Alias      HALT = /sbin/halt
Cmnd_Alias      REBOOT = /sbin/reboot
Cmnd_Alias      IPFW = /sbin/ipfw

# UPS remote command executor
upsadmin        ALL=(root) NOPASSWD: SHUTDOWN, ↵
        NOPASSWD: HALT, NOPASSWD: REBOOT, NOPASSWD: IPFW
```

Последняя команда (IPFW) нам понадобится только для тестирования, после того как все будет отлажено, ее

Рисунок 1. Последовательность обработки события UPS для двух серверов



можно и нужно убрать. Что мы описали данными строками? Мы описали, что при выполнении через sudo команд shutdown, halt, reboot и ipfw пользователь upsadmin по уровню привилегий равен администратору (root) и при выполнении этих команд ему нет необходимости вводить свой пароль.

Проверяем.

```
# su -l upsadmin
$ sudo ipfw sh
```

```
ipfw: DEPRECATED: 'sh' matched 'show' as a sub-string
… (вывод удален)
65000         0             0 allow ip from any to any
65535       457         45540 deny ip from any to any
```

Отлично. Теперь создадим предыдущий элемент схемы – скрипт, который будет выдавать команду выключения питания. Выполняться этот скрипт будет от пользователя upsadmin. Для того чтобы не указывать расположение скрипта, просто создадим в домашнем каталоге пользователя upsadmin каталог bin.

Почему именно bin? Мы указали /bin/sh в качестве стартового шелла. При запуске /bin/sh автоматически выполняется его стартовый скрипт – файл .profile в домашнем каталоге пользователя (подробнее – man sh). Файл .profile, как правило, содержит следующую строку:

```
PATH=/sbin:/bin:/usr/sbin:/usr/bin:/usr/games:/usr/local/ ↵
        sbin:/usr/local/bin:/usr/X11R6/bin:$HOME/bin; ↵
        export PATH
```

Эта строка определяет, где /bin/sh будет искать программу в том случае, если она запускается без указания пути. Проверьте, что PATH содержит $HOME/bin. Разумеется, можно создать каталог с другим именем и указать его имя в PATH или вообще не создавать ничего и каждый раз указывать полный путь.

Сам скрипт – это всего лишь две строчки:

```
#!/bin/sh

logger -i -p local6.info -t lansafe Shutdown started
sudo shutdown -p now
```

Первая строка добавлена исключительно для удобства – при выполнении скрипта всякий раз в файл, определяемый /etc/syslog.conf как local6, будут записываться сообщения о причине выключения. Предварительно необходимо определить в /etc/syslog.conf facility local6, например, следующим образом:

```
local6.*        /var/log/powerdown
```

При этом все сообщения с facility local6 будут направляться в файл /var/log/powerdown. Строго говоря, пример не совсем корректный, так как в этот файл могут быть направлены сообщения от любой программы, присвоившей своим сообщениям facility local6. Для большей корректности следовало бы написать так:

```
!lansafe
local6.*        /var/log/powerdown
```

Ну вот, теперь достаточно запустить скрипт powerdown (так называется файл скрипта) – питание будет выключено. Переходим еще левее по схеме – к скрипту, выполняемому на Windows. Вот здесь нам и понадобится заполнить файл authorization в каталоге .ssh2. Прописываем в него такую строчку:

```
Key winbox.pub
```

Прописывается сюда имя файла публичного ключа, по которому будет выполняться авторизация. Файл этот, разумеется, сначала должен быть создан. Причем именно на том компьютере, с которого будет запускаться команда, в нашем случае на том, к которому подключен UPS. Вот для этого нам и понадобится программа puttygen.exe, которая в PuTTY играет роль, аналогичную команде ssh2-keygen, Запускаем и создаем личный и общий ключи (см. рис. 2).

Сохраняем файлы winbox.ppk (приватный ключ) и winbox.pub (публичный ключ). Пароль на приватный ключ не устанавливаем. Мы используем авторизацию по публичному ключу только для того, чтобы избежать задания паролей в открытом тексте. Какой же нам смысл задавать пароль на ключ, который все равно потребуется указывать в открытом виде?

Рисунок 2. Создание пары ключей с помощью программы puttygen.exe

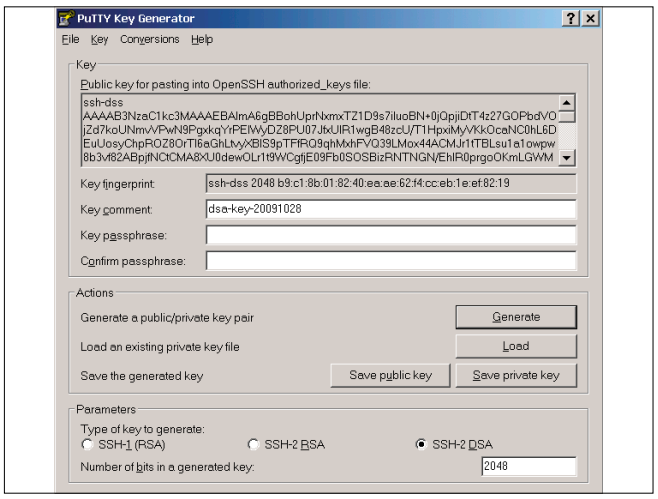
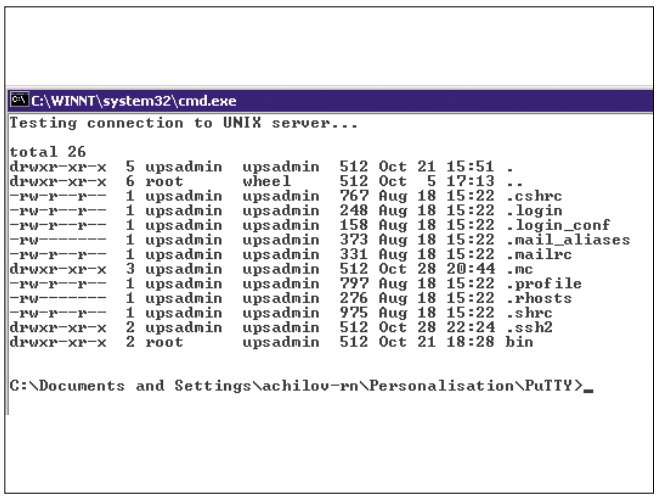


Рисунок 3. Выполнение команды из среды Windows на удаленном сервере



Файл winbox.pub распространяем свободно – в частности, его нужно поместить в подкаталог .ssh2 домашнего каталога пользователя upsadmin на сервере FreeBSD. Файл же winbox.ppk оберегайте насколько возможно – именно он является аналогом пароля. Хищение приватного ключа, не защищенного паролем, автоматически означает, что любой сможет выдать себя за лицо, которое этим ключом авторизуется. Разумеется, это недостаток данного способа, но... за удобство надо чем-то платить?

После этого мы можем, запуская программу plink.exe, выполнять команды на сервере FreeBSD от имени того пользователя, которое указываем, в нашем случае upsadmin (см. рис. 3):

```
echo "Testing connection on UNIX server..."
plink.exe -ssh -P 22 -2 -i winbox.ppk ↵
       upsadmin@192.168.1.1 ls -la
```

Подробно все ключи к команде plink описаны в документации на PuTTY – файле putty.chm.

Создаем для файлов отдельный каталог, например c:\upsmgmt, и переносим в него все файлы, что нам понадобятся: приватный ключ winbox.ppk, программу plink.exe. Сюда же можно положить и публичный ключ winbox.pub.

И наконец, пишем собственно сам скрипт:

```
c:
cd \upsmgmt
plink.exe -ssh -P 22 -2 -i winbox.ppk ↵
       upsadmin@192.168.1.1 "powerdown"
```

Относительно этого скрипта следует сделать несколько замечаний.

- Во-первых, когда мы запускаем скрипт вручную, он запускается в том каталоге, в котором мы в данный момент находимся. Когда же скрипт будет запущен программой LanSafe, то он запустится в том каталоге, который Windows полагает «по умолчанию». Скорее всего, это не будет интересующий нас каталог, потому перед запуском команды необходимо явно указать переход в нужный нам каталог (в примере c:\temp\2).
- Во-вторых, powerdown – не стандартная команда FreeBSD, а имя скрипта, который мы создали на предыдущих шагах. Этот скрипт лежит в %HOME%/bin, потому путь к нему можно не указывать.

Что ж, осталось последнее. Сохраняем скрипт в файл, например testlink.bat, и помещаем его в тот каталог, в который делаем переход (в примере c:\upsmgmt).

Запускаем программу LanSafe, выбираем пункт Configuration → Event Notification и в настройках для события Power failure задаем выполнение команды c:\upsmgmt\testlink.bat с задержкой в 5 секунд (такая короткая задержка только для тестирования, в реальной жизни, конечно же, задержка должна быть больше) (см. рис. 4).

Все. Цепочка замкнулась. При возникновении события Power failure после заданной задержки будет запущен скрипт testlink.bat, который запустит plink.exe, который запустит powerdown на удаленном сервере, который в свою очередь запустит команду «shutdown -p now».

К особенностям поведения здесь можно отнести одну вещь – при первом запуске скрипта непосредственно самим LanSafe plink.exe выдает запрос на кэширование ключа узла (точно так же, как он это делает всегда при первом подключении). Все бы ничего, но только нужно учитывать тот факт, что запрос этот выдается исключительно на консоль сервера Windows.

Если настраивать LanSafe, находясь в терминальной сессии, запроса мы не увидим до тех пор, пока не перехватим консоль с помощью программ NetOp или Radmin, поэтому перед тем как начинать эксплуатацию системы, необходимо провести тестовый запуск, для чего в настройках LanSafe есть кнопка Test, позволяющая вручную имитировать наступление тестируемого события.

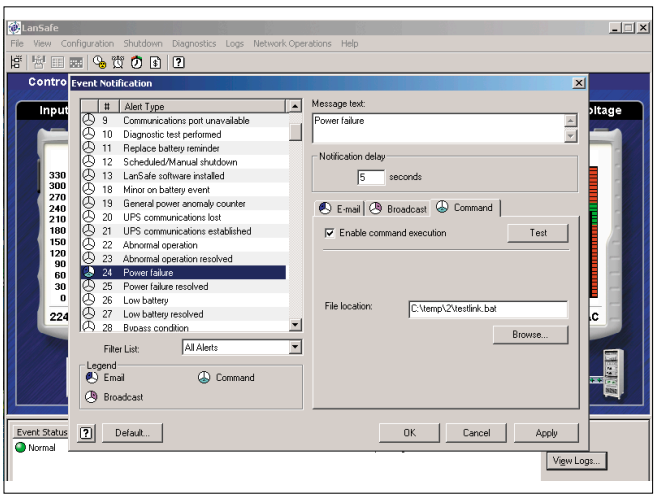
Существуют ли альтернативы описанному способу? Да, компания Eaton выпускает LanSafe не только для операционных систем Windows, но и для Linux тоже [4], в списке поддерживаемых дистрибутивов – Red Hat версии 3 (ESи AS), версии 4 и 5 (ES, AS и Desktop), Fedora Core версии 5-8 и SuSE версии 8-10 (в том числе и Enterprise Linux Server). Вполне возможно попробовать его запустить, установив модуль совместимости с Linux-программами (ports/emulators/linux-base-f7 или ports/emulators/linux-base-f8). При этом прием оповещения от контроллера UPS настраивается стандартным способом.

***

Описанный здесь способ – это всего лишь еще одна «малая механизация», набор несложных инструментов, который один раз настраиваешь на совершение определенных действий – и про них можно навсегда забыть, скрипты будут выполняться, серверы выключаться в нужное время... И дом, который построил Джек, не рухнет, по крайней мере, из-за неверного выключения питания. EOF

1. Сайт American Power Conversion, производителя UPS APC – http://www.apcc.com.
2. Сайт Eaton, производителя UPS Powerware – http://powerquality.eaton.com/Russia/?cx=67.
3. Сайт программы PuTTY – http://www.putty.org.
4. Ссылка для загрузки LanSafe версии 6 – http://powerquality.eaton.com/Support/Software-Drivers/Downloads/lansafe6.asp.

Рисунок 4. Настройка LanSafe на выполнение команды при пропадании питания



Визитка

АЛЕКСЕЙ ВАТУТИН, MCSE, MCITP, ICASA. Системный инженер компании «КРОК» и автор статей, посвященных управлению мобильными устройствами, на портале ITband

Удобно, безопасно, недорого

Управление мобильными устройствами на предприятии

Мобильные устройства все чаще становятся заменой обычных ПК вне офиса, но можно ли управлять ими наравне с ПК, сохраняя удобство использования?

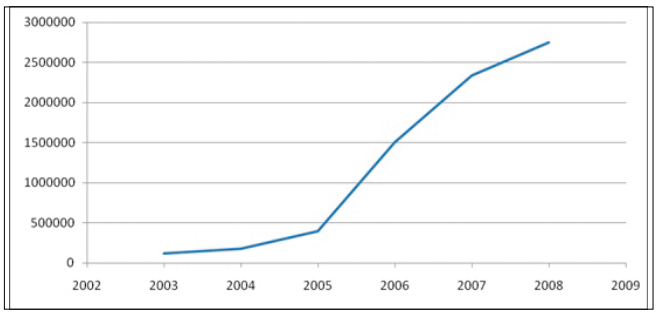
Количество всевозможных мобильных устройств растет день ото дня. По мнению аналитиков SmartMarketing [1], в 2008 году в России было продано более 2,7 млн смартфонов, коммуникаторов, КПК и интернет-планшетов. При этом 35,79% составили так называемые WID-устройства, которые имеют крупный и/или сенсорный экран, а также алфавитную клавиатуру. Остальной процент составили смартфоны, т.е. устройства с несенсорным экраном и цифровой клавиатурой (преимущественно это устройства на базе платформы Symbian, но также и некоторые модели с Linux). Согласно исследованиям компании IDC (Worldwide Mobile Worker Population, IDC), в 2009 году более 23% всех работающих в мире можно было отнести к категории мобильных пользователей, которые периодически либо постоянно работают вне офиса, при этом доля сотрудников, работающих удаленно, из дома, не превышает 5% (см. рисунок).

Сейчас уже ни у кого не вызывает удивления возможность получения корпоративной почты на мобильный телефон либо использование ресурсов корпоративной сети с коммуникатора, но зачастую подобные возможности обеспечиваются путем индивидуальной настройки каждого устройства. В качестве последствий такого подхода можно выделить проблемы, указанные в таблице.

При этом доступ к ресурсам внутренней сети нередко сопровождается грубыми нарушениями правил информационной безопасности и ставит под удар всю деятельность компании. В условиях существующей жесткой конкуренции любая утечка информации негативно отражается на деловом имидже компании и способна повлечь за собой крупные финансовые потери.

Еще одной причиной пересмотра подхода к управлению мобильными устройствами может стать стремление к сокращению затрат в данной области.

Рост продаж мобильных устройств в России, исследования SmartMarketing 2003-2008



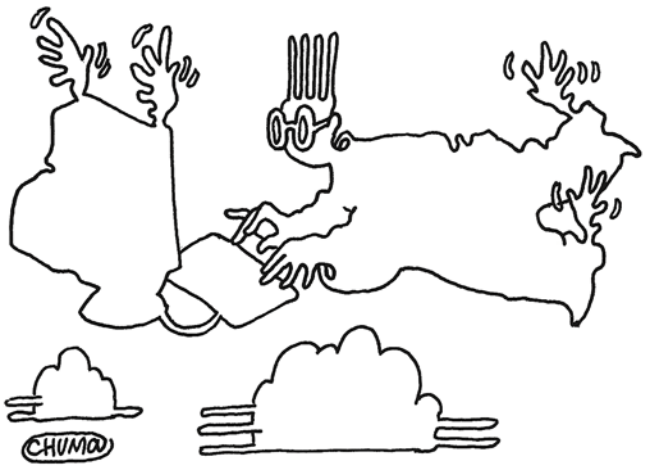
Как нужно управлять мобильными устройствами?

Как же можно организовать работу сотрудников с использованием современных средств коммуникации, чтобы это было удобно, безопасно и недорого? Предлагаю рассмотреть концепцию управления мобильными устройствами (Mobile Device Management, MDM). Если коротко, то данный подход предполагает внедрение некоего ПО, при помощи которого в компании организуется процесс по инициализации, использованию и поддержке мобильных устройств. Однако наличие данного ПО не отменяет необходимости четкого понимания, для каких целей производится внедрение и что принимается в качестве конечного результата.

Управление мобильными устройствами – это действия, осуществляемые при помощи вспомогательного ПО, позволяющие доставлять приложения и данные на мобильные устройства (ноутбуки, мобильные телефоны, КПК и т.д.), а также выполнять их настройку. Основной целью этих действий является достижение оптимального состояния между безопасностью и удобством использования данных устройств при минимизации затрат на обслуживание и времени простоя.

Обычно в качестве цели при внедрении службы управления мобильными устройствами обозначается достижение следующих результатов:

Обеспечение безопасности при передаче информации и при ее хранении на устройстве – обязательное использование пароля либо PIN-кода на устройстве, шифрование содержимого устройства и карт памяти, шифрование потока с данными при работе с корпоративной сетью, использование цифровых сертификатов.



Внедрение службы управления мобильными устройствами способно облегчить пользователям жизнь

Упрощение и удешевление процесса ввода устройства в эксплуатацию – автоматизация рутинных действий, использование единообразной конфигурации.

Упрощение и удешевление процесса поддержки пользователей и устройств – проведение инвентаризации, удаленное внесение изменений в конфигурацию устройства, возможность удаленной очистки устройства в случае его утери, делегирование пользователю ряда действий – восстановление пароля, блокировка устройства.

Внедрение ограничивающих политик – запрет на изменение сетевых настроек, на установку ПО, на подключение к определенным узлам сети, на использование периферии – камер, Bluetooth- и Wi-Fi-адаптеров.

Как это отражается на работе сотрудников после внедрения? В качестве ответа можно рассмотреть некий абстрактный пример, когда сотруднику Иванову, работающему в компании «Компания», выдается корпоративный коммуникатор, при помощи которого г-н Иванов выполняет следующие задачи:

- получение и отправка корпоративной почты с использованием адресной книги предприятия, синхронизация календаря, контактов и запланированных задач;
- общение с коллегами при помощи службы мгновенного обмена сообщениями;
- использование корпоративного портала;
- использование корпоративного бизнес-приложения.

При этом системный администратор компании должен быть уверен в том, что любая информация, находящаяся в устройстве, не станет достоянием конкурентов даже в случае его утери. Также администратор должен иметь возможность выполнения следующих действий вне зависимости от места нахождения устройства:

- производить обновление установленного ПО и установку нового;
- изменять конфигурацию устройства;
- заблокировать аппарат либо удалить данные на устройстве и вернуть его в исходные заводские настройки в случае необходимости.

До внедрения службы управления мобильными устройствами схема введения устройства в эксплуатацию и его последующая поддержка выглядят следующим образом:

Системный администратор, получив устройство со склада, тратит от 2 до 4 часов на подготовку устройства для пользователя – конфигурирование аппарата, установка и настройка необходимого ПО и т.д. После этого пользователь получает устройство и начинает его эксплуатацию. В этот период г-н Иванов может беспрепятственно устанавливать на устройство любые программы. В случае возникновения каких-либо проблем ему приходится воз-

Список проблем при использовании мобильных устройств

Проблема	Категория
Отсутствие единого регламента по инициализации устройств и организации доступа к ресурсам компании	Организационные
Временные и материальные затраты на настройку каждого устройства, притом что большая часть этих настроек у них совпадает	Организационные Инициализация устройства
Слабые возможности по обеспечению безопасности информации, хранимой на устройстве	Безопасность информации
Затруднения при обновлении установленного ПО либо при установке нового. Невозможность ограничения установки ПО пользователем	Поддержка устройства Безопасность информации
Затруднения при оказании технической поддержки пользователей	Поддержка устройства
При необходимости внесения изменений в конфигурацию администратор должен иметь физический доступ к устройству	
Отсутствие возможности инвентаризации устройств	

вращать аппарат системному администратору для проведения полной диагностики. Также пользователю приходится посещать сотрудника ИТ-службы для обновления ПО и при необходимости внесения изменений в конфигурацию аппарата. В случае утери устройства любая информация, находящаяся на нем (например, электронная почта и локальные копии корпоративных отчетов), может попасть в руки конкурентов.

Теперь предположим, что в компании «Компания» работают 200 сотрудников, которым, как и г-ну Иванову, необходим коммуникатор для выполнения служебных обязанностей. Становится ясно, что большая часть действий, выполняемых системным администратором, является повторяемой и может быть автоматизирована. Уровень безопасности информации, обеспечиваемый при таком подходе, также является очень низким, а удобство для конечного пользователя сомнительным.

Итак, г-н Иванов, устав от необходимости постоянного посещения системного администратора, покидает компанию. Однако по прошествии некоторого времени он возвращается назад и обнаруживает, что в компании изменился подход к работе с мобильными устройствами. Теперь «Компания» использует ПО для управления мобильными устройствами, и схема инициализации и поддержки устройства выглядит так:

Системный администратор, получив устройство со склада, затрачивает от 5 до 15 минут для установки на устройство клиентской части ПО либо для включения устройства в домен компании и передачу устройства пользователю. Сразу после установки клиентской части устройство, используя один из возможных сценариев подключения (GPRS, EDGE, 3G, Wi-Fi), связывается с сервером MDM и получает свои настройки в зависимости от использованного шаблона либо от подразделения, в которое оно входит. Также происходят автоматическая установка и настройка ПО. Во избежание потенциальных проблем администратор может использовать запрет на установку и удаление ПО пользователем, а также задействовать механизмы шифрования (в т.ч. памяти устройства и карт памяти) и аутентификации, способные обеспечить должный уровень безопасности хранимой на устройстве информации. При необходимости внесения изменений в конфигурацию устройства администратору достаточно внести изменения в шаблон, используемый данным устройством, либо изменить/добавить групповую политику, а обновление и установка ПО не являются поводом для того, чтобы тревожить пользователя. В случае утери устройства администратор может выполнить его удаленную очистку (Remote Wipe).

Таким образом, вернувшись, г-н Иванов почувствовал, что внедрение службы управления мобильными устройствами способно облегчить пользователям жизнь. А поскольку г-н Иванов вернулся в компанию с повышением, став руководителем одного из подразделений, он может наблюдать общее снижение количества времени, затрачиваемого его сотрудниками на инциденты с мобильными устройствами. При этом если бы г-н Иванов был руководителем ИТ-подразделения, он бы знал, что материальные затраты, связанные с разрешением подобных инцидентов, также резко сократились.

Такой вот пример…

Что использовать?

Так какое ПО можно использовать для управления мобильными устройствами? На рынке представлено достаточно большое количество всевозможных разработок, но наиболее известными из них являются следующие:

BlackBerry от Research In Motion (http://www.rim.com) – одна из наиболее известных разработок в данной области. Широкую известность продукту принесло использование закрытого алгоритма шифрования, что гарантирует высокий уровень защиты информации. Но из этого плюса вытекают два минуса – не во всех странах данный продукт может быть внедрен (из-за разногласий с местным законодательством) и жесткая привязка к типу используемых устройств.

Afaria от Sybase (http://www.sybase.com) – ПО от известного разработчика, позволяющее управлять устройствами, работающими на практически всех известных платформах. В качестве плюсов можно отметить модульную компоновку ПО и возможность использования OMA DM (в качестве отдельного модуля).

Продукты из пакета Good for Enterprise от Good Technology (http://www.good.com).

MobiControl от SOTI (http://www.soti.net).

Exchange Server от Microsoft (http://www.microsoft.com) – предоставляет ряд возможностей по управлению устройствами на платформах Windows Mobile, Symbian с пакетом Mail for Exchange, iPhoneOS при помощи политик ActiveSync. В версии Exchange 2003 есть возможность использования политик паролей для устройств, шифрования данных, получаемых от Exchange Server, использования S/MIME и удаленной очистки устройства. В Exchange Server 2007 добавились следующие функции – восстановление паролей, управление периферией устройства, шифрование памяти и карты памяти устройства, управление приложениями и т.д.

System Center Mobile Device Manager от Microsoft (http://www.microsoft.com) – легко интегрируется в существующую среду на базе продуктов от Microsoft, обеспечивает высокую степень защиты, в том числе благодаря использованию т.н. Mobile VPN. Из минусов можно отметить требование к клиентам – только Windows Mobile версии не ниже 6.1, чувствительность к «прокачанным» прошивкам, а также трудности использования с некоторыми мобильными операторами в России по техническим причинам.

В любом случае выбор того или иного ПО должен определяться поставленными целями.

***

В этой статье я попробовал коротко рассказать о том, для чего применяется служба управления мобильными устройствами на предприятии. Также хочу отметить, что, несмотря на традиционное техническое отставание нашей страны от мировых тенденций, интерес к данному направлению в России становится все больше, и, возможно, уже подходит время задуматься о том, как это может быть реализовано в вашей компании. EOF

1. http://www.smartmarketing.ru/node/4 – исследование «Российский рынок WID и смартфонов за 2008 г.», SmartMarketing.

Визитка

ВИТАЛИЙ БАНКОВСКИЙ, технический директор, автор более 20 статей, главный двигатель прогресса в одной из интернет-компаний

Тратим меньше, спим больше

с VMware Sphere 4.0. Часть 2

Продолжение цикла статей по разворачиванию кластерной системы виртуальных серверов масштаба предприятия. Построение отказоустойчивых виртуальных машин

Обзор средств VMware для создания отказоустойчивых систем

VMware предоставляет два типа отказоустойчивости:

High Availability (далее по тексту – HA). В этой конфигурации VMware анализирует состояние виртуальных машин и перезапускает виртуальную машину на другом сервере, если обнаружен сбой.

Fault Tolerance (далее по тексту – FT). В этом режиме на различных серверах VMware запущены идентичные копии виртуальных машин, причем одна является ведущей, вторая – ведомой, она получает те же самые запросы, что и ведущая. Если обнаружена фатальная ошибка ведущей виртуальной машины, то происходит переключение на ведомую виртуальную машину, причем без потери данных и с минимальным временем переключения.

Ключевые компоненты кластера

Кластер VMware состоит из следующих компонентов:

- сервера VMware;
- сервер Vcenter для управления серверами VMware;
- разделяемое хранилище данных и операционных систем VM;
- сетевая структура.

Для проверки совместимости оборудования необходимо обратиться к списку рекомендуемого оборудования http://www.vmware.com/go/hcl.

Для поддержки FT необходимы процессоры следующих серий:

- Intel 31xx;
- Intel 33xx;
- Intel 52xx;
- Intel 54xx;
- Intel 55xx;
- Intel 74xx;
- AMD 13xx;
- AMD 23xx;
- AMD 83xx.

Одним из главных компонентов при создании кластера является разделяемое между серверами VMware хранилище, представленное с помощью одной из следующих технологий:

- NFS;
- iSCSI;
- SAN.

В моем случае я использовал SAN (Storage Area Network, Сеть хранения данных), причем каждый сервер VMware был подключен к двум коммутаторам SAN, к которым в свою очередь было подключено хранилище SAN с двумя независимыми контролерами, таким образом обеспечивалась отказоустойчивость на уровне доступа серверов VMware к хранилищу.

Установка и предварительная настройка серверов

Установка и предварительная настройка серверов VMware не отличается оригинальностью, но при этом нужно учесть следующие особенности.

Каждый сервер должен иметь как минимум два сетевых интерфейса. Первый интерфейс используется для рабочего трафика виртуальных машин, второй – для служебного трафика, например для миграции виртуальных машин между серверами и для поддержки FT, причем производитель рекомендует выделенный канал для последнего.

Установка и настройка VMware vCenter Server

Для установки Vcenter нам понадобится компьютер с операционной системой Windows. Сейчас поддерживаются следующие версии:

- Windows 2000 Server;
- Windows 2000 Advanced Server;
- Windows XP Professional;
- Windows Server 2003 (Web, Standard, and Enterprise).

Системные требования:

- минимум 1 Гб оперативной памяти;
- минимум процессор Pentium 4.

В зависимости от поставки программного обеспечения VMware установка может быть произведена с компакт-диска, локально или с удаленного хранилища. После уста-

новки необходимо перезапустить сервер для активации сервисов Vcenter.

Настройка серверов VMware

На этом этапе мы научимся управлять серверами VMware через центральную программу Vcenter. Для этого запускаем программу Vsphere Client (установка описана в моей предыдущей статье [1]) и подключаемся к серверу, где установлена программа Vcenter. В качестве имени пользователя и пароля необходимо использовать учетную запись администратора самой операционной системы Windows.

Далее нам необходимо произвести следующие шаги:

- создать виртуальный дата-центр;
- создать кластер;
- подключить к созданному кластеру наши серверы VMware.

Создание виртуального дата-центра

После подключения к Vcenter с помощью клиента Vsphere мы увидим окно, где и нужно создать дата-центр.

Создание кластера

Выделяем созданный дата-центр, при этом появится ссылка для создания кластера (см. рис. 1). При создании нужно указать, что будет использоваться VMware High Availability. Это также необходимо, если будет использоваться Fault Tollerance.

Подключение серверов VMware к Vcenter для централизованного управления

Для этого необходимо выделить кластер и нажать на ссылку Add Host. При этом появится приглашение для ввода адреса сервера VMware и учетной записи администратора сервера VMware на этом сервере (см. рис. 2).

Рисунок 1. Создание кластера

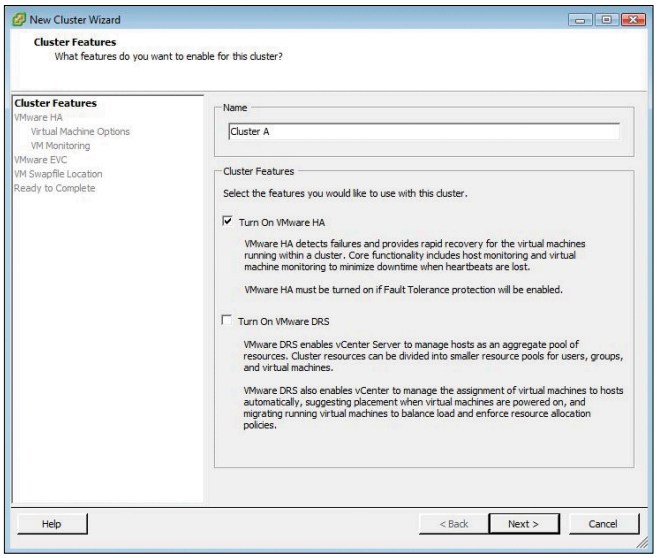


Рисунок 2. Подключение сервера VMware

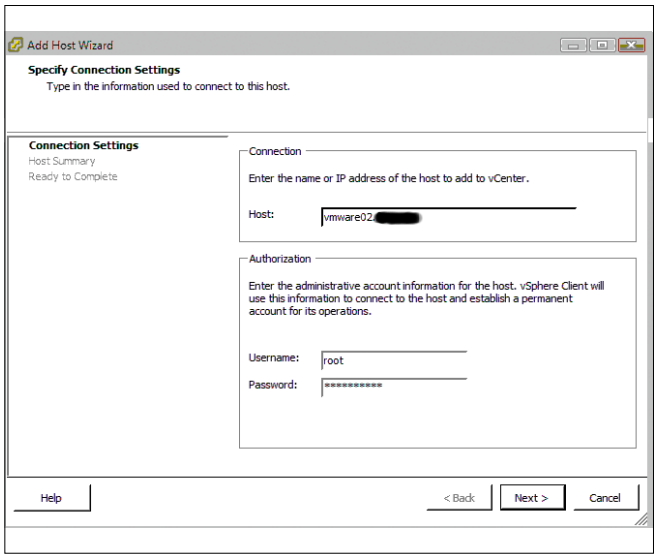


Рисунок 3. Сетевая структура сервера VMware

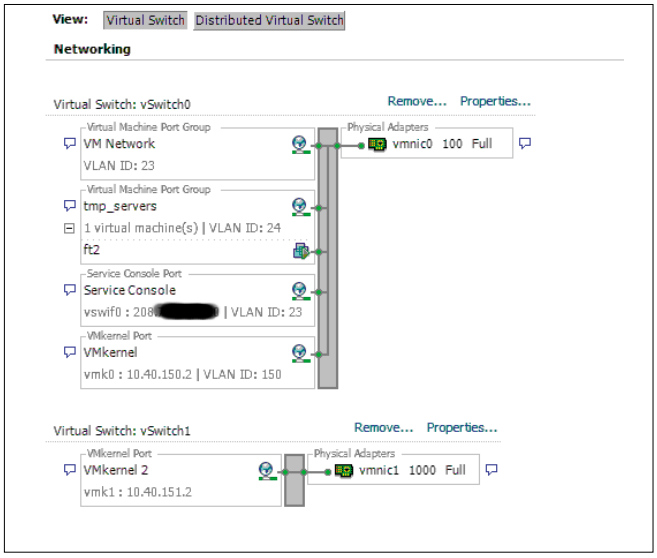
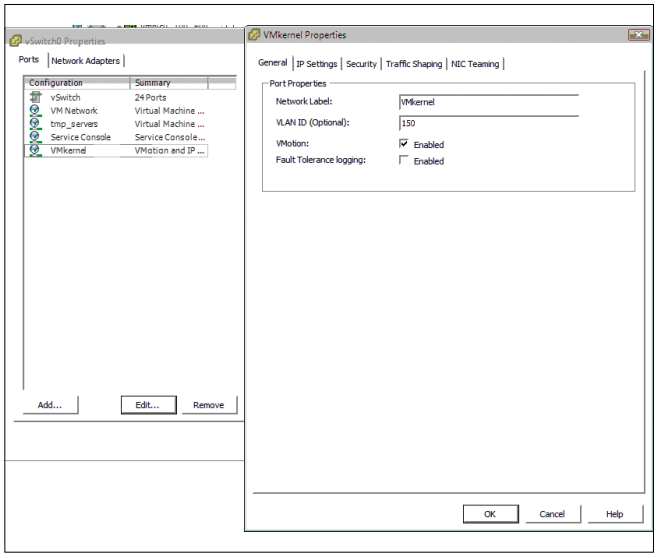


Рисунок 4. Настройка виртуальной сети для поддержки Vmotion



Далее, последовательно необходимо добавить остальные серверы VMware.

Настройка серверов VMware

Как я упомянул в предыдущей статье [1], программа VMware поддерживает «Виртуальный коммутатор», на котором могут быть созданы виртуальные сети, как указано в таблице.

Для включения поддержки VMware HA и VMware FT настоятельно рекомендуется создать три виртуальных коммутатора и привязать их к выделенным сетевым адаптерам, потому что эти функционалы потребляют значительное количество ресурсов. В моем случае трафик виртуальных машин составлял всего несколько мегабит в секунду, поэтому я создал два виртуальных коммутатора и ассоциировал их к двум сетевым адаптерам.

Как видно из рис. 3, наши виртуальные коммутаторы, кроме виртуальных сетей для управления и виртуальных машин, содержат виртуальную сеть VMKernel, причем я использовал VMKernel для Vmotion, и VMKernel2 для Fault Tolerant (см. рис. 4 и 5).

Создание виртуальной машины и тестирование

На данном этапе у нас создан кластер с поддержкой VMware. Для тестирования работы попробуем создать виртуальную машину и проверим, как работает ее запуск с запасного сервера VMware в случае краха активной копии.

Процесс создания виртуальной машины производится так же, как я описывал в предыдущей статье, но с небольшой поправкой – она должна быть создана на кластере.

По завершении создания состояние виртуальной машины можно просмотреть в разделе Vrtual Machines.

Как видно из рис. 6, виртуальная машина запущена на сервере vmware02. Для тестирования можно отключить питание сервера vmware02, подождать некоторое время и проверить состояние снова. При этом если все правильно настроено, виртуальная машина будет запущена на сервере vmware03.

Настройка Fault Tolerance и тестирование

Для активации Fault Tolerante необходимо выделить нашу виртуальную машину и, нажав правую кнопки мышки, выбрать Turn fault tolerance on. Далее нужно подождать, пока синхронизируются виртуальные машины на обоих серверах VMware, после чего в закладке Vitual Machines можно увидеть следующую картину (см. рис. 7).

Если отключить сервер vmware03, то можно будет увидеть, что виртуальная машина продолжит работу на сервере vmware02. EOF

1. Банковский В. Тратим меньше, спим больше с VMware Sphere 4.0: установка. //Системный администратор, №7, 2009 г. – С. 46-49.
2. Официальный сайт VMware – http://vmware.com.
3. Список рекомендуемого оборудования – http://www.vmware.com/go/hcl.

Рисунок 5. Настройка виртуальной сети для поддержки Fault Tolerant

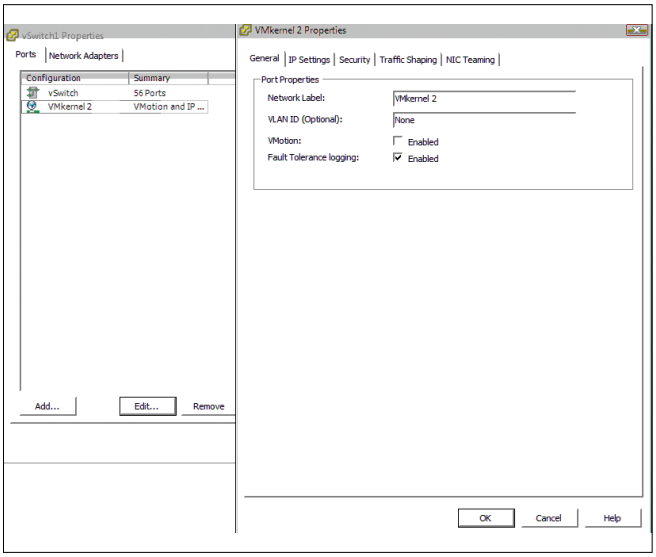


Рисунок 6. Состояния виртуальных машин

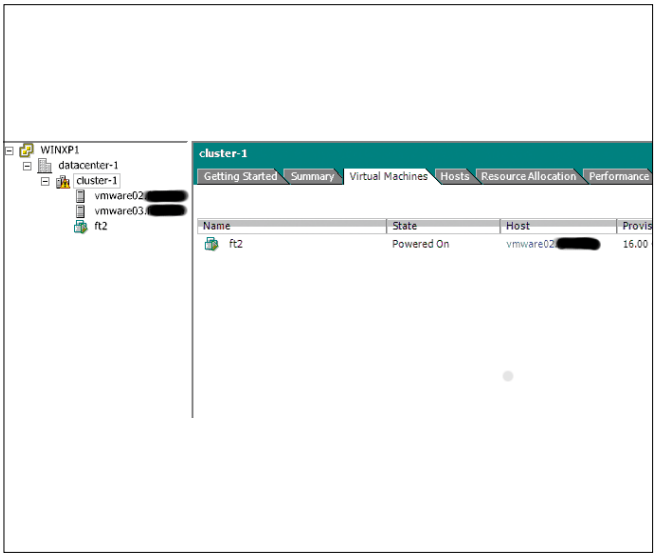


Рисунок 7. Состояния виртуальных машин

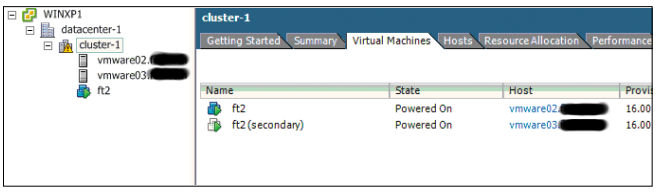


Таблица. Типы виртуальных сетей VMware

Название	Назначение
Virtual Machine	Трафик виртуальных машин
VMKernel	Поддержка функционала Vmotion и Fault Tolerant
Management Console	Для управления сервером VMware

Визитка

СЕРГЕЙ ЯРЕМЧУК, фрилансер. Автор более 800 статей и 4 книг. С «СА» с первого номера. Интересы: сетевые технологии, защита информации, свободные ОС

Настройка Webacula

Веб-интерфейс к Bacula

Bacula — популярная Open Source-система резервного копирования. Использование веб-интерфейса Webacula позволит сделать работу с ней на порядок удобнее

Bacula [1] достаточно мощная утилита резервного копирования и восстановления данных, о возможностях которой уже не раз говорилось на страницах журнала [2-4]. Тем не менее процесс установки, повседневного управления и получения отчетов недостаточно удобен и нагляден. Именно поэтому написан не один десяток графических интерфейсов к Bacula, разной степени готовности и возможностей. Одним из первых и до сих пор наиболее функциональных является Webacula (Web + Bacula) [5], о настройке которого и поговорим дальше.

Возможности Webacula

Webacula является типичным веб-приложением, требующим для своей работы стандартную связку LAMP (Linux + Apache + MySQL + PHP). Построен при помощи Zend Framework. Используя веб-браузер, администратор может удаленно запустить задание, восстановить файлы по идентификатору задания, из самой последней копии клиента, данные перед указанным временем, вывести запланированные и выполненные задания (все и в 24-часовом промежутке), задания, выполненные с ошибками (за 7 дней), вывод временной диаграммы заданий и состояния хранилищ, монтирование и размонтирование хранилищ, состояние томов. Возможно транслирование результатов в RSS-ленту. Кроме этого, доступен журнал (хранится в базе данных Webacula), в который администратор самостоятельно записывает необходимую ему информацию (описание задания, сбои и причины и так далее). Такие записи могут содержать гиперссылки на задания, для быстрого перехода.

Интерфейс Webacula переведен на шесть языков, в списке есть и русский. Основные номера версий совпадают с Bacula, текущей на момент написания статьи является 3.1. Распространяется на условиях GPLv3.

Установка Webacula в Ubuntu

В статье буду описывать установку Webacula на Ubuntu 8.04 LTS, хотя для других дистрибутивов общий принцип сохраняется. Для Webacula нам потребуется: веб-сервер с модулем mod_rewrite и поддержкой PHP, Zend Framework версии 1.8.3 (требует PHP 5.2.4 или выше) с установленным расширением PDO, пакет php-gd. В качестве СУБД можно использовать MySQL или PostgreSQL (поддержка появилась в текущей версии Webacula). И собственно Bacula. Разработчики рекомендуют использовать версию 3.x, но в репозитарии Ubuntu 8.04 находится более ранняя версия Bacula 2.4.2, с которой Webacula отлично срабатывается.

Устанавливаем пакеты для MySQL:

```
$ sudo apt-get install mysql-server mysql-client
```

В процессе установки должен появиться запрос на ввод пароля администратора базы данных, иначе устанавливаем его самостоятельно:

```
$ sudo mysqladmin -u root password пароль
```

По умолчанию Bacula устанавливается с поддержкой MySQL для директора (Director) и сервера хранения (Storage Daemon, SD).

```
$ sudo apt-get install bacula
```

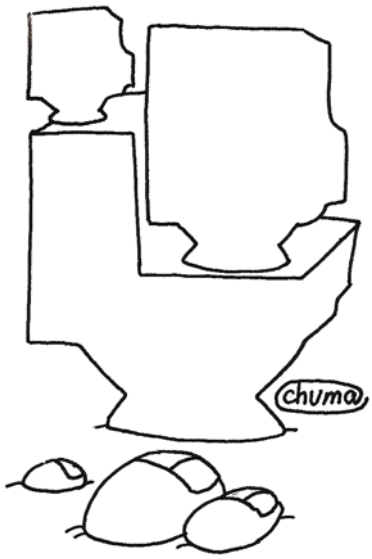
```
НОВЫЕ пакеты, которые будут установлены:
 bacula bacula-client bacula-director-mysql
 bacula-sd-mysql bacula-server
```

По ходу инсталляции будет задан запрос на установку имени учетной записи (по умолчанию bacula) и пароля. Соответствующая база данных создается, но в конфигурационный файл директора учетные данные пользователя не заносятся. Поэтому редактируем bacula-dir.conf.

```
$ sudo nano /etc/bacula/bacula-dir.conf

Catalog {
 Name = MyCatalog
 dbname = "bacula"; dbuser = "bacula"; ↵
        dbpassword = "baculapass"
}
# Для вывода сообщений выводимых при выполнении заданий
```

Для показа сообщений, которые выводятся во время выполнения заданий, измените блок Messages:



Это простая и наглядная система резервного копирования и восстановления информации

```
Messages:
Messages {
Name = Standard
...
catalog = all, !skipped, !saved
}
```

Настройку остальных параметров работы Bacula рассматривать не будем. Перезапускаем демон директора.

```
$ sudo /etc/init.d/bacula-director restart
```

Утилита netstat должна показать активность трех демонов:

```
$ netstat -l | grep bacula
```

```
tcp 0 0 comp.serv:bacula-dir *:* LISTEN
tcp 0 0 comp.serv:bacula-fd *:* LISTEN
tcp 0 0 comp.serv:bacula-sd *:* LISTEN
```

С Bacula закончили.

Готовим среду для веб-панели

Веб-сервер Apache устанавливается стандартным образом.

```
$ sudo apt-get install apache2 php5 libapache2-mod-php5 ↵
       php5-mysql php5-gd
```

Активируем модуль PHP и mod_rewrite.

```
$ sudo a2enmod php5
```

```
This module already enabled.
```

```
$ sudo a2enmod rewrite
```

```
Module rewrite installed; run /etc/init.d/apache2
force-reload to enable.
```

Создаем рабочий каталог для Webacula и копируем в него файлы:

```
$ sudo mkdir /var/www/webacula
$ wget -c http://dfn.dl.sourceforge.net/project/webacula/ ↵
       webacula/3.1/webacula-3.1.rc1.tar.gz
```

После загрузки распаковываем архив в /var/www/webacula.

Я использовал svn-версию Webacula.

```
$ sudo svn co http://webacula.svn.sourceforge.net/ ↵
       svnroot/webacula/trunk/webacula /var/www/webacula
```

В Ubuntu Apache работает от имени www-data, устанавливаем владельца на файлы:

```
$ sudo chown -R www-data:www-data /var/www/webacula
```

В последней версии Webacula появился скрипт для проверки наличия необходимых компонентов.

```
$ php5 /var/www/webacula/install/check_system_requirements.php
```

```
Check System Requirements...
Current MySQL version = 5.0.51 OK
Current PHP version = 5.2.4-2ubuntu5.6 OK
php pdo installed. OK
php gd installed. OK
php xml installed. OK
php dom installed. OK
php pdo_mysql installed. OK
Warning. php pdo_pgsql extension not installed.
php-dom, php-xml installed. OK
```

Разработчики уже включили в комплект поставки Zend (находится в каталоге library), поэтому отдельно его скачивать нет необходимости (в репозитарии Ubuntu текущая версия к тому же сильно запаздывает 1.5.1).

```
$ cd /var/www/webacula/library
$ sudo tar xzvf Zend-1.8.3.tar.gz
```

Для распаковки Zend предложен скрипт, поэтому можно просто ввести:

```
$ sudo ./runme
```

Для настройки параметров подключения к базе данных используется конфигурационный файл config.ini:

```
$ sudo nano /var/www/webacula/application/config.ini

# Указываем учетную запись для подключения к базе bacula
[general]
db.adapter = PDO_MYSQL
; db.adapter = PDO_PGSQL
db.config.host = localhost
```

```
db.config.username = bacula
db.config.password = baculapass
db.config.dbname = bacula

; Часовой пояс http://www.php.net/timezones
def.timezone = "Europe/Moscow"

; Webacula попытается определить язык автоматически,
; иначе снимаем комментарий
; locale = "ru"

; Каталог для временных файлов, директор должен иметь
; права на чтение
tmpdir = "/tmp"

; Webacula, а если точнее, то веб-сервер должен иметь
; возможность запуска консоли управления bconsole.
; Предусмотрено два варианта реализации, через sudo
; и установкой прав. Подробности INSTALL, мы используем
; второй вариант
; bacula.sudo = "/usr/bin/sudo"
; Проверяем пути к файлам
bacula.bconsole = "/usr/bin/bconsole"
bacula.bconsolecmd = "-n -c /etc/bacula/bconsole.conf"

; Подключение к базе webacula
[webacula]
db.adapter = PDO_MYSQL
; db.adapter = PDO_PGSQL
db.config.host = localhost
db.config.username = wbuser
db.config.password = wbpass
db.config.dbname = webacula
email.to_admin = root@localhost
email.from = webacula@localhost
```

При установке bacula создается одноименная системная группа, проверяем ее наличие и добавляем в нее учетную запись, от которой работает веб-сервер.

```
$ sudo usermod -aG bacula www-data
$ cat /etc/group | grep bacula
```

```
bacula:x:125:www-data
```

Устанавливаем необходимые права на файлы:

```
$ sudo chown root:bacula /usr/bin/bconsole
$ sudo chmod 750 /usr/bin/bconsole
$ sudo chown root:bacula /etc/bacula/bconsole.conf
$ sudo chmod 640 /etc/bacula/bconsole.conf
```

Конфигурационный файл веб-сервера:

```
$ sudo nano /etc/apache/sites-available/webacula

Alias "/webacula" "/var/www/webacula/html"
<Directory "/var/www/webacula/html">
        Options Indexes FollowSymLinks
        AllowOverride All
        Order deny,allow
        Allow from 127.0.0.1
        Allow from 192.168.1.0/255.255.255.0
        AuthType Basic
        AuthName "Webacula"
        AuthUserFile      /etc/apache2/webacula.users
        Require valid-user
</Directory>
```

Активируем сайт.

```
$ sudo a2ensite webacula
```

```
Site webacula installed; run /etc/init.d/apache2 reload
to enable.
```

Пароль для доступа к интерфейсу:

```
$ sudo htpasswd -c /etc/apache2/webacula.users admin
```

```
New password:
Re-type new password:
Adding password for user admin
```

В корневом каталоге сайта по умолчанию создается .htaccess такого содержания:

```
$ cat /var/www/webacula/html/.htaccess

php_flag magic_quotes_gpc off
php_flag register_globals off
RewriteEngine On
RewriteBase /webacula
RewriteRule !\.(js|ico|gif|jpg|png|css)$ index.php
```

Разработчики рекомендуют увеличить значения memory_limit и max_execution_time в /etc/php5/apache/php.ini:

```
memory_limit = 32M
max_execution_time = 300
```

При больших нагрузках устанавливаем еще большие значения. После перезапускаем веб-сервер:

```
$ sudo /etc/init.d/apache2 reload
```

Рисунок 1. Проверка наличия необходимых компонентов

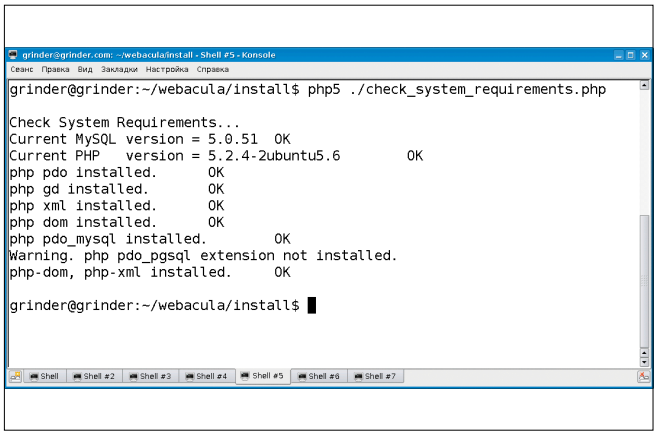
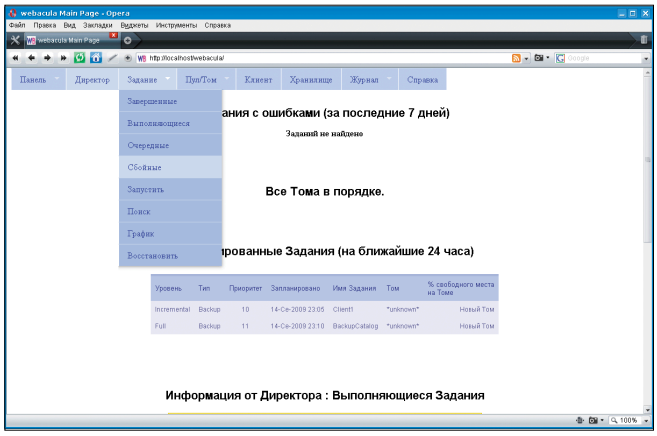


Рисунок 2. Веб-интерфейс Webacula



Проверить, загружен ли mod_rewrite, можно двумя способами. При помощи apachectl:

```
$ sudo apachectl -t -D DUMP_MODULES 2>&1 | grep rewrite

rewrite_module (shared)
```

Также для проверки работы разработчики предлагают тестовую веб-страницу, просто набираем ссылку http://localhost/webacula/test_mod_rewrite.

Теперь нужно создать учетную запись пользователя MySQL для работы Webacula и установить необходимые права:

```
$ mysql -u root -p

mysql> CREATE USER 'wbuser'@'localhost' IDENTIFIED BY ↵
        'wbpass';
mysql> GRANT ALL PRIVILEGES ON *.* TO ↵
        'webacula'@'localhost' WITH GRANT OPTION;
mysql> FLUSH PRIVILEGES;
mysql> quit
```

Для создания базы данных и таблиц разработчики предлагают скрипты, которые находятся в подкаталоге install. Перед их запуском следует указать в webacula_mysql_create_database.sh учетные данные пользователя для подключения к СУБД (для PostgreSQL другие файлы):

```
$ sudo nano /var/www/webacula/install/ ↵
        webacula_mysql_create_database.sh

db_user="wbuser"
db_password="wbpass"
host="localhost"
```

Теперь выполняем два скрипта – webacula_mysql_create_database.sh и webacula_mysql_make_tables.sh.

Перезапускаем Bacula и заходим по адресу http://localhost/webacula. Интерфейс достаточно прост, поэтому разобраться с дальнейшей работой с Webacula очень просто. В первом окне показывается список запланированных и выполненных заданий, а также заданий, выполненных с ошибками. Остальную информацию можно получить, перемещаясь по меню, расположенному в верхней части окна.

Для удаления временных файлов в каталоге tmpdir следует использовать скрипт wb_clean_tmp.sh, запускаемый при помощи cron. EOF

***

В итоге мы получили простую в использовании и наглядную систему резервного копирования и восстановления информации.

1. Сайт проекта Bacula – http://bacula.org.
2. Гринько А. Архивируем данные с помощью Bacula. //Системный администратор, №4, 2005 г. – С. 42-46.
3. Яремчук С. Обзор систем резервного копирования. //Системный администратор, №9, 2006 г. – С. 22-29.
4. Яремчук С. Полезные советы по Bacula. //Системный администратор, №11, 2006 г. – С. 53.
5. Сайт проекта Webacula – http://webacula.sourceforge.net/ru.

Визитка

ПАВЕЛ ПЛОТНИКОВ, социальный педагог. Имеет награды за продвижение современных электронных технологий в обучающий процесс. С 2008 года работает в компании «Доктор Веб», занимаясь дизайном взаимодействия ПО

Dr.Web CureNet!

От идеи до воплощения

Идея создания Dr.Web CureNet! созрела в компании «Доктор Веб» летом 2009 года. Хотелось разработать продукт для страховки и усиления безопасности корпоративных сетей, в которых есть антивирус других производителей

Наша утилита при этом задумывалось как средство, которое всегда готово прийти на помощь в критической ситуации. К примеру, когда используемый корпоративный антивирус что-то упустил, не смог справиться с какой-либо вредоносной программой. Кроме того, по нашему замыслу, лечащие сканеры Dr.Web CureNet! должны были самостоятельно распространяться по локальной сети, запускаться на рабочих станциях, а потом самоудаляться, что, как вы понимаете, очень удобно. Исходя из этого мы построили схему нового продукта:

- утилита работает без установки;
- загружается с любого носителя;
- состоит из трёх компонентов: GUI-настроек, модуля запуска и отслеживания процесса работы утилиты (сетевой агент, ищущий компьютеры в сети и доставляющий на них антивирус) и собственно антивируса;
- пользователь работает с утилитой только через графический интерфейс.

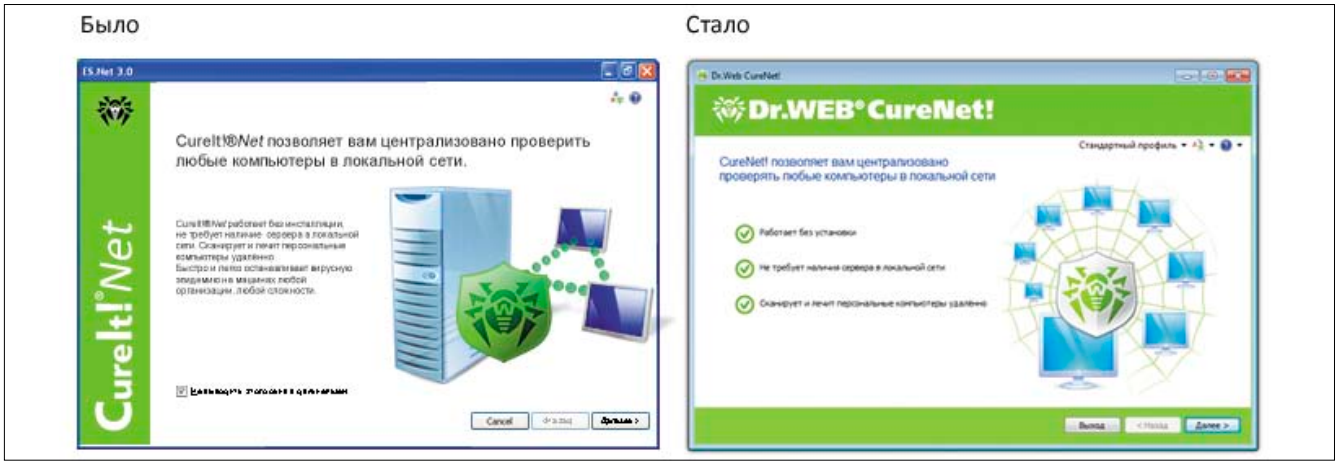
Очевидно, что у каждой компании – своя архитектура сети, свой уровень специализированной подготовки персонала. Поэтому мы должны были создать продукт, с которым было бы одинаково просто иметь дело и сетевому администратору с десятилетним стажем, и первокурснику, работающему по совместительству.

Скажете, что это нереально? Возможно, но ведь стремиться к совершенству надо всегда!

Итак, с чего мы начали. Поскольку Dr.Web CureNet! является сетевым аналогом антивирусной утилиты Dr.Web CureIt!, позволяющей провести антивирусную проверку на локальном компьютере, мы решили создать для новой утилиты преемственный, узнаваемый дизайн.

Сначала мы очень хотели уместить весь функционал программы на одной странице. Может показаться, что архитектура Мастера больше похожа на допрос, но ведь нельзя забывать и о новых пользователях, которые ранее не сталкивались с подобным интерфейсом. Именно для них мы пошли по пути создания Мастера, который дал бы опытным пользователям возможность пропустить большинство шагов и максимально быстро запустить антивирусную проверку, но при этом оставался оптимальным вариантом для начинающих. Однако об этом пойдет речь чуть ниже.

Рисунок 1. Первый и финальный дизайн окна Dr.Web CureNet!



Сейчас же расскажем о создании интерфейса выбора сканируемых станций. Как мы уже отмечали, топология сети в компаниях может быть различной: простая сеть в школьном классе и Active Directory в компании на 10 000 машин. Как предоставить пользователю возможность быстро указать те станции, которые надо просканировать? Особенно если пользователь не знает, какие станции в сети есть…

В Dr.Web CureNet! мы предоставили пользователю три возможности:

сканер сети – пользователю достаточно нажать одну кнопку, и утилита сама найдёт все сети, доступные с ПК, с которого запускается утилита;

вручную – указать IP-адрес необходимой машины или диапазон адресов (т.е. одной записью указать все машины в сети);

смешанный способ – найти автоматически, а также вручную добавить к найденному IP-адрес или диапазон адресов.

Таким образом, анализ сети анимирован, утилита не замирает, а отображает процесс поиска станций. Кроме того, все станции, найденные или добавленные вручную, автоматически отмечаются для проверки. Пользователь может снять чек-бокс напротив станций, которые не следует проверять при запуске сканирования. Ненужные пользователю для работы станции и адреса можно удалить из таблицы.

Как бы ни был процесс создания списка необходимых станций прост и нагляден, создавать его каждый раз перед сканированием будет трудоёмко. Поэтому мы ввели такую сущность, как «профили», благодаря которым пользователь может сохранить все настройки. Даже если пользователь ввел в утилиту пароли для доступа к станциям, Профиль шифруется и требует введения пароля. Таким образом, однажды создав несколько профилей, пользователь может использовать Dr.Web CureNet! в различных организациях, не затрачивая каждый раз своё время и нервы.

Собственно, больше от пользователя ничего и не нужно. Осталось выбрать тип сканирования (по умолчанию уже отмечен самый часто используемый режим – «Быстрая проверка») и нажать кнопку «Старт».

Процесс доставки антивируса на станции и сканирования анимирован. Отображается как информация, общая по всем станциям, так и конкретные данные по каждому объекту.

Отчёт о сканировании сохраняется в формате xml и доступен для просмотра в любом современном браузере. Он также снабжён функцией фильтрации и перехода на сайт компании «Доктор Веб» для получения подробной информации о функционале найденных угроз. Мы старались сделать отчёт не только удобным и информативным, но и в некотором роде внушительным.

В процессе работы над утилитой мы меняли её интерфейс три раза. Оставляя общую концепцию и функционал, мы пытались максимально упростить внешний вид утилиты. Избавиться от лишнего использования шрифтов, цветов, размеров. Сделать главные кнопки видными сразу и, наоборот, спрятать настройки, нежелательные для изменения. Чётко и недвусмысленно отображать навигацию по страницам Мастера.

Например, мы убрали информацию о владельце лицензии в окно «О программе», а на первую страницу поместили функционал выбора текущего профиля, так как в ходе тестирования оказалось, что пользователи просто не видят «Профили» при другом расположении.

Иллюстрации и иконки создавались, переделывались, улучшались на протяжении всего процесса работы над программой. Надеемся, вам также понравятся такие приятные мелочи, как, например, подкрашивание цветом полей ввода при несовпадении текстов или подрагивание окна пароля при неверном вводе.

Нельзя не отметить, что мы гордимся минимальным количеством окон подтверждения. Везде, где возможно, мы выводим предупреждающую информацию в самом окне программы, и там подробно описываем ситуацию и тут же приводим способы исправления ошибки.

Что ждет пользователей дальше?

Добавление новых настроек, таких как выключение ПК после окончания проверки (чтобы не выключать компьютеры вручную после этого), снятие значка Dr.Web CureNet! на сканируемой станции, настройки прокси-сервера для обновлений.

В перспективе нам хотелось бы отказаться от Мастера и вернуться к нашему первоначальному видению Dr.Web CureNet! как программы в одном окне. EOF

Рисунок 2. Процесс сканирования в Dr.Web CureNet!

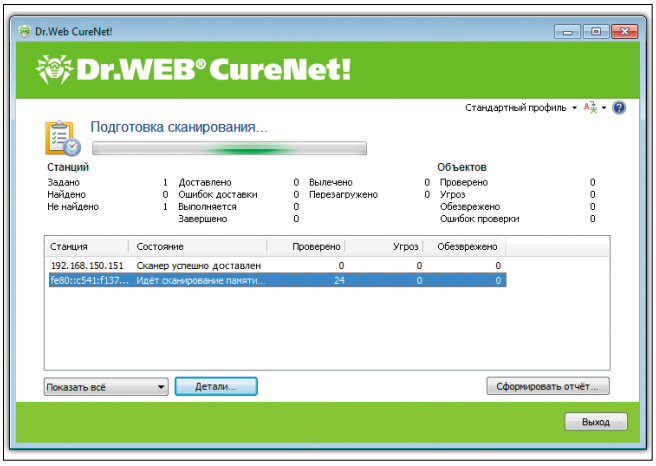
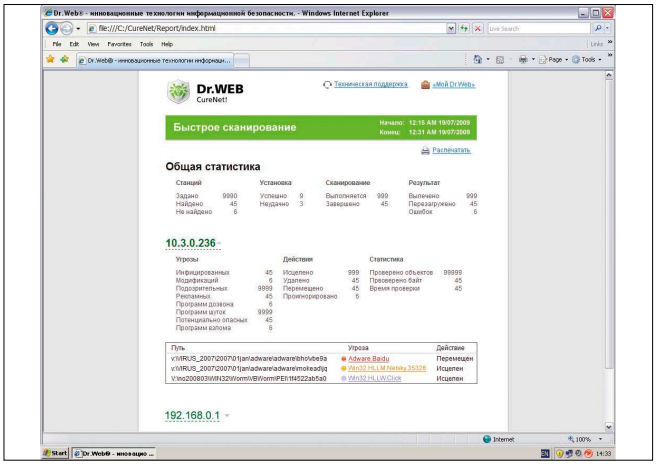


Рисунок 3. Отчет по результатам сканирования в Dr.Web CureNet!



Визитка

АРСЕН ИБРАГИМОВ, НИЯУ «МИФИ», инженер-программист, занимается теорией алгоритмов, дискретной математикой, а также разработкой приложений на C, C++ и Java

Компонентная модель EJB

Преимущества и недостатки

В российской корпоративной среде возрос интерес к платформе Java 2 Enterprise Edition (J2EE) и, в частности, к входящей в её состав компонентной модели Enterprise Java Bean (EJB)

Не стреляйте из пушки по воробьям

Я хотел бы кратко описать устройство этой модели, а также рассмотреть её преимущества, недостатки и границы применимости. Последний пункт очень часто опускается, хотя, безусловно, он чрезвычайно важен.

Выбор технологий, подходящих для реализации поставленных задач, всегда должен производиться с учётом множества факторов. Не секрет, что любое программное решение, присутствующее на рынке, имеет как краткосрочные, так и долгосрочные преимущества и недостатки. То, что на начальном этапе реализации может быть главным достоинством выбранной технологии, часто становится её основным и к тому же трудно устранимым недостатком при дальнейшем расширении создаваемого продукта. Среди факторов выбора следует особенно выделить:

- необходимость присутствия или отсутствия дальнейшей расширяемости приложения;
- упрощение разработки бизнес-логики при увеличении размеров и сложности приложения в течение всего его жизненного цикла (усложнения, которые потребуют больших усилий на начальных этапах разработки кода, могут привести к более простой его доработке в будущем);
- затраты на техническую поддержку и устранение неисправностей.

Оценка всех этих факторов должна производиться комплексно. Не стоит, как порой делают многие компании, стрелять из пушки по воробьям. Однако также не стоит и экономить на том, что может пригодиться в ближайшем будущем. Всегда нужно учитывать, что решения уровня предприятия чрезвычайно сложно перестраивать, если неправильно выбрать первоначальную архитектуру.

С чего начиналось?

Когда Интернет перестал полностью состоять из статического HTML, в нем появилось динамическое содержимое, весь динамизм обеспечивался с использованием интерфейса CGI, а впоследствии – интерпретируемыми языками, среди которых особенно выделялись Perl, PHP и ASP.

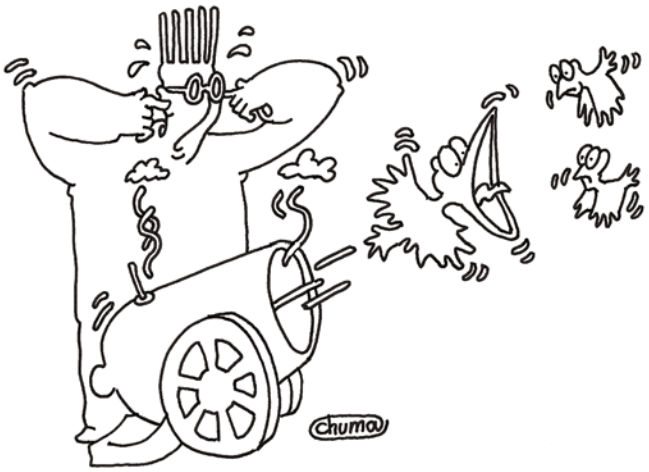
Архитектура такого рода динамических приложений (см. рис. 1) была трудно расширяемой, достаточно скудной в планах разделения задач разработчика и дизайнера, возможностей повторного использования кода. Кроме того, она не обеспечивала стандартных средств для лёгкого выполнения множества рутинных операций, которые совершенно не были связаны с бизнес-логикой, но отнимали массу сил и времени разработчиков.

Принцип работы описанной выше архитектуры читателям хорошо известен. Пользователь, желающий получить доступ к некоторой информации, запрашивает веб-страницу (точнее, веб-скрипт), которая в свою очередь напрямую обращается к базе данных и другим источникам для получения динамической информации. После этого все данные записываются на языке HTML и полученный документ передаётся браузеру пользователя.

Указанные выше недостатки не кажутся значительными в случае приложений малого масштаба, но для корпоративной среды, где необходимы гибкая масштабируемость, динамический модульный подход и быстрота разработки сложного кода, этот путь нельзя назвать практичным. Чем более функциональным и сложным будет готовое приложение в результате своей эволюции, тем труднее станет его поддерживать и расширять. По сути, в этом случае возникает та же самая проблема со сложностью, которая когда-то привела к возникновению объектно ориентированного программирования в противоположность процедурному, однако здесь она проявляется на более высоком уровне абстракции – не на уровне объектов, а на уровне модулей.

Многоуровневая структура приложения

Если принять во внимание текст предыдущего раздела, мы приходим к мысли о том, что модульная архитектура является наиболее приемлемой для сложных приложений бизнес-уровня. Кроме того, в любом таком приложении всегда можно выделить как минимум три основных логических уровня: уровень представления, уровень бизнес-логики и уровень данных (см. рис. 2). Как минимум, потому что для обеспечения большей масштабируемости вышеназ-



Нужно учитывать, что решения уровня предприятия сложно перестраивать, если неправильно выбрать архитектуру

ванные уровни могут быть разделены ещё на несколько частей. Рассмотрим теперь каждый уровень подробнее.

Уровень представления – как следует из его названия, отвечает за представление данных пользователю. Одни и те же данные могут иметь различное визуальное представление (таблица, диаграмма и т.д.). Таким образом, заменяя уровень представления, мы можем полностью изменить вид и способ работы пользователя с приложением. При этом имеющиеся функциональные возможности приложения остаются без изменения. Кроме того, уровень представления может быть одновременно реализован в различных видах (толстый, тонкий клиенты) для возможностей расширенной обработки данных различными пользователями и т.п.

Уровень бизнес-логики – по сути, содержит в себе всю функциональную часть приложения. Таким образом, функциональность приложения расширяется добавлением к этому уровню новых возможностей. При этом не нужно для каждого существующего представления заново реализовывать бизнес-логику, как это было до разделения уровня представления и уровня бизнес-логики. Всё это, несмотря на некоторое технологическое усложнение, вызванное абстракцией уровней и необходимостью взаимодействия между ними, приводит к ощутимому увеличению производительности и надёжности за счёт отсутствия повторных реализаций функциональности и совокупного действия других факторов.

Уровень данных – это структурированное хранилище данных. Обычно для крупномасштабных приложений он представляет из себя одну из множества корпоративных СУБД, например Oracle, но могут применяться и другие способы хранения (бинарные файлы, XML и т.п.), однако они используются редко. Уровень данных не имеет никакого отношения ни к представлению данных пользователю, ни к возможностям их обработки. Все что он умеет – хранить и извлекать данные, необходимые уровню бизнес-логики.

Может возникнуть вопрос: почему нельзя просто объединить все функции, отвечающие за бизнес-логику, в отдельную библиотеку и таким образом добиться отделения кода бизнес-логики от кода представления? Все не так просто, как кажется на первый взгляд. Например, таким путём не-

Рисунок 1. Двухуровневая модель приложения

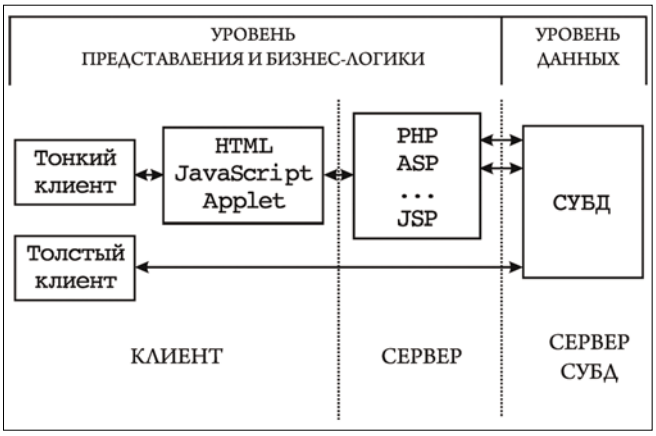
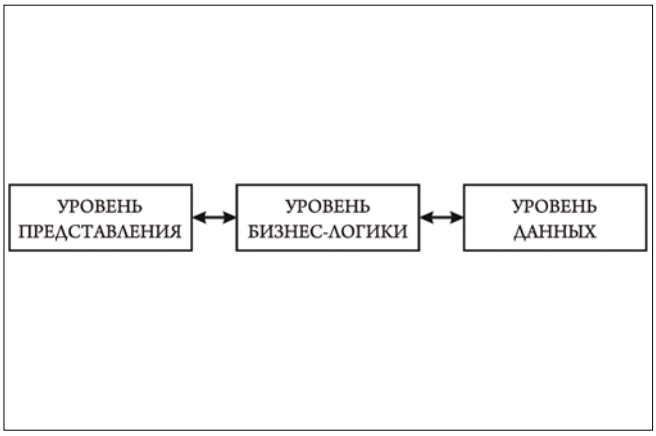


Рисунок 2. Трёхуровневая модель приложения



возможно добиться лёгкой масштабируемости, а кроме того, многие дополнительные задачи наподобие создания пула соединений с базой данных и управления параллельной обработкой данных придётся заново реализовывать в каждом новом приложении. Таким образом, выделив второстепенные (то есть не включающие саму бизнес-логику) задачи, необходимые для функционирования уровня бизнес-логики, и реализовав их однажды в виде сервера приложений, можно добиться стабильности, надёжности, безопасности, а также лёгкой расширяемости приложения. Это происходит за счёт того, что прикладной программист может сконцентрировать внимание только на бизнес-логике приложения, ведь второстепенная работа уже сделана за него разработчиками сервера приложений.

Организация трёхуровневой модели приложения в J2EE

Рассмотрим теперь, каким образом архитектура J2EE организует каждый уровень трёхуровневой модели приложения. Принципиальная схема, иллюстрирующая это, представлена на рис. 3. Из неё видно, что уровень представления реализуется либо толстым клиентом, либо связкой тонкого клиента, HTML, JavaScript и Applet на стороне клиента и JSP, Servlet на стороне сервера. Уровень бизнес-логики реализуется компонентами EJB, взаимодействие которых с уровнем представления производится только через Home и Remote интерфейсы, а уровень данных – одной из имеющихся на рынке баз данных, доступ к которой осуществляется при помощи Java Database Connectivity (JDBC).

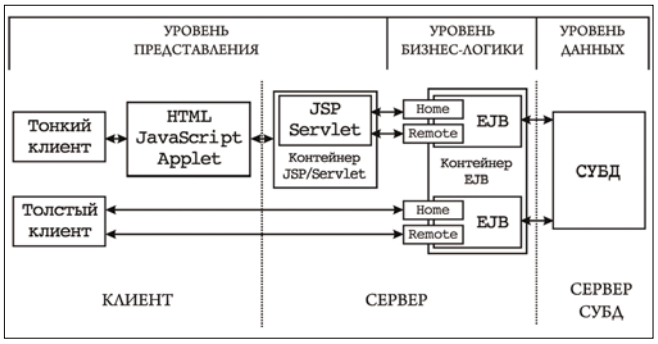
Не буду останавливаться на устройстве JSP, Applet и Servlet, а тем более HTML и JavaScript, так как, во-первых, не хочу отступать от основной темы обсуждения, а во-вторых, думаю, что читающие данную статью, скорее всего, работали с этими технологиями. Аналогично я не буду рассматривать уровень данных, так как он напрямую не имеет отношения к J2EE. Корпоративным разработчикам всего лишь нужно приобрести и настроить одну из имеющихся на рынке СУБД. Таким образом, всё дальнейшее изложение будет посвящено EJB и уровню бизнес-логики.

Уровень бизнес-логики в J2EE

Уровень бизнес-логики в J2EE реализуется с использованием компонентов EJB. Однако каждый такой компонент взаимодействует с уровнем представления не напрямую, а с помощью домашнего (Home) и удалённого (Remote) интерфейсов. Назначением функций первого является создание, удаление и поиск объектов EJB, функции же второго как раз и составляют бизнес-логику приложения.

EJB, как и любая другая технология, имеет свои преимущества и недостатки. Хотел бы подробно рассмотреть основные из них, чтобы помочь читателю сделать выбор: стоит ему использовать EJB-компоненты в своём приложении или стоит отказаться от них в пользу другой, более простой по архитектуре технологии.

Рисунок 3. Организация трёхуровневой модели приложения в J2EE



Основные преимущества EJB

Наличие открытой спецификации. Спецификация определяет все основные аспекты разработки и реализации EJB-компонентов, начиная от используемых типов данных и заканчивая распределением ролей между разработчиками. Такая подробная спецификация служит одной-единственной цели – возможности полной взаимозаменяемости серверов приложений и контейнеров различных поставщиков. В идеале должна существовать возможность взять любой такой сервер и развернуть на нем разработанное приложение без каких-либо изменений. К сожалению, на практике с этим обычно возникают проблемы, связанные как с некоторыми неточностями самой спецификации, так и с неточностями её реализации.

Переносимость компонентов. Компоненты, как впрочем, и серверы приложений написаны на языке Java, что даёт им полную переносимость на все платформы, для которых существуют Java Virtual Machine (JVM). Этот аспект не имеет отношения к спецификации EJB напрямую, но всё же является неоспоримым преимуществом.

Полная интеграция с другими технологиями Java. Этот аспект позволяет говорить о EJB не как об отдельной, обособленной технологии, а как о части платформы J2EE, являющейся полномасштабным решением практически любой задачи, которая ставится перед современным прикладным программистом. Кроме EJB платформа J2EE включает в себя такие технологии, как Servlet, Java Server Pages (JSP), Java Messaging Service (JMS), Java Connectors Architecture (JCA), Java Database Connectivity (JDBC), Java Authorization and Authentication System (JAAS), Remote Method Invocation (RMI) и др. Всё это даёт единый подход к разработке всевозможных приложений.

Быстрый цикл разработки. Наличие большого количества технологий, составляющих J2EE, а также огромного количества разработчиков Java позволяет утверждать, что для большинства распространённых задач созданы библиотеки классов. Это даёт возможность повторно использовать написанный кем-то ранее код в своих приложениях.

Лёгкость разработки. При разработке EJB-компонентов большинство функций по управлению ресурсами с целью увеличения производительности перекладываются на плечи разработчиков серверов и контейнеров приложений EJB. Все эти функции реализуются на основе достаточно сложных алгоритмов, так как должны обеспечить отличное функционирование как при повседневных, так и при пиковых нагрузках. Это позволяет прикладному программисту сосредоточиться на бизнес-логике приложения и к тому же сокращает время разработки.

Масштабируемость. Серверы приложений позволяют на лету добавлять в приложение новые компоненты EJB, а также приостанавливать и прекращать работу старых.

Это позволяет легко масштабировать приложения в соответствии с текущими потребностями.

Доступ к системам управления ресурсами. Серверы приложений включают в себя целый спектр решений, связанных с управлением ресурсами: системы управления транзакциями, безопасностью, временем жизни компонентов, системными ресурсами, Java Naming and Directory Interface (JNDI).

Компонентность. Технология EJB основана на компонентах, а значит, не обязательно разрабатывать всё приложение самостоятельно. Можно купить или использовать свободно распространяемые компоненты EJB, реализующие часть (или даже все) функций разрабатываемого приложения.

Возможность конфигурирования без перекомпиляции. Каждый компонент EJB сопровождается дескриптором развёртывания, который содержит конфигурационную информацию для развёртывания приложения. Это позволяет производить переконфигурирование приложения без изменения кода и перекомпиляции самих компонентов.

Распространённость. Технология EJB поддерживается почти всеми крупными ИТ-компаниями, что говорит о её стабильности и перспективности, а кроме того, гарантирует, что данная технология будет поддерживаться ещё в течение достаточно длительного периода, а не канет в Лету в один момент.

Основные недостатки EJB

Сложность изучения. Это, пожалуй, один из главных недостатков, останавливающих многие компании от перехода к EJB. Мало того что спецификация самих EJB достаточно объёмна, она ещё основывается или вплотную связана с другими J2EE-технологиями, поэтому для понимания структуры EJB нужно как минимум знать JNDI, RMI и JDBC.

Сложность архитектуры. Здесь имеется в виду сложность по сравнению с другим Java-кодом. Например, для создания одного EJB придётся написать как минимум два интерфейса, один класс и один дескриптор развёртывания, а потом скомпоновать их в JAR-архив. Это создаёт некоторые сложности для отладки приложения и увеличивает время его разработки, хотя и не всегда.

Возможность неоправданного усложнения приложения. Это в большей степени не недостаток EJB, а проблема плохой квалифицированности специалистов, принимающих решение об использовании той или иной платформы для разрабатываемого приложения. Иными словами, если архитектор не до конца понимает все тонкости спецификации EJB, он может неэффективно её использовать. Кроме того, есть риск, что какая-то часть приложения будет дублировать уже реализованные возможности сервера приложения, или, что ещё хуже, EJB будет использоваться там, где и сейчас, и в будущем вполне достаточно таких технологий, как Servlet и JSP.

Быстрое изменение спецификации. Как и любая другая динамично развивающаяся технология, EJB подвержена постоянным, порой очень серьёзным, изменениям. Это опять-таки усиливает роль первых трёх недостатков, так как разработчикам постоянно приходится переучиваться (доучиваться) и принимать во внимание всё новые возможности серверов приложений.

Обилие серверов приложений. Хотя спецификация и призвана как можно полнее описать технологию EJB, однако вследствие различий в реализации каждый сервер приложений имеет некоторые особенности, которые должны учитываться при проектировании и развёртывании приложений. Различия могут проявляться как в другом расположении и распределении конфигурационных файлов, так и в ошибках (неточностях) реализации, не позволяющих правильно работать конкретному приложению.

Границы применимости и возможные Java-альтернативы

Сложнее всего ответить, когда стоит, а когда не стоит применять EJB для разработки приложения. В большинстве случаев такой выбор делается на основе опыта. Более того, многое зависит от специфики конкретного приложения, имеющегося оборудования, финансовых возможностей компании и других факторов. Поэтому установить исчерпывающий критерий невозможно.

Но всё же можно дать несколько рекомендаций, позволяющих даже плохо разбирающемуся в спецификации EJB системному архитектору решить, стоит ли ему вообще рассматривать EJB как один из возможных вариантов для разрабатываемого приложения. Необходимо обратить внимание на описанные выше преимущества и недостатки. Если недостатки являются более существенными с экономической или кадровой позиции, то, скорее всего, стоит отказаться от архитектуры на основе EJB.

Использование EJB будет полностью оправданно, если приложению нужны лёгкая масштабируемость, управление транзакциями и повышенная безопасность, или если разрабатывается система, где должны соблюдаться ACID-правила (Atomicity, Consistency, Isolation, and Durability).

Не стоит применять EJB на системах с малым числом пользователей, малой загруженностью серверов, если не нужны масштабируемость, безопасность и управление транзакциями, для каталогов продукции и прочих систем, где большинство пользователей не могут изменять информацию, так как это будет совершенно избыточным. Также, по описанным выше причинам, не стоит браться за построение систем на основе EJB, если вы не имеете достаточного опыта в этой области.

Всегда следует учитывать, что компонентная модель EJB разрабатывалась для масштабных, распределённых приложений уровня предприятия, рассчитанных на достаточно сильные нагрузки, вызванные большим количеством обращений клиентов. Она необходима и в динамических приложениях, где важную роль играет лёгкая и быстрая масштабируемость. Для простых приложений, которые не требуют много ресурсов и не будут расширяться в ближайшем будущем, использовать EJB не стоит. В этом случае значительно дешевле и эффективнее обратить своё внимание на JSP/Servlet и реализовать с их помощью, например, хорошо зарекомендовавшую себя Model 2.

Стоит также ознакомиться с классическим документом от компании Sun: Designing Enterprise Applications with the J2EE Platform, находящимся по адресу: http://java.sun.com/blueprints/guidelines/designing_enterprise_applications/index.html. EOF

Повреждение памяти в Opera

Программа: Opera версии 9.64 и 10.01, возможно, другие версии.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки при выделении пространства для чисел с плавающей запятой. Удаленный пользователь может с помощью специально сформированного веб-сайта вызвать повреждение памяти и выполнить произвольный код на целевой системе.
URL производителя: www.opera.com.
Решение: Установите последнюю версию 10.10 с сайта производителя.

Множественные уязвимости в Wireshark

Программа: Wireshark версии до 1.0.10 и 1.2.3.
Опасность: Средняя.
Наличие эксплоита: Нет.
Описание: 1. Уязвимость существует из-за ошибки в функции dissect_paltalk() в файле epan/dissectors/packet-paltalk.c диссектора Paltalk. Удаленный пользователь может произвести DoS-атаку. Уязвимость распространяется на версии 1.2.0 по 1.2.2.

2. Уязвимость существует из-за ошибки разыменования нулевого указателя в диссекторе DCERPC/NT. Удаленный пользователь может аварийно завершить работу приложения. Уязвимость распространяется на версии 0.10.10 по 1.2.2.

3. Уязвимость существует из-за ошибки завышения на единицу в функции dissect_negprot_response() в файле epan/dissectors/packet-smb.c диссектора SMB. Удаленный пользователь может аварийно завершить работу приложения. Уязвимость распространяется на версии 1.2.0 по 1.2.2.

4. Уязвимость существует из-за ошибки в диссекторе RADIUS. Удаленный пользователь может аварийно завершить работу приложения. Уязвимость распространяется на версии 0.10.13 по 1.0.9.
URL производителя: www.wireshark.org.
Решение: Установите последнюю версию 1.0.10 или 1.2.3 с сайта производителя.

Уязвимость в драйвере SDP-протокола в Sun Solaris

Программа: Sun Solaris 10.
Опасность: Средняя .
Наличие эксплоита: Нет.
Описание: Уязвимость существует из-за ошибки в Sockets Direct Protocol (SDP)-драйвере sdp(7D). Удаленный пользователь может потребить всю доступную память на системе..
URL производителя: www.sun.com.
Решение: Установите исправление с сайта производителя.

Выполнение произвольного кода в Microsoft Internet Explorer

Программа: Microsoft Internet Explorer 6.x, 7.x.
Опасность: Критическая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки в Microsoft HTML Viewer (mshtml.dll) при обработке определенных CSS-объектов в методе getElementsByTagName(). Удаленный пользователь может с помощью специально сформированного веб-сайта выполнить произвольный код на целевой системе.

Уязвимость активно эксплуатируется злоумышленниками в настоящее время.
URL производителя: www.microsoft.com.
Решение: В настоящее время способов устранения уязвимости не существует.

Уязвимость при обработке встроенных OpenType-шрифтов в Microsoft Windows

Программа: Microsoft Windows 2000; Microsoft Windows XP; Microsoft Windows 2003.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки при обработке кодов Embedded OpenType (EOT)-шрифтов в win32k.sys. Удаленный пользователь может с помощью специально сформированного EOT-шрифта выполнить произвольный код на целевой системе.
URL производителя: www.microsoft.com.
Решение: Установите исправление с сайта производителя.

Переполнение буфера в Symantec Altiris ConsoleUtilities ActiveX-компоненте

Программа: Symantec Altiris Deployment Solution 6.9.x; Symantec Altiris Notification Server 6.0.x; Symantec Management Platform 7.0.x.
Опасность: Высокая.
Наличие эксплоита: Нет.
Описание: Уязвимость существует из-за ошибки проверки границ данных в методе RunCmd() в ConsoleUtilities ActiveX-компоненте (AeXNSConsoleUtilities.dll). Удаленный пользователь может с помощью специально сформированного веб-сайта передать слишком длинную строку уязвимому методу, вызвать переполнение буфера и выполнить произвольный код на целевой системе.
URL производителя: www.symantec.com.
Решение: Установите исправление с сайта производителя.

Составил Александр Антипов

Визитка

АЛЕКСАНДР МАЙОРОВ, программист @MAIL.RU, семь лет занимается веб-разработкой, три из котрых посвятил программированию под мобильные устройства

Создаем свой YouTube

Как обработать видео средствами веб-сервера

Каждый веб-разработчик должен иметь представление о том, как можно обработать видео средствами веб-сервера и автоматизировать конвертирование из одного формата в другой

С чего начать?

Существует несколько инструментов для работы с видео, среди которых наибольшее распространение получили Mencoder, Transcoder и FFmpeg.

Мы рассмотрим работу с FFmpeg, так как он наиболее удобен, является кроссплатформенным и обладает всем необходимым функционалом для работы с видео в WEB, а главное, у него есть API для удобной работы из языка программирования PHP, поставляемый в виде расширения. Мощь данному инструменту придает библиотека libavcodec. Родословные почти всех открытых программ для работы с медиа восходят к этим двум компонентам.

Mencoder – часть медиаплеера Mplayer и DVD-риппера AcidRip. Transcode используется в таком известном приложении, как dvd::rip. И Mencoder и Transcode тоже основаны на использовании библиотеки libavcodec.

Немного слов о FFmpeg

FFmpeg – это набор библиотек с открытым исходным кодом, которые позволяют работать (записывать, конвертировать и транслировать) аудио и видео в различных форматах. Сам набор включает в себя следующие компоненты:

ffmpeg – утилита командной строки для конвертирования формата видеофайла. С её помощью можно также захватывать видео в реальном времени с TV-карты;

ffserver – HTTP (RTSP в настоящее время разрабатывается) потоковый сервер для видео/радиовещания;

ffplay – простой медиаплеер, основанный на SDL и библиотеках FFmpeg;

libavcodec – библиотека, в которой содержатся все аудио/видеокодеки. Большинство кодеков были разработаны «с нуля» для обеспечения наилучшей производительности;

libavformat – библиотека мультиплексирования и демультиплексирования в медиаконтейнер;

libavutil – вспомогательная библиотека, содержащая стандартные, общие подпрограммы для различных компонентов ffmpeg. Включает в себя: adler32, crc, md5, sha1, lzo-декомпрессор, Base64-кодер/декодер, des-шифратор/дешифратор rc4-шифратор/дешифратор и aes-шифратор/дешифратор;

libpostproc – библиотека, содержащая стандартные подпрограммы обработки видео;

libswscale – библиотека, предназначенная для масштабирования видео.

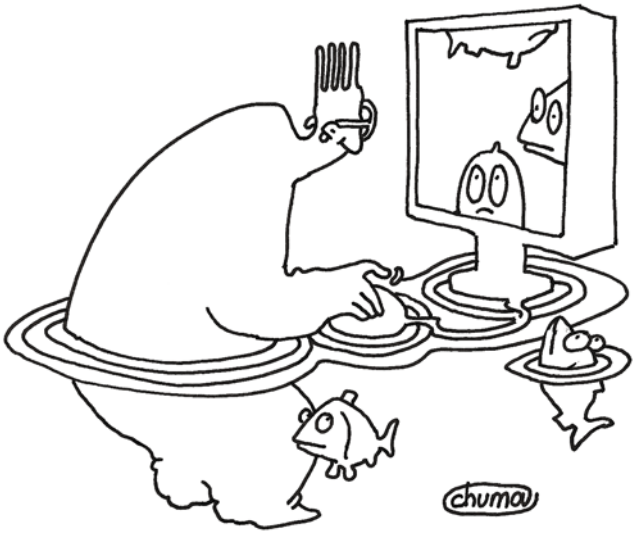
libavfilter – является заменой vhook, которая позволяет изменять видеопоток между энкодером и декодером на лету.

Название самого проекта FFmpeg происходит от названия экспертной группы «MPEG» и «FF», означающего «fast forward». Данный проект основал Фабрис Беллар (известный под псевдонимом «Gerard Lantau»). Фабрис Беллар (фр. Fabrice Bellard) – французский программист, автор ряда известных проектов в сфере свободного программного обеспечения, к примеру, таких как: QEMU (свободная программа для эмуляции аппаратного обеспечения различных платформ с открытым исходным кодом), Tiny C Compiler (компактный компилятор C). В настоящее время проект поддерживает Митчел Недермайер (Michael Niedermayer). Многие разработчики FFmpeg работают над проектом Mplayer. Кстати, сам FFmpeg располагается на сервере Mplayer.

FFmpeg разработан под операционные системы на основе GNU/Linux, однако может быть портирован и под многие другие ОС, в том числе и Windows. Разработчики не выпускают релизов и рекомендуют использовать последний снимок из Subversion. Распространяется FFmpeg под открытыми лицензиями GNU LGPL или GNU GPL.

FFmpeg работает со следующими кодеками:

- ATRAC3.
- H.261, H.263 и H.264.
- Intel Indeo 2 и 3.
- QDesign Music Codec 2, используемый в QuickTime до QuickTime 7.
- Sorenson 3 Codec используемый в QuickTime.
- Theora (вместе с Vorbis используется в контейнере Ogg).
- Truespeech.
- TXD.
- VP5 и VP6.



Ориентируясь на новые потребности, наш сервис будет доступен не только для обычных ПК, но и для мобильных телефонов

- Vorbis.
- Windows Media Audio.
- Некоторые Windows Media Video-кодеки, включая WMV1, WMV2 и WMV3.

Форматы, которые поддерживает FFmpeg:

- ASF, и через него оригинальную версию DivX.
- AVI.
- FLV.
- Matroska.
- MPEG transport stream.
- TXD.

Для веб-разработчиков на PHP к FFmpeg было написано расширение ffmpeg-php. Данное расширение добавляет удобный объектно-ориентированный программный интерфейс для работы с видео и аудио посредством afmpeg.

Первые шаги

Прежде чем приступить к работе с FFmpeg, для начала его надо установить. В зависимости от вашей ОС, установка, естественно, будет отличаться. Если у вас операционная система семейства GNU/Linux, то, скорее всего, вы будете устанавливать его из репозитория, через встроенный менеджер пакетов. Это самый лучший вариант, так как FFmpeg имеет очень много зависимостей, которые легко удовлетворяются менеджером пакетов в автоматическом режиме. Если у вас, к примеру, openSUSE, то вы воспользуетесь YaST'ом, если Ubuntu или Debian-подобный дистрибутив, то вы, скорее всего, будете ставить его через apt. Если у вас FreeBSD, то лучше воспользоваться системой портеджей. Во FreeBSD FFmpeg находится в /usr/ports/multimedia/ffmpeg. Далее выполняете стандартные команды конфигурации и сборки «make config && make configure && make && make install && make clean», после чего у вас будет собран и установлен FFmpeg из репозитория.

Надо заметить, что не все так просто. Многие кодеки имеют не полностью свободные лицензии. В силу этого, как правило, FFmpeg приходится скачивать со сторонних (не дистрибутивных) репозиториев. Например, для того же openSUSE с Packman. Там же и для Fedora, RedHut, CentOS. А это значит, что надо настраивать дополнительно yast/apt/yum/zypper в зависимости от вашей операционной системы.

Если все сделано правильно, то, набрав в консоли команду:

```
ffmpeg -v
```

вы должны получить сведения о версии программы и опциях сборки. Это может выглядеть примерно так:

```
FFmpeg version SVN-r20024, Copyright (c) 2000-2009
Fabrice Bellard, et al.
  configuration: --shlibdir=/usr/lib64 --prefix=/usr
--mandir=/usr/share/man --libdir=/usr/lib64 --enable-shared
--enable-libmp3lame --enable-libvorbis --enable-libtheora
--enable-libspeex --enable-libfaad --enable-libfaac
--enable-nonfree --enable-libxvid --enable-postproc
--enable-gpl --enable-x11grab --enable-libschroedinger
--enable-libdirac --enable-libgsm --enable-version3
--enable-libopencore-amrnb --enable-libopencore-amrwb
--enable-libx264 --enable-libdc1394 --enable-pthreads
  libavutil     50. 3. 0 / 50. 3. 0
  libavcodec    52.35. 0 / 52.35. 0
  libavformat   52.38. 0 / 52.38. 0
  libavdevice   52. 2. 0 / 52. 2. 0
  libswscale     0. 7. 1 /  0. 7. 1
  libpostproc   51. 2. 0 / 51. 2. 0
  built on Sep 27 2009 15:47:36, gcc: 4.3.2
  [gcc-4_3-branch revision 141291]
```

Как уже было сказано выше, для PHP-программистов существует удобный API для работы с FFmpeg, но это не значит, что работать с ним можно только через данную библиотеку. В жизни бывает так, что на целевом сервере у клиента нет возможности доставить нужные расширения для PHP и что-то переконфигурировать. Но там может стоять FFmpeg как системная утилита, доступная из консоли. Не стоит забывать, что ffmpeg-php – это всего лишь удобная обертка над консольной версией программы, поэтому нам ничто не мешает в наших программах обращаться напрямую к FFmpeg через системные вызовы. Работать в консоли с ffmpeg достаточно несложно, нужно просто знать заветные ключики и параметры. Итак, приступим.

Для начала определимся, что мы хотим получить и ради чего будем всем этим заниматься. Допустим, у нас есть боль-

шое желание сделать сервис, аналогичный YouTube и ему подобным. Ориентируясь на современные потребности и возможности технологий, наш сервис будет доступен не только для обычных персональных компьютеров, но и для мобильных телефонов. В «большом вебе» мы будем отдавать видео для просмотра в формате FLV, что позволит пользователю смотреть видео прямо в браузере. Для других устройств, в частности, мобильных телефонов, мы будем отдавать видео в формате 3GP. Так же у нас будет задача делать для всех загружаемых видеороликов превью и с последующим брендированием как самих превью файлов, так и загруженного видео. Собственно, с целями определились, теперь приступим к реализации нашего технического задания.

Начнем с полезных команд. Чтобы получить информацию о файле, достаточно его просто передать на вход программы. Входной файл задается ключом -i (от слова input):

```
ffmpeg -i 1.3gp
```

Вывод данной команды будет примерно таким (здесь опущена информация о версии FFmpeg):

```
Seems stream 0 codec frame rate differs from container
frame rate: 30060.00 (30060/1) -> 29.98 (45000/1501)
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from '1.3gp':
  Duration: 00:01:00.00, start: 0.000000, bitrate: 137 kb/s
    Stream #0.0(eng): Video: mpeg4, yuv420p, 176x144
    [PAR 1:1 DAR 11:9], 29.98 tbr, 90k tbn, 30060 tbc
    Stream #0.1(eng): Audio: libopencore_amrnb, 8000 Hz,
    1 channels, s16
  Metadata
    muxer           : avc2.0.11.1110
    muxer-jpn       : avc2.0.11.1110
At least one output file must be specified
```

Как правило, самый распространенный формат для загрузки на сервер – это AVI. Нам же надо будет еще иметь, соответственно, FLV, MP4 и 3GP, к примеру.

FLV-конвертирование

Формат FLV предназначен для потокового видео, однако существует возможность использовать его и для локального хранения и воспроизведения видео. FLV используется в Adobe Flash Player, который распространяется в качестве плагина для различных браузеров и операционных систем. Формат поддерживается многими мультимедиапроигрывателями, такими как FFmpeg и Mplayer. Так как FLV – это медиа-контейнер, а не формат, некоторые проигрыватели могут некорректно воспроизводить видео или звуковой поток при отсутствии кодеков, использованных при создании файла. FLV-формат частично проприетарный. Кодеки закрыты и защищены патентами, а сам контейнер открытый. FLV-файл представляет собой битовый поток, являющийся вариантом видеостандарта H.263. Это стандарт сжатия видео, предназначенный для передачи по каналам с низкой пропускной способностью (128 кбит/с и ниже).

Звук в FLV, как правило, закодирован в MP3. Но иногда могут использоваться Nellymoser codec, несжатое аудио или ADPCM-аудиоформат. В версии Flash Player 9 Update 3, в соответствии с внедрением Adobe формата ISO Base (MPEG-4 Part 12), добавлена поддержка AAC-аудио (профили AAC-LC, Main Profile и HE-AAC).

В общих чертах организация собственного видеохостинга происходит следующим образом: пользователь закачивает на сервер исходный файл, на сервере происходит его перекодирование в нужный формат, создается превью, которое в дальнейшем используется как заставка и минимизированная картинка. Исходный файл по желанию сохраняется или удаляется ради экономии дискового пространства.

Теперь переходим плавно от теории к практике. Чтобы сконвертировать AVI-файл в FLV-формат, достаточно вызвать FFmpeg таким образом:

```
ffmpeg -i uploaded_file.avi output_file.flv
```

В результате наших действий мы получим FLV-версию файла. Если мы хотим задать битрейт, то надо указать параметр «-b bitrate» – битрейт, параметр, определяющий качество, по умолчанию 200 кбит/с. К примеру: «-b 512k» означает, что задано качество в 512 кбит/с.

Также может быть полезной опция «-t duration» – продолжительность проигрывания и «-ss start_pos» – смещение от начала исходного видеофрагмента (можно задать как количество секунд, так и время в формате ЧЧ:ММ:СС.Д).

FFmpeg умеет по расширению имени выходного файла (output_file) определять параметры кодирования для этого файла (аудио/видео кодеки и прочее), так что обычно не приходится указывать их вручную при помощи соответствующих опций. Допустим, мы хотим для мобильной версии сайта оптимизировать файлы и убирать первые несколько секунд ролика, которые, как правило, не несут смысловой нагрузки.

```
ffmpeg -i uploaded_file.avi -ss 00:00:16.0 ↵
       -b 512k output_file.flv
```

Этой командой мы отрезаем первые 16 секунд ролика и задаем битрейт в 512 кбит/с.

3GP-конвертирование

3GP (сокращение от англ. 3rd generation (mobile) phone) – это видеофайлы для мобильных телефонов третьего поколения. Некоторые современные мобильные телефоны (не обязательно 3G) имеют функции записи и просмотра аудио и видео в формате 3GP. Этот формат – упрощённая версия ISO 14496-1 Media Format, который похож на MOV, используемый QuickTime. 3GP сохраняет видео как MPEG-4 или H.263. Аудио сохраняется в форматах AMR-NB или AAC-LC. Готовые видеоролики в формате имеют малый размер по сравнению с другими форматами видео, но, к сожалению, за счет жертвы качеством.

Чтобы произвести конвертацию файла в формат 3GP, нужно проделать почти то же самое, что и с конвертацией файла в FLV-формат, но только задать соответствующее расширение выходного файла и дополнительные настройки кодеков. Наша команда в консоли будет выглядеть так:

```
ffmpeg -i inputfile.avi -s qcif -vcodec h263 -acodec aac ↵
       -ac 1 -ar 8000 -r 25 -ab 32 -y outputfile.3gp
```

Вы можете не хранить оригинальный AVI-файл, но, если вдруг вам понадобится этот формат, вот пример обратного преобразования:

```
ffmpeg -i input_clip.3gp -f avi -vcodec xvid -acodec mp3 ↵
       -ar 22050 output_file.avi
```

Правда, надо заметить, что сжатие было с потерей качества изначально, поэтому мы получим файл в формате AVI, но качество будет таким как у 3GP.

Кодирование в MPEG-4

MPEG-4 – это международный стандарт, используемый преимущественно для сжатия цифрового аудио и видео. Он появился в 1998 году и включает в себя группу стандартов сжатия аудио и видео и смежные технологии, одобренные ISO – Международной организацией по стандартизации/IEC Moving Picture Experts Group (MPEG). Стандарт MPEG-4 в основном используется для вещания (потоковое видео), записи фильмов на компакт-диски, видеотелефонии (видеотелефон) и широковещания, в которых активно используется сжатие цифровых видео и звука.

MPEG-4 включает в себя многие функции MPEG-1, MPEG-2 и других подобных стандартов, добавляя такие функции, как поддержка языка виртуальной разметки VRML для показа 3D-объектов, объектно-ориентированные файлы, поддержка управления правами и разные типы интерактивного медиа. AAC (Advanced Audio Codec – или улучшенный аудиокодек) был стандартизован как дополнение к MPEG-2 (часть 7), был также расширен и включен в MPEG-4.

MPEG-4 всё ещё находится на стадии разработки и делится на несколько частей. Ключевыми частями стандарта MPEG-4 являются часть 2 (MPEG-4 part 2, включая Advanced Simple Profile, используемый такими кодеками как DivX, Xvid, Nero Digital и 3ivx, а также Quicktime 6) и часть 10 (MPEG-4 part 10/MPEG-4 AVC/H.264 или Advanced Video Coding, используемый такими кодеками как x264, Nero Digital AVC, Quicktime 7, а также используемый в форматах DVD следующего поколения, таких как HD DVD и Blu-ray Disc). MPEG-4 Part 10 имеет обозначение H.264. Данный формат предназначен для достижения высокой степени сжатия видеопотока при сохранении высокого качества. Соответственно, кодирование в H.264 будет осуществляться следующим образом:

```
ffmpeg -i input_file.avi -vcodec h264 -threads 0 -r 25 ↵
       -g 50 -b 500k -bt 500k -acodec mp3 -ar 44100 ↵
       -ab 64k out_file.mp4
```

Вытаскиваем саундтрек из фильма

Возможно, вы захотите позволить пользователям своего сервиса скачивать саундтреки из фильмов или просто аудиодорожки. Как это сделать? С FFmpeg проще простого. Для начала надо получить сведения о файле и определить параметры звуковой дорожки. Допустим, у нас есть видеофайл. Мы узнаем с помощью FFmpeg формат кодирования звука в требуемом файле. К примеру, звук в нашем файле идентифицируется как:

```
Stream #0.1:Audio: mp3, 44100Hz, stereo, s16, 80kb/s
```

Теперь у нас достаточно информации для извлечения звука из видеоролика. Мы просто пишем в консоли:

```
ffmpeg -i input_file.flv -vn -acodec copy soundtrack.mp3
```

Флаг -vn указывает на то, что мы не хотим работать с видео и оно нам не нужно. Команда copy указывает нашему FFmpeg на то, что следует кодировать выходной файл тем же самым кодеком, которым он раскодируется. По желанию вы можете изменить частоту дискретизации и битрейт.

FFmpeg также позволяет извлекать аудио из AVI с высоким качеством звуковой дорожки для записи, к примеру, на аудио-CD-носитель. Пример извлечения такой дорожки:

```
ffmpeg -i input_file.avi -vn -acodec pcm_s16le -ar 44100 ↵
       -ac 2 output_file.wav
```

Таким образом будет получена несжатая двухканальная аудиозапись с частотой дискретизации 44100 Гц и 16-битным качеством.

Контроль над содержимым

Допустим, у вас сервис получил свое развитие и вы стали популярны. К вам «льют» видеоролики в больших количествах, но при этом ваш сервис ориентирован на детскую аудиторию, к примеру. Вы сами решите, как применять полученные знания на практике. Итак, у нас кто-то залил мультфильм про Винни Пуха, но со своей версией озвучивания, содержащей нецензурные выражения и прочие недетские высказывания.

Вы, при модерации можете вырезать звук из данного ролика, оставив тем самым его без аудиодорожки, сообщив автору контента, что на вашем сайте запрещены ролики с ненормативной лексикой, и вы просите его перезалить с нормальным содержанием, либо ролик так и останется без аудиосодержимого. Чтобы проделать такую операцию, достаточно выполнить команду вида:

```
ffmpeg -i input_censored_file.flv -an -vcodec ↵
       copy output_censored_file.flv
```

Параметр -an аналогичен параметру -vn, только указывает на то, что мы не хотим обрабатывать звук.

Делаем превью для видео

Как мы уже говорили, на сайте нужно еще показать превью – маленькое изображение кадра, чтобы заинтересовать посетителя и дать предварительную информацию о содержимом. Все это также умеет делать FFmpeg. Достаточно указать ему, что мы хотим вытащить кадр и указать формат, в котором мы хотим сохранить его.

```
ffmpeg -i video_input_file.flv -an -ss 30 -vframes 1 ↵
       -s 340.180 -y -f mjpeg screenshot.jpg
```

Таким образом, мы создали графический файл в формате JPEG, взяв кадр на 30-й секунде.

Если нам надо сделать анимированное превью в формате GIF, то достаточно указать количество кадров, количество цветов и формат выходного файла:

```
ffmpeg -i input_file.avi -an -pix_fmt rgb24 -ss 40 ↵
       -vframes 64 -s 128.128 -loop_output 0 -f gif ↵
       -y screenshot.gif
```

Обратите внимание на -loop_output – это количество повторов. Если его не указать, то он будет по дефолту равен единице. Если указать ноль, то анимация будет бесконечной.

Если вы хотите узнать больше информации о ключах и параметрах FFmpeg, то вам достаточно набрать в консоли команду:

```
ffmpeg -h
```

либо пройтись по ссылке http://itbroadcastanddigitalcinema.com/ffmpeg_howto.html, где дана подробная расшифровка всего справочного материала.

Собираем ffmpeg-php под *nix

Сначала загружаем само расширение для PHP с сайта: http://sourceforge.net/projects/ffmpeg-php/files/ffmpeg-php.

Установка в принципе не составит особого труда. Распаковываем архив с исходными файлами командой:

```
tar -xjf ffmpeg-php.tbz2
```

Далее в директории с расширением запускаем утилиту phpize (входит в дистрибутив PHP) для сборки конфигурационного файла. Если у вас нет утилиты phpize, то вы можете установить пакет php-devel через ваш менеджер пакетов либо скачать и установить PHP с официального сайта.

```
$ cd ffmpeg-php/
$ phpize
```

Далее конфигурируем и собираем расширение:

```
$ ./configure && make
$ sudo make install
```

После сборки получаем библиотеку ffmpeg.so (для ОС Windows это будет ffmpeg.dll). Ее нужно вписать в php.ini с помощью директивы extension.

Проверить, подключен ли ffmpeg-php, можно несколькими способами. Например, посмотреть в phpinfo информацию о расширении. Если оно подключено, то вы увидите примерно следующее (см. рис. 2).

Если вы пишете консольную версию программы, то проверить информацию о ffmpeg-php можно и в консоли:

```
$ php -i | grep ffmpeg
```

Возможные проблемы

Не всегда сборка программного обеспечения проходит гладко. Бывают и проблемы. Я опишу два наиболее распространенных случая, которые случались в моей практике, и пути их решения. Один из вариантов ошибок приведен ниже:

```
/home/0xfa/ffmpeg-php/ffmpeg-php.c -fPIC -DPIC
-o .libs/ffmpeg-php.o
In file included from /home/0xfa/ffmpeg-php-0.6.0/
ffmpeg-php.c:42:
/usr/local/include/ffmpeg/avcodec.h:30:30:
error: libavutil/avutil.h: No such file or directory
/usr/local/include/ffmpeg/avcodec.h:262:5:
error: missing binary operator before token "("
/usr/local/include/ffmpeg/avcodec.h:323:5:
error: missing binary operator before token "("
/usr/local/include/ffmpeg/avcodec.h:436:5:
error: missing binary operator before token "("
/usr/local/include/ffmpeg/avcodec.h:442:5:
error: missing binary operator before token "("
In file included from /home/0xfa/ffmpeg-php-0.6.0/
ffmpeg-php.c:42:
/usr/local/include/ffmpeg/avcodec.h:817:
error: expected ':', ',', ';', '}' or '__attribute__'
before '*' token
```

У меня данная проблема встречалась на FreeBSD. При сборке FFmpeg не мог найти нужных заголовочных файлов. Данную проблему я решил тем, что скопировал /usr/local/include/libavutil в /usr/local/include/ffmpeg/libavutil.

Еще один распространенный случай:

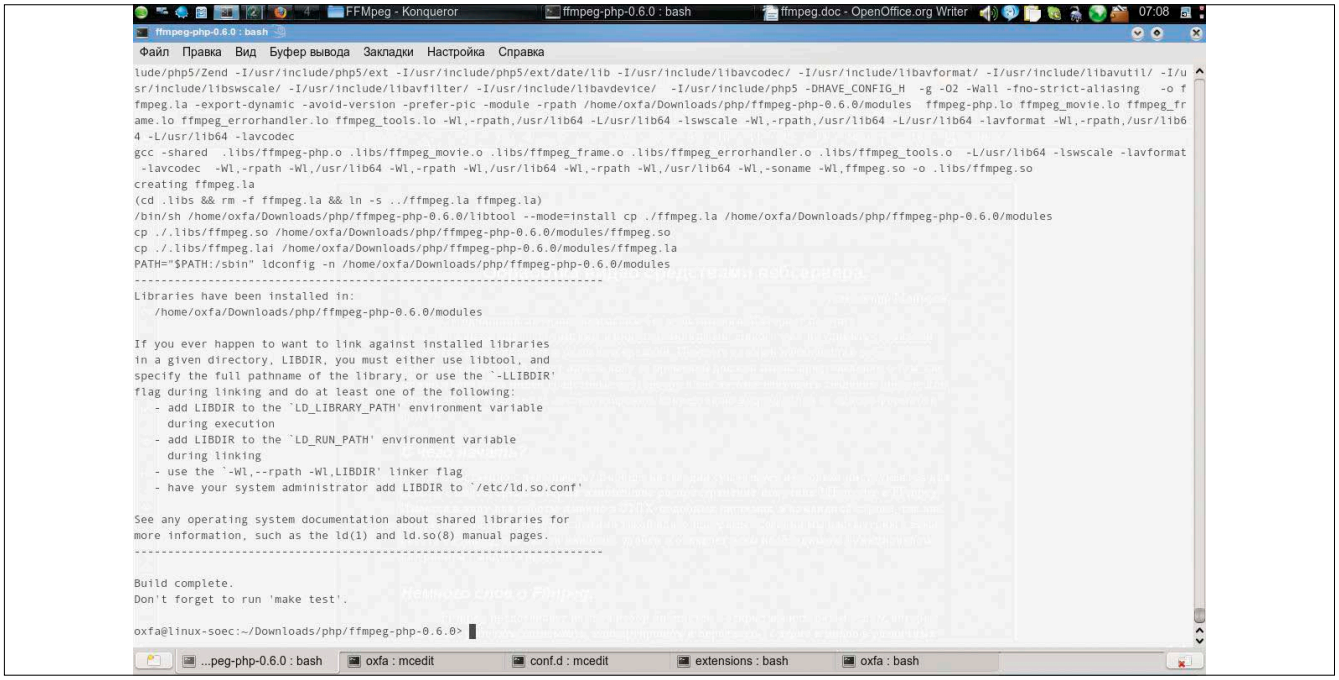
```
Exit with error ... /ffmpeg-php/ffmpeg_frame.c:498: error:
`PIX_FMT_RGBA32` undeclared (first use in this function) …
```

Лечится это правкой исходных файлов. Достаточно открыть файл ffmpeg_frame.c и сделать в нем исправления, а именно – заменить PIX_FMT_RGBA32 на PIX_FMT_RGB32. Сохраняем и пересобираем. Все должно пройти гладко.

Установка ffmpeg-php под ОС Windows

Если у вас операционная система Windows, то сборка будет немного сложнее, как это ни странно. Скорее всего я просто

Рисунок 1. Сборка ffmpeg-php



так привык к UNIX-подобным системам и отвык от Windows, что мне кажется это немного сложнее. Итак, у меня стоит операционная система Windows XP SP2. Чтобы собрать библиотеку, понадобится Microsoft Visual Studio 2005 Express Edition, которую можно взять отсюда http://www.microsoft.com/express/2005/download/default.aspx.

Также нужно установить Microsoft Platform SDK, который забираем отсюда http://www.microsoft.com/downloads/details.aspx?familyid=0baf2b35-c656-4969-ace8-e4c0c0716adb.

Еще потребуются LGPL Shared-библиотеки и заголовочные файлы FFmpeg для Windows. Их можно взять по этой ссылке http://ffmpeg.arrozcru.org/builds.

Далее нам потребуются заголовочные файлы inttype.h и stdint.h, которые можно взять здесь:

- http://msinttypes.googlecode.com/svn/trunk/inttypes.h.
- http://msinttypes.googlecode.com/svn/trunk/stdint.h.

Создайте директорию, в которой будут размещаться все необходимые файлы. У меня это C:\dev\php\ffmpeg. Распакуйте туда исходные коды PHP и ffmpeg-php, все библиотеки и заголовочные файлы.

Создайте новый проект в студии, задав тип проекта – Win32, шаблон – Win32 Project. Также скопируйте файлы из папки ffmpeg-lgpl-lshared-win32\dll и файл pthreadGC2.dll в папку system32. В качестве места расположения укажите созданную ранее директорию. В окне Application Wizard перейдите на вкладку Application Settings и установите тип приложения DLL, а Additional Options выставте в Empty Project. Жмите Finish, чтобы завершить создание проекта.

В окне Solution Explorer добавьте в раздел Header Files заголовочные файлы ffmpeg-php:

- ffmpeg_animated_gif.h;
- ffmpeg_frame.h;
- ffmpeg_movie.h;
- gd.h;
- gd_io.h;
- php_ffmpeg.h.

В раздел Source Files добавьте файлы исходных кодов ffmpeg-php:

- ffmpeg_animated_gif.c;
- ffmpeg_errorhandler.c;
- ffmpeg_frame.c;
- ffmpeg_movie.c;
- ffmpeg_php.c.

Приступим к конфигурированию проекта. Откройте меню Project и выберите пункт Properties. Перед вами появится окно настройки проекта. Откройте вкладку Configuration Properties. Выберите конфигурацию – Release. Перейдите на вкладку General, вкладкиC/C++. В поле Additional Include Directories добавьте следующие пути:

- C:\dev\php\ffmpeg;
- C:\dev\php\ffmpeg\php5;
- C:\dev\php\ffmpeg\php5\TSRM;
- C:\dev\php\ffmpeg\php5\Zend;
- C:\dev\php\ffmpeg\php5\main;
- C:\dev\php\ffmpeg\ffmpeg-include\include\ffmpeg;
- C:\Program Files\Microsoft Platform SDK\Include.

Перейдите на вкладку C/C++, раздел Preprocessor, и в поле Preprocessor Definitions добавьте следующие строки:

```
PHP_WIN32
ZEND_WIN32
ZTS=1
ZEND_DEBUG=0
HAVE_LIBGD20=1
COMPILE_DL_FFMPEG
```

Затем перейдите на вкладку Linker, далее в General. В Additional Library Directories добавьте два пути:

- C:\dev\php\ffmpeg\php5-Win32\dev;
- C:\dev\php\ffmpeg\ffmpeg-lgpl-lshared-win32\lib.

Рисунок 2. Вывод информации о ffmpeg-php в phpinfo

Далее перейдите на вкладку Linker, в Input и введите в поле Additional Dependencies:

- php5ts.lib;
- avcodec-51.lib;
- avformat-51.lib;
- avutil-49.lib.

Затем на вкладке Linker, в Command Line и в поле Additional options добавьте:

```
/FORCE:MULTIPLE
```

Перейдите на вкладку Linker, в General. В поле Output File введите:

```
$(OutDir)\ffmpeg.dll
```

Прежде чем собирать проект, нужно сделать еще пару исправлений, а именно – в файле ffmpeg-php\ffmpeg_frame.c найдите следующую строку:

```
#include "config.h"
```

и замените её на следующую запись:

```
#ifdef HAVE_CONFIG_H
#include "config.h"
#endif
```

Попробуйте собрать все это, нажав клавишу <F7>. Если в консоли появится сообщение об ошибке:

```
error C2466: cannot allocate an array of constant size 0
```

то откройте файл php\main\config.w32.h и закомментируйте строку:

```
#define _USE_32BIT_TIME_T 1
```

Попробуйте собрать расширение. На выходе должен получиться наш файл библиотеки ffmpeg\release\ffmpeg.dll. Этот файл нужно скопировать в папку с расширениями PHP и добавить в php.ini строку:

```
extension=php_ffmpeg.dll
```

На этом процесс сборки и установки расширения можно считать законченным.

Работаем с FFmpeg в PHP

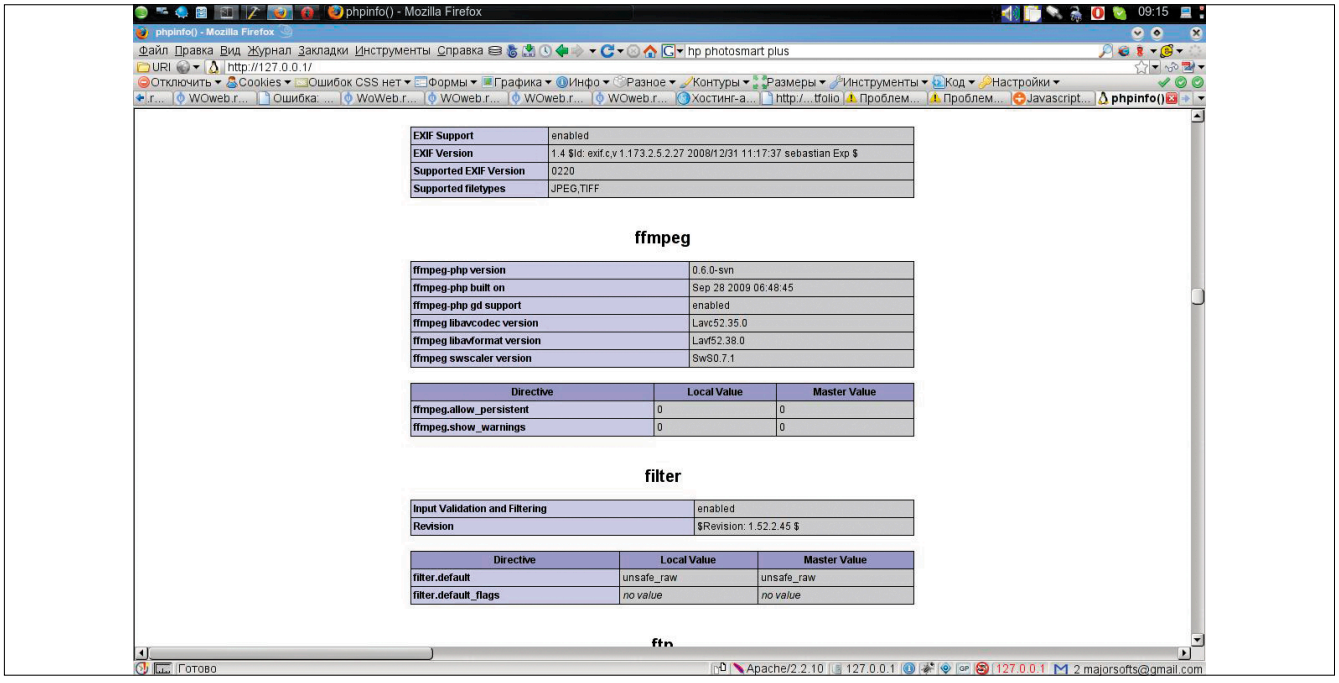
Теперь приступим к программированию. Опираясь на полученные знания, весь ваш скрипт для работы с видео может уложиться в одну строку, а именно:

```
<?php
        system($cmd);
?>
```

Где $cmd – это одна из консольных команд, что мы рассматривали выше. Так можно описать все нужные вам вызовы. Но это бывает не очень удобно для некоторых задач. Удобнее всего работать с оберткой, которая выполнена в виде расширения PHP и позволяет избавиться от вызова системных команд через консоль.

Для правильной работы ffmpeg-php необходима библиотека GD, которая позволяет работать с графическими файлами различных форматов. GD – это библиотека для динамической работы с изображениями. Она позволяет создавать изображения в форматах GIF, JPEG, PNG и WBMP. Название GD изначально обозначало «GIF Draw». Однако после аннулирования лицензии Unisys, аббревиатура расшифровывается как «Graphics Draw». GD позволяет создавать изображения, состоящие из линий, дуг, текста (включая программный выбор шрифтов) и других изображений, а также использовать различные цвета. С версии 2.0 добавлена поддержка 32-битных (truecolor) изображений, альфа-каналов, дискретизация изображений (для плавного изменения размера 32-битных

Рисунок 3. Вывод информации о ffmpeg-php в консоли



изображений) и многое другое. GD поддерживает множество языков программирования, в том числе и PHP, где библиотека значительно расширена. Начиная с версии PHP 4.3, входит в стандартную поставку интерпретатора. До этой версии могла подключаться как отдельная библиотека.

Ffmpeg-php оперирует с тремя типами объектов, а именно:

- ffmpeg_movie;
- ffmpeg_frame;
- ffmpeg_animated_gif.

Рассмотрим подробнее работу с ffmpeg_vovie. Чтобы создать объект данного типа, достаточно инициализировать его следующим образом:

```
$FfmpegMovie = new ffmpeg_movie(String path_to_media, ↵
        boolean persistent);
```

Данный код открывает аудио или видеофайл и возвращает объект. В качестве аргументов конструктора передаются: path_to_media – путь к аудио- или видеофайлу и persistent – открыть как постоянный ресурс. О постоянных ресурсах читайте в документации по PHP. После того, как будет создан объект, вам будут доступны методы для работы, которые описаны ниже.

Методы объекта ffmpeg_movie

Продолжительность аудио или видео:

```
$FfmpegMovie->getDuration();
```

Возвратит продолжительность файла в секундах.

Количество кадров:

```
$FfmpegMovie->getFrameCount();
```

Возвратит количество фреймов аудио- или видеофайла.

Частота кадров:

```
$FfmpegMovie->getFrameRate();
```

Возвратит частоту кадров видео в кадрах в секунду (fps – frame per second)

Путь к файлу:

```
$FfmpegMovie->getFilename();
```

Возвратит путь к видео- или аудиофайлу.

Поле комментария:

```
$FfmpegMovie->getComment();
```

Возвратит поле комментария из аудио- или видеофайла.

Поле заголовка:

```
$FfmpegMovie->getTitle();
```

Возвратит поле заголовка из аудио- или видеофайла.

Высота видео:

```
$FfmpegMovie->getFrameHeight();
```

Возвратит высоту видео в пикселях.

Ширина видео:

```
$FfmpegMovie->getFrameWidth();
```

Возвратит ширину видео в пикселях.

Скорость потока аудио:

```
$FfmpegMovie->getAudioBitRate();
```

Возвратит битрейт аудио у видео- или аудиофайла в битах в секунду.

Номер кадра:

```
$FfmpegMovie->getFrameNumber();
```

Возвратит текущий номер кадра.

Название видеокодека:

```
$FfmpegMovie->getVideoCodec();
```

Возвратит название видеокодека, который использован в видеофайле, как строку.

Название аудиокодека:

```
$FfmpegMovie->getAudioCodec();
```

Возвратит название аудиокодека, который использован видеофайл как строку.

Количество аудиоканалов:

```
$FfmpegMovie->getAudioChannels();
```

Возвратит количество аудиоканалов как целое число. Если 1 – это моно, если 2 – стерео, и т.д.

Кадр из видео:

```
$FfmpegMovie->getFrame([Integer framenumber]);
```

Возвратит кадр из видео как ffmpeg_frame объект. Возвратит логическую ложь (false), если кадра нет, а framenumber – это номер кадра, который надо возвратить. Если framenumber не определен, то возвращается следующий кадр из видео.

Методы объекта ffmpeg_frame

Теперь рассмотрим работу с ffmpeg_frame. Чтобы создать объект данного типа, достаточно инициализировать его следующим образом:

```
$Frame = new ffmpeg_frame(Resource gd_image);
```

Данный объект создается из ресурса GD либо возвращается некоторыми методами объекта ffmpeg_movie.

```
$Frame->getWidth();
```

Возвратит ширину кадра.

```
$Frame->getHeight();
```

Возвратит высоту кадра.

```
$Frame->resize(Integer width, Integer height ↵
        [, Integer crop_top [, Integer crop_bottom ↵
        [, Integercrop_left [, Integer crop_right ]]]]);
```

Изменяет размер и обрезает кадр. Параметры:

width – новая ширина кадра (должно быть натуральным числом);

height – новая высота кадра (должно быть натуральным числом);

croptop – удалить указанное количество рядов пикселов сверху кадра;

cropbottom – удалить указанное количество рядов пикселов снизу кадра;

cropleft – удалить указанное количество рядов пикселов слева от кадра;

cropright – удалить указанное количество рядов пикселов справа от кадра.

Обрезание всегда происходит до изменения размера кадра. Значения параметров должны быть натуральными числами!

```
$Frame->crop(Integer crop_top [, Integer crop_bottom ↵
        [, Integer crop_left [, Integer crop_right ]]]);
```

Обрезать кадр. Параметры:

croptop – удалить указанное количество рядов пикселов сверху кадра;

cropbottom – удалить указанное количество рядов пикселов снизу кадра;

cropleft – удалить указанное количество рядов пикселов слева от кадра;

cropright – удалить указанное количество рядов пикселов справа от кадра.

Замечание: параметры должны быть натуральными числами.

```
$Frame->toGDImage();
```

Возвращает truecolor GD картинку (ресурс) кадра. Функция недоступна, если нет библиотеки GD.

```
$Frame->addFrame(ffmpeg_frame frame_to_add);
```

Добавляет кадр в конец анимированного GIF. Параметры:

frame_to_add – ffmpeg_frame-объект для добавления в конец анимированного GIF.

```
$Frame->getPresentationTimestamp();
```

Возвращает время создания кадра.

Осталось рассмотреть последний тип объекта, это ffmpeg_animated_gif.

Методы объекта ffmpeg_animated_gif

Чтобы создать объект, не потребуется много усилий:

```
$GifFile = new ffmpeg_animated_gif( ↵
        String output_file_path, Integer width, ↵
        Integer height, Integer frame_rate, ↵
        [Integer loop_count])
```

Данный код создает новый ffmpeg_animated_gif-объект. Параметры, которые нужно передать:

output_file_path – путь в файловой системе, куда будет записан анимированный GIF;

width – ширина анимированного GIF;

height – высота анимированного GIF;

frame_rate – частота кадров анимированного GIF в кадрах в секунду;

loop_count – количество повторений анимации. Укажите 0 для бесконечного повторения или пропустите параметр для отключения повторений.

```
$frame->addFrame(ffmpeg_frame frame_to_add);
```

Добавляет кадр в конец анимированного файла.

frame_to_add – ffmpeg_frame-объект для добавления в конец анимированного файла.

Теперь можно рассмотреть небольшой пример создания превью с помощью ffmpeg-php:

```
<?php

    if ( $Ffmpeg = new ffmpeg_movie( $input_file ) )
        {
            if ( $Frame = $ffmpeg->getFrame( ↵
                $number_of_frame ) )
            {
                imagejpeg
                (
                    $Frame->toGDImage(),
                    $path_to_image_file,
                    $quality
                );
            }
        }

?>
```

В данном примере мы создаем объект ffmpeg_movie, затем получаем кадр с помощью метода getFrame(), и если он существует, то сохраняем его с помощью GD в JPG-формат, передав кадр в функцию imagejpeg() предварительно превратив его в ресурс GD через метод toGDImage().

Конфигурирование сервера для нашего веб-хостинга

Видеофайлы достаточно объемные, поэтому, чтобы обеспечить возможность загрузки больших файлов на сервер, нужно дополнительно сконфигурировать PHP. В глобальном файле php.ini необходимо указать следующие параметры:

Post_max_size – максимально допустимый размер POST-данных (в мегабайтах). Допустим, мы разрешаем загружать файлы размером 256 Мб, для этого мы пишем:

```
post_max_size=256M
```

Но данного параметра недостаточно. Также надо указать upload_max_filesize – максимальный размер закачиваемого файла. Данный параметр должен быть равен post_max_size.

Так как загрузка файла и обработка файлов будут занимать достаточное количество времени, следует дополнительно в скриптах через ini_set указать max_execution_time – максимальное разрешенное время выполнения скрипта (в секундах). Делается это так:

```
ini_set('max_execution_time', 9000);
```

Также надо выставить max_input_time – максимально разрешенное время (в секундах), в течение которого скрипту разрешается анализировать входные данные. Я выставил в 9000 секунд данный параметр.

Все эти параметры можно также указать через файл .htaccess для каждого корня документов или даже директории вместо изменения глобального php.ini. Делается это следующим образом:

```
php_value post_max_size 256M
php_value upload_max_filesize 256M
php_value max_execution_time 9000
php_value max_input_time 9000
```

***

В следующей статье. Мы рассмотрим с вами, как делать брендирование файлов несколькими способами, изучим способы оптимизации и снижения нагрузки. EOF

1. http://ffmpeg-php.sourceforge.net.
2. http://ffmpeg-php.sourceforge.net/doc/api.
3. http://itbroadcastanddigitalcinema.com/ffmpeg_howto.html.
4. http://ffmpeg.mplayerhq.hu/ffmpeg-doc.html.

Множественные уязвимости в Microsoft Excel

Программа: Microsoft Office XP; Microsoft Office 2003; Microsoft Office 2007; Microsoft Office 2004 for Mac; Microsoft Office 2008 for Mac; Open XML File Format Converter for Mac; Microsoft Office Excel Viewer; Microsoft Office Excel Viewer 2003; Microsoft Office Compatibility Pack for Word, Excel, and PowerPoint 2007 File Formats.
Опасность: Высокая.
Наличие эксплоита: Нет.
Описание: 1. Уязвимость существует из-за неизвестной ошибки при обработке электронных таблиц Excel. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать повреждение памяти и выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за неизвестной ошибки при обработке определенных объектов записи. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе.

3. Уязвимость существует из-за неизвестной ошибки при обработке Binary File Format (BIFF)-записей. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе.

4. Уязвимость существует из-за неизвестной ошибки при обработке формул, встроенных в ячейки. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе.

5. Уязвимость существует из-за неизвестной ошибки при загрузке Excel-формул. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать повреждение указателя и выполнить произвольный код на целевой системе.

6. Уязвимость существует из-за неизвестной ошибки при обработке Excel-записей. Удаленный пользователь может с помощью специально сформированного Excel-файла вызвать повреждение памяти и выполнить произвольный код на целевой системе.

7. Уязвимость существует из-за неизвестной ошибки при обработке объектов Excel-записей. Удаленный пользователь может с помощью специально сформированного Excel-файла выполнить произвольный код на целевой системе.

URL производителя: www.microsoft.com.
Решение: Установите исправление с сайта производителя.

Переполнение буфера в Symantec Altiris ConsoleUtilities ActiveX-компоненте

Программа: Symantec Altiris ConsoleUtilities ActiveX Control 6.x; Symantec Management Platform 7.x; Altiris Notification Server 6.x; Altiris Deployment Solution 6.x.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки проверки границ данных в методе BrowseAndSaveFile() в ConsoleUtilities ActiveX-компоненте (AeXNSConsoleUtilities.dll версия 6.0.0.1846). Удаленный пользователь может с помощью специально сформированного Web сайта передать уязвимому методу слишком длинную строку, вызвать переполнение стека и выполнить произвольный код на целевой системе.
URL производителя: www.symantec.com.
Решение: Установите исправление с сайта производителя.

Множественные уязвимости в Adobe Shockwave Player

Программа: Adobe Shockwave Player 11.5.1.601 и более ранние версии.
Опасность: Высокая.
Наличие эксплоита: Нет.
Описание: 1. Уязвимость существует из-за ошибки, относящейся к использованию некорректного индекса. Удаленный пользователь может с помощью специально сформированных Shockwave данных выполнить произвольный код на целевой системе.

2. Уязвимость существует из-за ошибки, относящейся к использованию некорректного указателя. Удаленный пользователь может с помощью специально сформированных Shockwave данных выполнить произвольный код на целевой системе.

3. Уязвимость существует из-за еще одной ошибки, относящейся к использованию некорректного указателя. Удаленный пользователь может с помощью специально сформированных Shockwave данных выполнить произвольный код на целевой системе.

4. Уязвимость существует из-за ошибки проверки границ данных при обработке длины строк. Удаленный пользователь может вызвать повреждение памяти и выполнить произвольный код на целевой системе.

Также сообщается об ошибке проверки границ данных, которая позволяет злоумышленнику вызвать повреждение памяти и вызвать отказ в обслуживании приложения.

URL производителя: http://www.adobe.com/products/shockwaveplayer.
Решение: Установите последнюю версию 11.5.2.602 с сайта производителя.

Составил Александр Антипов

ВЛАДИМИР ГАКОВ, журналист, писатель-фантаст, лектор. Окончил физфак МГУ. Работал в НИИ. С 1984 г. на творческой работе. В 1990-1991 гг. – Associate Professor, Central Michigan University. С 2003 г. преподает в Академии народного хозяйства. Автор 8 книг и более 1000 публикаций

Тотальная мобилизация

Третья попытка Пола Галвина

Редкое упорство, которое многие называли упертостью, помогло создателю компании Motorola завоевать мир

В последнее время персональный компьютер неотвратимо «перебирается» в сотовый телефон. А он с той же необратимостью обрастает таким количеством добавочных функций и опций, что о главной – средстве голосовой связи – скоро, кажется, и позабудут. Так что самое время вспомнить о компании-пионере, с которой, собственно, и началась эпоха современных «мобил».

Если кто забыл или не в курсе: это Motorola. Забавно, что компания, подарившая миру первое автомобильное радио, первую переносную рацию, первый пейджер, которая ныне является одним из мировых лидеров сотовой связи, была создана в 1928 году… «патентованным» неудачником! Создание Motorola стало третьей попыткой Пола Галвина затеять собственный бизнес – первые две привели его к банкротству…

Его фарцовое детство

Впрочем, тогда, 80 лет назад, основанная в Чикаго компания называлась не Motorola, а Galvin Manufacturing Corporation.

Ее основатель, 33-летний Пол Галвин, вряд ли мог рассчитывать на скорый успех – ведь все его прежние бизнес-проекты с удручающей регулярностью терпели крах в самом начале. Но упрямец не собирался опускать руки. Заявив жене, что он много раз падал, а потому хорошо усвоил, как следует подниматься на ноги, Галвин инвестировал оставшиеся у него $750 в то, на что этих денег как раз хватило. А именно – в оборудование для производства источников электропитания, выставленное на аукцион в счет уплаты долгов таких же неудачников, как и сам Галвин.

С помощью приобретенных «производственных мощностей» он собирался освоить выпуск домашних трансформаторов и выпрямителей (при этом выдумал для них эффектное рыночное название – battery eliminators, «приборы, делающие ненужными батарейки») и с этой целью пригласил в долю младшего брата Джозефа, который смог наскрести еще $565 «живых» денег. Они и стали первыми активами новой компании, зарегистрированной 25 сентября 1928 года. Будущей Motorola.

Детство ее основателя прошло в одном из центров американской научной мысли – в Гарварде. Сызмальства Пол ни о какой науке не думал, зато активно занимался бизнесом. Прогуливая уроки в школе, предприимчивый мальчуган торговал попкорном на вокзале (в зависимости от сезонной конъюнктуры заменяя кукурузные хлопья горячими сандвичами либо мороженым). Случались у малолетнего торговца без лицензии и стычки с «контролирующими органами» в лице начальника станции и его помощников. Но эти издержки нелегального бизнеса не остановили Пола – он с переменным успехом окучивал свою выгодную торговую площадку вплоть до поступления в вуз.

Галвин окончил Университет штата Иллинойс в 1917 году – в самый разгар Первой мировой войны, и, едва получив диплом, тут же отправился воевать в Европу. На фронт он ушел рядовым, а домой вернулся уже с капитанскими лычками. Перебиваясь временной работой на заводе в соседнем штате Висконсин, Галвин не оставлял планов заняться собственным бизнесом – тем более что теперь отставному капитану нужно было содержать собственную семью.

Но основанная в 1920 году (на пару с коллегой по заводскому цеху) фирма по производству электрических батареек спустя три года была закрыта налоговыми органами штата как злостная неплательщица. Еще быстрее прогорела аналогичная компания, созданная теми же партнерами в Чикаго, – она просуществовала всего два года.

С таким негативным опытом за плечами Пол Галвин отправился в свое третье плавание по неспокойному морю бизнеса. Но на сей раз удача улыбнулась предпринимателю, проявившему столь редкое упорство, которое многие не без оснований считали упертостью.

Корпорацией Galvin Manufacturing на первых порах называлась чисто номинально. Ее штат составляли всего пять сотрудников, из них более половины занимали «руководящие посты» – братья Галвины занимались научными исследованиями, разработками и продажами, а за менеджмент отвечала супруга Пола. Сохранилась информация о том, что за первую неделю работы все пятеро получили совокупную зар-

плату в $63. Даже по меркам 1928 года это были сущие копейки – то есть центы. И хотя за первый год объем продаж тысячекратно превысил эту сумму (ровно $63 тыс.!), чистая прибыль по-прежнему оставалась более чем скромной – всего $6 тыс.

Авторадио

Нужен был какой-то новый товар, способный обеспечить решающий прорыв на рынке. Поразмыслив, Галвин решил, что из всех новомодных технических новинок на американском рынке наибольшим успехом пользуются две – радиоприемники и автомобили. Оставалось сделать следующий логический шаг – объединить этих двух лидеров потребительского спроса, иначе говоря, радиофицировать автотранспорт. Однако все подобные попытки в 1920-е годы терпели неудачу. Причины были чисто технические – работа генератора и системы зажигания неизбежно засоряла эфир помехами, да и хрупкие радиолампы оказывались малопригодными для тогдашних грохочущих колымаг на четырех колесах. В общем, слушать радио в салоне удавалось лишь при выключенном моторе.

Поэтому когда знакомый инженер предложил Галвину создать специальное радио для автомобилей, глава Galvin Manufacturing увидел в этом предложении тот самый шанс, о котором давно мечтал. Правда, запал на идею авторадио Галвин не сразу. Сначала он решил, что даже если подобная штука технически осуществима, то ее тут же запретят – чтобы не отвлекала водителей от дороги. А во время командировки в Нью-Йорк его скепсис только усилился после того, как на глаза попалось объявление об установке в автомобиль специально переделанного переносного радиоприемника. По цене «всего» $240! Для справки – автомобиль тогда стоил примерно $650.

Для таких людей, как Галвин, неразрешимая задача – это как раз та единственная, которой и стоит заниматься. Со свойственным ему максимализмом глава компании поставил перед сотрудниками задачу вдвойне неразрешимую – создать не просто автомобильное радио, а дешевое радио, розничная цена которого не должна была превышать $200. А в идеале – укладывалась бы в сотню с хвостиком. Галвина всегда заботили стратегические, глобальные вопросы, частности – реализацию – он обычно перекладывал на плечи подчиненных.

Скоро появилась опытная модель, но для серийного запуска новинки и ее продвижения на рынок нужны были кредиты. Галвин попытался выбить их у одного чикагского банкира, и для наглядной демонстрации преимуществ своего товара предложил банкиру установить авторадио в его машине. Эта идея погорела в буквальном смысле – не успел потенциальный кредитор отъехать, как галвиновский агрегат вспыхнул, вызвав легкий пожар в салоне.

Но и это фиаско не остановило настырного Галвина. Он и впредь предпочитал тягучим переговорам эффектные демонстрации своих новинок – правда, ужесточив требования к их «предпоказному» тестированию.

Эксперимент на самом себе

К первой презентации авторадио на съезде Ассоциации производителей радиоприемников в Атлантик-Сити в 1930 году компания Галвина готовилась серьезно, работали днем и ночью. И все же, как часто бывает при подобной штурмовщине, не хватило буквально считанных дней. Рабочая модель была готова, но на ее отладку времени уже не оставалось. И тогда Галвин установил первый авторадиоприемник в собственном «студебекере», собственноручно отлаживая аппаратуру по дороге из Чикаго в Атлантик-Сити, пока жена рулила.

В частности, успешная презентация авторадио состоялась в том же 1930 году на съезде Ассоциации производителей радиоприемников в Атлантик-Сити. Демонстрация нового чуда бытовой техники – работающего авторадио по розничной цене чуть более сотни долларов! – прошла эффектнее некуда. Участники съезда с утра до вечера по одному – по двое забирались в машину к изобретателю, и он катал их по улицам города, давая возможность насладиться четкой, не забиваемой помехами радиопередачей. Многие и после этого не решались поверить в рыночные перспективы новинки, кто-то даже высказывал опасения, не нарушит ли новый агрегат работу двигателя…

И все же к последним дням работы съезда стена недоверия была сломана. Компания Галвина получила первые заказы на модель Motorola 5T71, которая продавалась по розничной

Авторадио Motorola

Армейская рация Motorola во времена Второй мировой

«Алло! Луна на проводе!»

Спустя десятилетие после смерти основателя компании Motorola, 21 июля 1969 года не только Америка – весь мир, затаив дыхание, следил за первыми шагами Нила Армстронга по поверхности Луны и слушал его исторические слова о «маленьком шаге одного человека и огромном шаге всего человечества». Эти слова без искажений долетели до земных слушателей благодаря специальной аппаратуре, установленной на лунном модуле Apollo-11 компанией Motorola.

цене $110-130. Конец года Galvin Manufacturing Corporation закончила с объемом продаж $287 тыс. Оставался, правда, небольшой убыток в $3,7 тыс. Но, во-первых, он был наследием «проклятого прошлого», а во-вторых, стал последним в истории компании, пока ею руководил Пол Галвин.

Копы на связи!

С этого момента вся продукция Galvin Manufacturing выходила под торговой маркой Motorola. Название появилось из соединения слова motion (движение) с названием популярной тогда марки домашнего радиоприемника Victorola.

Немногие, вероятно, в курсе, что прообраз современной «мобилы» был создан компанией Галвина почти три четверти века назад – в 1936-м! Тогда, правда, новомодной «трубкой» пользовались только привилегированные клиенты – чикагские полицейские. И хотя существенным недостатком первых мобильников была исключительно односторонняя связь, американские копы по достоинству оценили новинку – с ее помощью они теперь могли отдавать на своей частоте команды одновременно всем патрулям в городе. Забегая вперед, можно отметить, что в 2003 году Motorola получила специальную награду Интерпола за пионерские разработки в «мобилизации» стражей порядка.

В том же 1936 году Пол Галвин, отдыхая с семьей в Европе, обратил внимание на хваленые немецкие автобаны. В отличие от многих европейских и американских политиков, бизнесмен быстро сообразил, что идеальные дороги построены не для гражданской техники, а для военной. Вернувшись домой, он поставил перед сотрудниками новую задачу, причем, срочную – создание портативных переносных

Компания Motorola, подарившая миру первое автомобильное радио, первую переносную рацию, первый пейджер, которая ныне является одним из мировых лидеров сотовой связи, была создана в 1928 году

К середине 1930-х годов последние еще не успели превратиться в массовый товар – радио имели менее половины американских семей. Но постепенно хорошо известную и раскрученную марку Victorola начала уверенно теснить «новенькая» – Motorola. Развивая свой успех на рынке авторадио, компания Галвина вторгалась на рынок и домашних радиоприемников, предлагая потребителям новинку за новинкой – кнопочную настройку, более удобные регуляторы частоты и громкости и многое другое.

Следующим шагом стало производство переносных радиоприемников. армейских раций. Иначе говоря, найти замену агрегатам, которыми тогда были оснащены все армии мира, – громоздким, с трудом умещавшимся в рюкзаке и не позволявшим одновременно говорить и слушать. Имевшиеся в то время портативные устройства размерами с кирпич и с торчавшими во все стороны микрофоном, телефоном и длинной антенной были слабосильными – «брали» всего несколько миль.

Чутье бизнесмена не подвело Галвина и на этот раз. Спустя три года в Европе началась большая война, а в 1941-м в нее вступили и Штаты. Тут-то и подоспел желанный госзаказ на портативные и мощные армейские рации, которые тогда разрабатывала только Galvin Manufacturing. Когда год спустя правительство США запретило выпуск гражданского автотранспорта, компания Галвина, в 1943 году ставшая публичной, не только не понесла убытков, но и довела объем продаж до $80 млн, а свой штат – до символической тысячи человек.

К тому времени должность директора по перспективным исследованиям занимал видный специалист по радиосвязи и полупроводникам – профессор Дэниел Нобл, которого переманил из университета Галвин. Под руководством Нобла в 1940 году и было создано первое переносное радио с двусторонней связью – типа телефонной. Рация Handie-talkie весила всего два килограмма и работала на расстоянии 15 миль. Впоследствии на смену ей пришла еще более миниатюрная и вдвое более сильная Walkie-talkie (правда, работать на предельном охвате в 30 миль она могла лишь при выпущенной 3,5-метровой антенне).

Этот прибор разошелся по армиям и полициям всего мира, получив международное признание. Всего за годы войны Galvin Manufacturing выпустила более 130 тысяч армейских раций.

К концу войны Пол Галвин еще раз продемонстрировал свою отменную деловую интуицию. Поняв, что золотые денечки военных заказов скоро закончатся, он распорядился, чтобы тысячи пылившихся на складах комплектов автомобильного радио были переделаны в домашние радиоприемники и пущены в продажу. Таким образом компания пережила неизбежный послевоенный штиль, сохранив главное – свою налаженную дистрибьюторскую сеть.

Галвинизация бизнеса

Как руководитель, Пол Галвин был неоценим для компании и невыносим для сотрудников – по крайней мере для тех из них, кто рассчитывал на спокойное место работы. Покой им только снился – взрывной и бескомпромиссный стиль руководства Галвина позже назвали «ежедневным судилищем» над персоналом.

Хотя уничижительное судилище в данном случае вряд ли уместно – суд Галвин вершил скорый, но правый. Ежедневные разборки с подчиненными руководитель устраивал не ради

демонстрации власти и не для мелочных придирок, а с целью ежедневного повышения качества работы. Галвин считал, что эффективность компании прямо пропорциональна эффективности каждого ее сотрудника, и не отступал от своих жестких требований все время, пока руководил компанией.

Любопытно, что даже при таких драконовских методах руководства большинство сотрудников Galvin Manufacturing застревали в компании на десятилетия. И даже в периоды экономического застоя или кризиса, когда американский бизнес погружался в кратковременную кому, компания Галвина одна из немногих демонстрировала поразительную живучесть.

После окончания Второй мировой войны ее руководитель снова занялся поиском новых продуктов и направлений деятельности, которые обеспечили бы компании, в 1947 году сменившей название на Motorola, лидирующее положение на десятилетия вперед.

Еще во время войны Галвин обратил пристальное внимание на бурно прогрессировавшее телевидение и в очередной раз предложил своим сотрудникам догнать и перегнать тогдашнего лидера – компанию Radio Corporation of America (RCA). В 1940-е годы она выпускала телевизоры с диагональю экрана 10 дюймов по цене $300, а Галвин решил сэкономить на размерах экрана (всего 7 дюймов), значительно выиграв в габаритах, весе и, разумеется, в цене. В 1947 году модель Golden Vision VT71 была готова к запуску в серийное производство. На робкое возражение отдела продаж, что заявленная розничная цена $179,95 принесет одни убытки, глава корпорации ответил в свойственном ему духе: «Я назвал эту цену, и она таковой и останется, а вам придется с ней работать».

И снова не ошибся. Доступная цена сразу же сделала телевизоры Motorola массовым товаром – за первый год было продано 100 тысяч штук, и вскоре компания за счет телевизоров компенсировала неизбежный послевоенный спад производства, вызванный исчезновением военных госзаказов. К 1950 году Motorola более чем вдвое увеличила объемы продаж «золотого» 1944-го ($177 млн).

В том же году 55-летний глава фирмы начал готовить себе в преемники единственного сына, а спустя шесть лет пост президента Motorola занял 32-летний Роберт Уильям Галвин. Его назначение совпало с выпуском очередного прорывного продукта – первого в мире пейджера. Он стал незаменимым средством коммуникации среди самых широких слоев населения – от медперсонала и менеджеров до священников и даже… девочек по вызову.

В 1959 году Пол Галвин умер, оставив сыну процветавшую компанию с годовым оборотом в $225 млн.

Что может быть лучше хорошего бизнеса?

До начала 1970-х Motorola выпускала телевизоры на американский рынок, успешно конкурируя с японскими «агрессорами». А затем, неожиданно для многих, свернула столь доходное производство! Это, на первый взгляд, нелогичное решение – зачем же отказываться от хорошего бизнеса? – тем не менее, вошло во многие учебники по менеджменту как решение тонко просчитанное и беспроигрышное.

Хорошее меняют только на лучшее, именно это проделала Motorola, открывшая для себя более перспективные направления – производство микропроцессоров, кабельных модемов, спутниковых систем связи и непосредственно связанных с ними мобильных телефонов. В результате радикальной смены приоритетов обороты компании за следующее десятилетие выросли более чем вчетверо – с $796 млн в 1970 году до $3,1 млрд в 1980-м.

Правда, на обоих главных рынках – интегральных схем и мобильной связи – «пионерке» Motorola в итоге пришлось потесниться, пропустив вперед конкурентов – соответственно, Intel и Nokia.

Рынок микрочипов Motorola открыла в 1952 году, создав первый серийный полупроводниковый транзистор. Спустя короткое время компания, следуя генеральной линии отца-основателя, стала выпускать первые автомобильные транзисторные приемники. Они приобрели такую популярность, что Motorola пришлось открывать производственные мощности по всему миру. В 1974 году специалисты компании создали первую микросхему – 8-битный процессор, положивший начало обширному семейству микрочипов, без которых не смогли бы работать персоналки Apple – компьютеры Macintosh. Однако безальтернативная ставка на «яблочников» со временем оказала Motorola дурную службу – Apple

Виржиния и Пол Галвины

Роберт Уильям Галвин

в итоге проиграла битву титанов компании IBM с ее фирменным «Intel Inside».

Это поражение, однако, не охладило пыла Motorola в других секторах рынка бытовой электроники. Один из топ-менеджеров компании объективно оценил сложившуюся на рынке ситуацию афористическим вердиктом: «Если бы мы в свое время додумались до слогана «Motorola Inside», то он сегодня мозолил бы глаза не реже, чем всем известный слоган Intel».

Справедливость этого утверждения на рынке мобильных телефонов трудно оспорить. Его поистине неограниченные перспективы Motorola снова осознала раньше других. Потратив в 1970-х годах на соответствующие исследования более $100 млн, компания, благодаря своей оперативности, десятилетием позже вышла в единоличные лидеры нового направления. Первый серийный аппарат сотовой связи – килограммовый Motorola DynaTAC, созданный командой исследователей под руководством Мартина Купера в 1973 году, поступил в продажу ровно через десять лет. И еще год спустя в мире действовало около сотни сотовых сетей, собранных на оборудовании Motorola.

Лишь полтора десятилетия назад компания Галвина уступила лидерство финской Nokia. Во многом это произошло из-за стратегического просчета руководства Motorola.

Еще в конце 1980-х оно пришло к выводу (как показало время, глубоко ошибочному) об ограниченности рынка сотовой связи и решило инвестировать $5 млрд в амбициозный проект Iridium, казавшийся наиболее перспективным. Проект предполагал охват сотовой связью всего земного шара с помощью 66 космических спутников, но в результате «космическими» оказались цены этой услуги. Компании Iridium LLC, разработавшей проект, удалось набрать лишь 50 тысяч абонентов, готовых платить за сотовую связь такие деньги, в то время как для достижения уровня рентабельности требовалось вдесятеро больше участников. Со временем операторы сотовой связи нашли куда более дешевый и эффективный способ предоставлять ее в любой точке земного шара – всем известный сегодня роуминг.

В итоге на проекте Iridium был поставлен крест. И если бы не вмешательство Пентагона, имевшего свои виды на космическую связь, все выведенные к тому времени на орбиту спутники ждала участь затопленной космической станции «Мир».

Так Motorola в канун нового столетия и тысячелетия потеряла значительные деньги и, что хуже – инициативу на столь динамично развивающемся рынке. Хотя и стабильное второе место на нем – само по себе достижение, о котором другие игроки могут только мечтать.

Mobile in mobilis

Если кто забыл – таков был девиз капитана жюльверновского «Наутилуса». «Подвижное в подвижном» – сегодня эта фраза на латыни вполне могла бы стать корпоративным слоганом Motorola. Потому что компания, три четверти века назад созданная Полом Галвином, в последнее время опять взялась за старое – снова пытается объединить двух китов современного рынка: автомобиль и мобильную связь.

На повестке дня – сетевой автомобиль, превращающий его из средства передвижения в мобильный центр коммуникаций. И это уже не фантастика – Motorola заключила контракт на соответствующие совместные разработки с компанией DaimlerChrysler, ведутся переговоры с Renault и другими лидерами мирового автопрома.

Кроме того, исследовательский центр Motorola занят более масштабным (и оттого еще более фантастическим) проектом – созданием «интеллектуальной» среды обитания. Разумная экология по замыслу специалистов Motorola должна будет состоять из четырех основных узлов – человека, его дома, рабочего места и автомобиля, – между которыми постоянно осуществляется связь и взаимодействие.

Руководство компании оптимистично смотрит на перспективу реализации очередного грандиозного проекта. Ведь по объему инвестиций в новые разработки Motorola не уступает крупнейшим технологическим компаниям мира, а число ее партнеров среди университетов и исследовательских центров давно перевалило за сотню.

И вообще компания со штатом в 88 тысяч человек, занимавшая в 2007 году 205-е место среди 500 крупнейших компаний мира (по версии журнала Fortune) с капитализацией $41,3 млрд, за три четверти века уже привыкла к статусу идущей впереди планеты всей. Как завещал великий Галвин. EOF

Walkie-talkie

Первый сотовый телефон DynaTAC

Мобильный телефон Motorola

Переполнение буфера в AOL Instant Messenger

Программа: AOL Instant Messenger (AIM) версии до 6.8.7.7.
Опасность: Средняя.
Наличие эксплоита: Нет.
Описание: 1. Уязвимость существует из-за знаковой ошибки в реализации библиотеки SIP-протокола sipXtapi.dll при обработке пакетов от RTCP-отправителя. Удаленный пользователь может вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе. Для успешной эксплуатации уязвимости пользователь должен принять Video Messaging сессию.

2. Уязвимость существует из-за ошибки в библиотеке sipXtapi.dll при обработке RTP-данных. Удаленный пользователь может с помощью специально сформированного Extension Length RTP-заголовка вызвать переполнение динамической памяти и выполнить произвольный код на целевой системе. Для успешной эксплуатации уязвимости требуется наличие активной голосовой сессии пользователя.
URL производителя: www.aim.com.
Решение: Установите последнюю версию 6.8.7.7 с сайта производителя.

Переполнение буфера в продуктах IBM Informix

Программа: IBM Informix Client Software Development Kit (CSDK) 3.50, возможно, другие версии; IBM Informix Connect 3.50, возможно, другие версии.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки проверки границ данных при обработке .nfx-файлов, содержащих слишком длинную строку HostList в файле setnet32.exe (версия 3.50.0.13752). Удаленный пользователь может с помощью специально сформированного .nfx-файла вызвать переполнение стека и выполнить произвольный код на целевой системе.
URL производителя: www.ibm.com.
Решение: В настоящее время способов устранения уязвимости не существует.

Отказ в обслуживании в HTML-Parser

Программа: HTML-Parser версии до 3.63.
Опасность: Средняя.
Наличие эксплоита: Нет.
Описание: Уязвимость существует из-за ошибки в функции decode_entities() в файле utils.c. Удаленный пользователь может с помощью специально сформированной строки вызвать зацикливание приложения, использующего уязвимую библиотеку.
URL производителя: search.cpan.org/dist/HTML-Parser.
Решение: Установите последнюю версию 3.63 с сайта производителя.

Переполнение буфера в Haihaisoft Universal Player ActiveX-компоненте

Программа: Haihaisoft Universal Player ActiveX Control 1.4.8.0, возможно, другие версии; Haihaisoft Universal Player 1.4.8.0, возможно, другие версии.
Опасность: Высокая.
Наличие эксплоита: Да.
Описание: Уязвимость существует из-за ошибки проверки границ данных при обработке ActiveX-компонента (MyActiveX.ocx). Удаленный пользователь может с помощью специально сформированного веб-сайта передать слишком длинную строку свойству «URL», вызвать переполнение стека и выполнить произвольный код на целевой системе.
URL производителя: www.haihaisoft.com.
Решение: В настоящее время способов устранения уязвимости не существует.

Множественные уязвимости в Achievo

Программа: Achievo 1.4.2, возможно, более ранние версии .
Опасность: Средняя.
Наличие эксплоита: Да.
Описание: 1. Уязвимость существует из-за недостаточной обработки входных данных в параметре description в сценарии dispatch.php, когда параметр atknodetype установлен в значение scheduler.scheduler_category. Удаленный пользователь может с помощью специально сформированного запроса выполнить произвольный код сценария в браузере жертвы в контексте безопасности уязвимого сайта. Для успешной эксплуатации уязвимости злоумышленнику требуются привилегии на создание групп расписаний.

2. Уязвимость существует из-за ошибки в сценарии modules/docmanager/attributes/class.documentfileattribute.inc, которая позволяет удаленному пользователю загрузить файлы с произвольным расширением на систему посредством сценария dispatch.php, когда параметр atknodetype установлен в значение docmanager.documenttype. Удаленный пользователь может загрузить и выполнить произвольный PHP-сценарий на целевой системе с привилегиями веб-сервера. Для успешной эксплуатации уязвимости злоумышленнику требуются привилегии на создание шаблонов документов.
URL производителя: www.achievo.org/product.
Решение: Установите последнюю версию 1.4.3 с сайта производителя.

Составил Александр Антипов

Визитка

СТАНИСЛАВ ШПАК, более 5 лет занимается сопровождением Active Directory и Windows-серверов. Имеет сертификаты MCSE по Windows Server 2000/2003

POWER OFF

Я не помню, когда это началось. Может быть, три дня назад, а может быть, и неделю. В тот день я пришел домой с работы и, хотя всего час назад оторвался от компьютера на работе, тут же уселся за домашний. Как всегда.

Стандартные «пять минут» на проверку почты, сообщений «ВКонтакте», просмотр ЖЖ и все такое, растягивающееся обычно на часок-другой. Но в тот день я вдруг обнаружил, что открывающиеся передо мной интернет-страницы я воспринимаю не так, как обычно. Даже не воспринимаю, а ощущаю. Поверьте мне на слово, блогосфера – самая насыщенная чувствами часть Интернета. Все пестрое разнообразие эмоций, размазанное тонким слоем по новостным и развлекательным сайтам, полыхнуло на меня из ЖЖ. Я с удивлением обнаружил, что не просто чувствую эмоции писавших там людей – я их как будто переживаю сам.

Страницу, на которую я наткнулся, вела девочка лет 14-15. Обычные посты про прогулки с друзьями, про школу, собственные стихи, первые размышления о любви в закрытых записях, которые я без труда видел, а потом… Потом была последняя запись – предсмертная записка, оставленная ею в виртуале. И комменты – сначала шутливые или недоуменные, а потом печальные: «любим, скорбим», «прощай», «зачем?». И ощущение чьего-то присутствия между строк.

И тогда я нашел эту девочку – это было нетрудно, я просто откуда-то знал, где ее искать. Вышел из дома, прыгнул в маршрутку, спрятав уши в плеер, а голову – в мысли о том, почему я еду к ней. Без труда разыскал ее дом, поднялся по не очень ухоженной лестнице и вошел в квартиру. Пока я поднимался, звуки и шорохи жилого дома незаметно смолкли, а краски окружающего мира поблекли, превратившись в градации серого. Мрачное запустение, толстый слой пыли на мебели, приунывшие плюшевые мишки в углу и компьютер, перед которым сидела маленькая, почти детская фигурка.

– Почему они это пишут? – спросила она тихо, похоже, ничуть не удивившись моему визиту. – Я же это написала в шутку. Да, я тогда действительно хотела это сделать, но я же жива!

Заглянув в монитор, я увидел уже знакомую страницу ЖЖ. «Любим, скорбим»… Тем временем девочка продолжила:

– Я пишу им ответы, но они не появляются на странице. Я хочу создать новую запись, но не могу. Я вижу, как мои друзья приходят и уходят в аське, но у меня не получается написать им, – она говорила спокойным бесцветным голосом, будто повторяла эти слова уже тысячу раз. Вот только когда она повернула ко мне свое личико, я увидел, что следы слез разделяют его на три равные части. – И почему не приходит домой с работы мама? Почему?

Сначала я не нашелся, что ответить. А потом начал придумывать.

– Мама, наверное, просто задержалась по делам. А я пришел, чтобы починить тебе Интернет – у тебя немного глючит компьютер, ему надо дать отдохнуть. Выключи его на пару минут, а когда потом включишь, все снова будет работать.

– Он перегрелся?

– Скорее всего.

Это была ложь, которая далась мне с трудом. Будучи сисадмином, я понимал, что перегрев компьютера, если он и был, вряд ли стал бы причиной так странно неработающего Интернета. Но я чувствовал, что компьютер надо выключить. Это было ничем не подкрепленное ощущение, сродни тому, которое и привело меня в квартиру незнакомой девочки в районе, где я ни разу не был.

– И это поможет, правда? – с надеждой спросила девочка.

– Правда, – бессовестно соврал я, – поверь мне, я же специалист.

Девочка повернувшись ко мне, спросила:

– Выключать?

У нее было миленькое личико, наверное, лет через пять она стала бы красавицей и все бы в ее жизни могло быть хорошо, только… Только вот не было у нее уже жизни.

– Выключай, – сказал я.

Она нажала OK, и началось стандартное завершение работы Windows. И по мере того, как компьютер останавливал запущенные процессы, закрывал сетевые подключения, сохранял параметры, вокруг начали проявляться краски, исчезала пыль и разбросанные вещи, а девчушка становилась все прозрачнее.

– Ой, – сказала она, – щекотно!

И засмеялась. А потом растаяла. Я стоял возле выключенного компьютера в чужой комнате чужой квартиры и смотрел на фотографию девочки, которой уже не суждено было вырасти. Хорошая, веселая фотография, улыбающееся лицо, озорные глаза. Только все портила черная лента, идущая через угол снимка. В углу комнаты кто-то за-

шевелился и всхлипнул. Женщина, наверное, мама девочки, обнимала подушку и плакала, отвернувшись лицом к стене. Она не заметила меня, и я тихонько вышел из комнаты, а затем и из квартиры. Мне не хотелось возвращаться домой к компьютеру, но больше идти мне было некуда.

Вот так это было в первый раз. Потом я начал специально искать в Сети «зависшие», как я их называю, души. В основном это, конечно, блоггеры и люди, не мыслящие свою жизнь без онлайн-общения. Но встречались и более редкие экземпляры – например, брокер, который не оторвался от экрана с замершими котировками, даже когда я начал с ним разговаривать. С большинством срабатывала ставшая уже стандартной байка про специалиста техподдержки, который пришел починить Интернет. Достаточно было заставить выключить компьютер, как душа растворялась, а я вываливался в обычный мир красок, запахов и звуков. Не скажу, что я успел таким образом спасти много душ, но как-то и не задавался целью их считать. Десяток? Два? Да какая разница, меня полностью увлек этот процесс. Но однажды я наткнулся на геймера.

Геймер был классический – из тех, для кого это состояние души, а не возраста, и кого насильно можно оторвать от компа только с истерикой и воплями. Вот такой мне и попался. Я стоял и смотрел на душу парнишки, увлеченно командующего нарисованными войсками. Как ни странно, его не смущало то, что в игре уже ничего не происходило, кроме перемещения его войск. Пришлось мне самому заявлять о себе:

– Эй, – сказал я, – я пришел тебе помочь. У тебя что-то с компьютером.

– Все нормально у меня с компьютером, не мешай, – не очень-то любезно буркнул парень в ответ, лишь вскользь удостоив меня взглядом.

– Но ты же играешь сам с собой, ты что, не видишь?!

– Отвали, сейчас просто мой ход, не знаешь игру, так не лезь с советами!

Похоже, уговаривать его выключить компьютер, было бесполезно, поэтому я пожал плечами и тут же совершил ошибку – потянулся к кнопке выключения системного блока. Вот тут парнишка мигом оторвался от экрана! Толкнув меня так, что я упал, он заорал:

– Не смей лапать компьютер! Даже родители не имеют права трогать мой комп! Уйди! – с этими словами он протянул руку в монитор и, даже не глядя, вытащил здоровенный тесак размером как минимум с половину собственного роста!

Зрелище достаточно хлипкого парня со здоровенным мечом в руке оказалось настолько неестественным, что я чуть было не поплатился за это, когда он без всяких усилий размахнулся и направил удар лезвия в меня. Я вскочил на ноги, но и противник оказался шустрым – не тратя времени на повторный замах, он с какой-то непостижимой грацией фехтовальщика нанес колющий удар в мою сторону. Лезвие со скрежетом наполовину вошло в бетонную стену и застряло. А я прыгнул к компьютеру, краем глаза заметив, как парнишка, оставив меч в стене, впопыхах слепил фаейрбол и швырнул в меня. «Заряд» попал мне прямиком в левое плечо и окатил волной жара. Так вот ты какой, магический огненный шарик! Похоже, навыки мага у моего противника были развиты меньше, чем навыки воина, а то меня испепелило бы на месте! Приземлившись на ламинированный пол, я проехал на животе остатки расстояния до системного блока, нажал и стал удерживать кнопку выключения. Парнишка завизжал и, с силой потянув меч, вырвал его из стены. Прямо король Артур, ей богу! А я держал кнопку и считал. Один, два, три… Конечно, можно было подождать, пока компьютер штатно завершит работу. Только вот геймер с мечом наперевес искромсает меня в салат своим гипертрофированным ножиком, а потом, мирно улыбаясь, растает в воздухе. Не-е-е-т, мне такая перспектива не нравилась. Четыре, пять! Он почти успел. Удар меча с разворота должен был бы разделить мое тело на две половинки, только «почти» – не считается.

…девчушка становилась все прозрачнее.
— Ой, — сказала она, — щекотно! А потом растаяла…

Я ожидал того, что он растает, как и все прошлые души. Но, похоже, он действительно был не готов расстаться со своей игрушкой, потому что, лишившись объекта приложения – компьютера, – душа не рассеялась мирно вокруг, а ударила упругой взрывной волной. Меня смело как пушинку, каким-то образом протащило сквозь стены соседней квартиры и вышвырнуло на улицу. Падение со второго этажа спиной на газон было не очень приятным, но не смертельным. А вот сфера черного света, как будто дышащая и светящая острыми световыми иглами, мне очень не нравилась. Она еще росла, выпирала из окон второго этажа, ее край уже вышел за фасад здания, но это было недолго –

как бы вздохнув, с легким свистом она втянулась куда-то внутрь квартиры и исчезла.

Не поленившись, я решил снова подняться и посмотреть, что происходит в квартире. Однако на втором этаже наткнулся на запертую дверь. Это меня озадачило – интересно, а как я до этого проникал в жилища? Все предыдущие разы я беспрепятственно заходил к людям, даже не задумываясь о замках. Хотя стоп, поправка – не к людям, а к душам. Значит, если сейчас передо мной дверь, то за ней уже нет зависшей души. Ну будем надеяться, что так. Плечо нестерпимо жгло, и мне ничего не оставалось делать, как отправиться домой.

Все души, которые мне попадались, были привязаны к компьютеру. И я хитростью или уговорами просил их его выключить

А дома меня ждали. Человек просто сидел на диване и рассматривал фотографии на стене. Компьютер был выключен, хотя я был уверен, что сам я его не выключал. Ну да мало ли – может быть, опять энергосеть в доме барахлила. За последнее время я уже привык ничему не удивляться, поэтому вполне буднично спросил:

– Простите, а вы кто?

– Ну, говоря в понятных тебе терминах, я – системный администратор, обнаруживающий разделенные энергетические сгустки, бывшие частью одного ментального целого. Ты можешь звать меня просто – Мастер.

Я наклонился и включил системный блок. Комп радостно пискнул и начал загружаться. Почему-то мне стало весело. Тоже мне, коллега!

– Системный администратор? И какую систему администрируете? Энергетические сгустки – это души? – с некоторой иронией спросил я.

– Да можно сказать и души. А остальное – сложный вопрос. Прежде чем я объясню тебе аналогию, позволь поинтересоваться: как ты думаешь, что с тобой происходит?

– Хм, я много об этом думал. Я чувствую души умерших людей, застрявшие за компьютером, нахожу их и стараюсь освободить, – это прозвучало так просто и связно, что даже сам немного удивился.

– А ты не думал, почему так происходит? Ведь не все же люди, умирая, прилипают к компьютеру.

– Э-э-э-э, вот чего не знаю, того не знаю.

– Тогда позволь мне немного пояснить. «Разделение» души известно еще с древних времен и появилось примерно одновременно с зарождением искусства. Наверное, ты слышал выражение – «он отдал этому занятию всю душу». Так вот это, конечно, перебор. Всю душу никто не может отдать, чем бы он ни занимался. Но часть души – возможно. А если человека занятие увлекло, то и большую часть. В прошлом это происходило нечасто и встречалось в основном у творческих людей – художников, писателей, музыкантов. Если ты отдаешь большую часть души своему делу, это неплохо. Вот только возникает небольшая проблемка, когда приходит время расставаться с жизнью…

– Тогда и происходит «разделение» души?

– Да. Часть души, как и положено, пытается покинуть этот мир, но другая часть даже не замечает, что уже лишилась физической оболочки. Однако без тела душа не может себя проявить в реальном мире. В результате обе половинки страдают, только вторая этого не замечает.

– И для этого нужны мы, те, кто будет освобождать «застрявшие» части?

– Ну не совсем, – Мастер поморщился, – вообще-то на это очень долго не обращали внимания. Подумаешь, разделенная душа. Происходило это редко, некоторые превращались в привидения, некоторые рассеивались через какое-то время сами собой, некоторых выталкивало эмоциональным фоном оставшихся в живых, ну обычное дело. И все бы хорошо, пока не наступили новые времена с научно-техническим прогрессом. Таких душ стало больше, и пришлось кому-то помогать им воссоединяться.

– И появились администраторы душ? – я никак не мог начать воспринимать разговор серьезно. Однако Мастер никак не отреагировал на мой сарказм.

– Да, примерно так. Мир был создан с набором правил и законов. Например, смена дня и ночи или закон всемирного тяготения. Представь, что ты администрируешь локальную сеть. Ты поставил рабочие станции, установил на них операционную систему и нужные программы, связал их в сеть, настроил серверы, коммутационное оборудование, проверил – все работает. А потом запустил в систему пользователей. И сначала все были смирные и работали по твоим заветам. Но потом кто-то переконфигурировал операционку, кто-то установил собственные программы, другой, наоборот, удалил те, которые ставил ты. И вроде бы система в целом работает, но нет-нет, но где-то начинают появляться сбои. Так и в мире – вся деятельность человечества направлена на то, чтобы обходить законы и раздвигать рамки. Конечно же, это не проходит бесследно. На большинство последствий можно не обращать внимания – они касаются только самого человечества и являются расплатой за его же необдуманные действия. К тому же система подвержена саморегуляции, но бывают и исключения, вот как с зависающими душами. Они стали угрожать стабильности всей системы. И тогда появились мы, системные администраторы. Некоторые из нас обучены находить и освобождать души, некоторые, такие как я, занимаются исследованиями этого процесса.

– А кто тогда я? Меня же никто ничему не учил!

– Да мы как раз подходим к самому главному. Дело в том, что такие сисадмины сами являются душами умерших когда-то людей. Мы не помним, кого именно, да это и не важно – главное, что у нас нет телесной оболочки. И те души, которые мы освобождаем, – души уже умерших людей. Да, ты похож на нас. Вот только… ты еще не умер.

Такого оборота я не ожидал и даже не нашелся, что сказать. А Мастер продолжил:

– Поэтому к тебе и пришел я. Нашли мы тебя благодаря последней освобожденной тобой душе. Не знаю, что у тебя пошло не так, но произошел выброс энергии, и мы тебя засекли.

– Я сам выключил компьютер, – пробормотал я.

– Что? – Мастер насторожился.

– Все души, которые мне попадались, были привязаны к компьютеру. И я хитростью или уговорами просил их его выключить. Когда они это делали, то тихо таяли в воздухе. В последнем случае у меня не получилось – и мне пришлось выключать компьютер самому.

Тут я покосился на уже готовый к работе компьютер, и меня осенило!

– Это вы выключили мой компьютер, да? Думали, что я – одна из зависших душ, и решили меня освободить?

– Ну для обычной души ты слишком легко можешь отходить от компьютера, – похоже, Мастер немного смутился, – но на всякий случай я должен был убедиться.

– Так что со мной? Я не умер, но я веду себя как зависшая душа, да?!

– Да. Похоже, ты – зомби.

Вот это были новости так новости! Я опешил и смог выдавить из себя только:

– Я – кто???

– Зомби. Когда число разделенных душ стало увеличиваться, мы провели экстраполяцию процесса. Люди все больше и больше уходят в виртуальный мир, предоставленный им компьютером. Как следствие, в скором времени мы можем ожидать появления разделения души не после, а до смерти физической оболочки. И ты – первый. Ты работал компьютерщиком, все свободное время проводил за компьютером, даже иногда спал перед ним: ты не мыслил своего настоящего и своего будущего без компьютера, ты совершил первый прорыв – твоя душа и тело разделились.

– Ух ты, как здорово! Я теперь могу жить здесь, помогать душам, быть где угодно, да?

Мастер покачал головой:

– Это не здорово. Ты можешь. Твоя душа большей частью здесь. А то, что осталось, – почти бездушный зомби, гнетущее зрелище. Дай руку – я покажу тебе оставшегося тебя.

Завороженный, я протянул руку Мастеру. И как только он меня коснулся, весь окружающий мир рухнул в туман, который тут же рассеялся, и я оказался в собственной квартире и смотрел на другого себя. Он, то есть другой я, сидел со стеклянными глазами и пялился в телевизор. И ел бутерброд, с него капал кетчуп на мой любимый пушистый белый ковер, который я сам когда-то и выбирал! Вошла мама.

– Мать приехала спустя неделю после того, как ты перестал отвечать на телефонные звонки. И она уже близка к тому, чтобы обратиться за медицинской помощью, но боится, что тебя упекут в психушку и будут накачивать лекарствами, – раздался откуда-то голос Мастера.

«Сынок, я приготовила твой любимый борщ, пойдем на кухню, поедим», – обратилась к моему телу мать. Я не среагировал. Она подошла и погладила меня по голове. Потом заметила на ковре капли кетчупа, вздохнула и вышла из комнаты. Спустя минуту вернулась с мокрой тряпкой и попыталась убрать мое свинство, ненароком заслонив собой экран телевизора.

«Уйди, ты мешаешь», – безразлично сказал я. Мать поняла, что нашлось что-то, на что я среагировал и, встав передо мной, с вызовом и болью сказала:

«Не уйду, пока ты не объяснишь, что с тобой происходит!»

А я... Я приподнялся в кресле и оттолкнул ее в сторону. Она зацепилась ногой за край ковра и упала, а я тем временем, ничуть не обратив на это внимание, уселся в кресло и снова уткнулся в телевизор. Мать не стала вставать. Она лежала на полу и плакала, не понимая, что произошло с ее сыном. Ведь я всегда любил свою мать.

– Я не хочу на это смотреть! – сказал я и тут же оказался в тумане, а затем снова в моей комнате, но уже без мамы и своего тела, зато с Мастером.

– Мастер, чего ты от меня хочешь?

– Ты все видел сам. Я хочу, чтобы ты вернулся. Вернулся и запомнил все, что с тобой было. Ты должен рассказать это другим, нужно, чтобы люди не превращались в зомби. В существа, которым все равно. В уродцев, которые не имеют чувств. В тупые биологические организмы.

– Мастер, но даже если я не сойду с ума, оттого что буду помнить все, что было, мне никто не поверит!

– Если ты будешь рассказывать это как правду, то да. Но, – Мастер улыбнулся, – люди любят небылицы. Напиши об этом книгу. Ну или хотя бы рассказ в компьютерный журнал. Закинь в популярные страницы в ЖЖ. И в любом случае – спасибо. Мы теперь знаем, что такие как ты, должны сами выключать свой компьютер, этого нельзя сделать за вас. Вы еще не умерли, а пока вы живете или хотя бы просто существуете – у вас остается свобода выбора.

– Знаешь, Мастер, мне иногда кажется, что мы свою свободу выбора зачастую используем для того, чтобы принять решение отказаться от этой свободы.

Он улыбнулся. Как-то грустно.

– Может быть. А теперь ты знаешь, что нужно сделать.

Я кивнул и щелкнул «завершение работы».

Мастер стал таять в воздухе, потом я понял, что это таю я, а может быть, тает сам воздух, а потом снова все заволокло серым туманом.

Проснулся я за компьютером. Подумал, что мне все приснилось – спать в обнимку с клавиатурой не очень удобно, поэтому сны всегда снятся странные. Потом сообразил, что слишком как-то все было реально, и пошел искать маму. Мы давно не живем вместе, но если она действительно приехала, как сказал Мастер, то это был не сон. Однако не только мамы, но и следов ее пребывания в квартире не было. Что ж, сон был очень натуральным, я его немного посмаковал в уме, пока варил пельмени. Спать мне уже не хотелось, после еды я снова сел за комп и проторчал за ним до рассвета.

Летом красивые рассветы, а балкон моей квартиры выходит на восток – поэтому, собираясь ложиться спать, я решил немного полюбоваться восходящим солнцем. И все бы ничего, только возвращаясь в комнату с балкона, что-то темное на полу привлекло мое внимание. Рассветного солнца было недостаточно, и я включил свет – на белом ковре у кресла перед телевизором бурым цветом темнело пятно от кетчупа…

Я медленно опустился в кресло. Значит, не такой уж и сон, да? Значит, это все было? Спать уже не хотелось. Я снова вернулся за компьютер. Мне надо написать рассказ. Пока воспоминания яркие. Пока сам верю в то, что это было. Пока не отмыл кетчуп с ковра. Пусть я всего лишь пинг в какой-то системе, но в отличие от ICMP-пакета я могу не только тупо нестись по маршруту, но и что-то сделать. Чистый виртуальный лист уже белел на экране монитора. Выключив аську и телефон, я напечатал первую строчку:

«Я не помню, когда это началось. Может быть, три дня назад, а может быть, и неделю…» EOF

Содержание журнала за 2009 год

Администрирование

«10-Страйк» Дмитрий Степанов №4
«Облачные» перспективы защиты корпоративных endpoint-компьютеров Алексей Лесных №5
Active Directory. Проводим профилактическое обслуживание Иван Коробко №7
CrossBow – сетевые технологии OpenSolaris Сергей Яремчук №8
FastReport Server 2.2. Построй свой собственный SaaS Михаил Филиппенко №11
Fedora8 –> Fedora 10: не пора ли обновиться? Павел Закляков, Глеб Рыжаков №3
FreeBSD tips. Periodic на службе сисадмина Сергей Супрунов №7
Hardware Inspector – ваш помощник Сергей Унагаев №3
Hyperic HQ – система мониторинга корпоративного уровня Дмитрий Петухов №1
Linux 2.6.30: выглядит неплохо! Игорь Штомпель №8
Linux Live CD. Создаем загружаемый диск Андрей Бирюков №10
Live Migration. Что нужно для ее использования? Александр Косивченко №10
Lustre FS. Настраиваем и используем кластерную систему в промышленных масштабах. Часть II Виталий Банковский №1
MyZCI поможет. Моя система автоматической инвентаризации Юрий Винник №11
NTI Shadow for ReadyNAS. Проводим резервное копирование данных Алексей Бережной №8
nUbuntu – дистрибутив для тестирования защищенности Сергей Яремчук №1
PowerShell. Определяем имя текущего домена Иван Коробко №4
PowerShell. Поиск объектов в каталоге Active Directory Иван Коробко №4
PowerShell. Поиск объектов в каталоге Active Directory Иван Коробко №5
QAD-командлеты. Простые команды вместо сложных конструкций Сергей Яремчук №8
SCCM 2007 R2 – резервное копирование и восстановление Алексей Тараненко №7
Slack – автоматизируем настройку сервисов Алексей Коробкин №2
Sun OpenBoot Prom – между железом и софтом Владимир Василькин №8
Sun Secure Global Desktop. Все ваши приложения в окне браузера Антон Борисов №9
Sun VirtualBox как персональная система виртуализации Алексей Бережной №12
VMware Player 2.5 Иван Коробко №3
VMware Workstation 6.5. Способы установки Иван Коробко №2
WDS поможет. Установка операционных систем. Часть 1 Андрей Бирюков №12
Windows 7: продолжаем знакомство Илья Рудь №6
Windows 7: что новенького? Андрей Бирюков №4
Windows Server 2008 R2. Новые возможности служб AD DS Александр Емельянов №10
А нужен ли банкам Linux? Как правильно запустить банк-клиент iBank 2 Владимир Закляков №11
Автоматическая установка Adobe Creative Suite 3 Иван Коробко №4
Автоматическая установка драйверов Иван Коробко №1
Альтернатива файловому серверу – дисковое хранилище NETGEAR ReadyNAS Алексей Бережной №7
Альтернативы MS Project. Пробуем популярные Open Source-решения Сергей Яремчук №9
Анализируем трафик с Nulog2 Сергей Яремчук №1
Безболезненная замена устаревшего или отказавшего контроллера домена на базе Windows Server Кирилл Семаев №3
Безопасность бизнеса. Защищаем информационные активы Артем Хазов №7
Больше, чем почта. Zimbra Collaboration Suite 6.0 Максим Бочкин №11
Веб-топ-решение на основе Ulteo Virtual Desktop Антон Борисов №10
Виртуализируем предприятие. Чтобы заработала бухгалтерия Сергей Крутских №11
Возможности VMBitrix. Разгадка виртуальной машины Игорь Антонов №11
Вы еще не используете Windows Vista? Андрей Бирюков №1
Графические инструменты для Server Core Сергей Яремчук №2
Делегируем права на перемещение учетных записей пользователей в Active Directory. Часть 1 Вадим Андросов №3
Делегируем права на перемещение учетных записей пользователей в Active Directory. Часть 2 Вадим Андросов №4
Делегируем права на перемещение учетных записей пользователей в Active Directory. Часть 3 Вадим Андросов №5
Делегируем права на перемещение учетных записей пользователей в Active Directory. Часть 4 Вадим Андросов №6
Домашний хостинг. Используем сервисы динамического DNS Иван Коробко №9
Доступная виртуализация: Citrix XenServer 5.0 Андрей Панченко №6
Доступный WiMAX Павел Закляков, Георгий Пахомов №5
Заменяем сервер MS Exchange. Установка Horde Groupware Рашид Ачилов №9
Запускаем сценарий загрузки от имени администратора Иван Коробко №3
Квартет: «CAMO», «1C», wine и Etersoft Сергей Барановский №1
Комплексное решение: виртуализация + отказоустойчивый кластер Александр Косивченко №9
Контролируем изменения в конфигурационных файлах Владимир Легеза №1
Контроль трафика. Прослушиваем и просматриваем вместе с BWMeter Андрей Понарев №10
Конфигурируем DHCP-серверы и настраиваем динамические обновления DNS Сергей Супрунов №9
Мониторинг Cisco IDS/IPS на примере модуля IDSM2 с помощью MRTG Андрей Дугин №5
Мониторинг Cisco IDS/IPS на примере модуля IDSM2 с помощью MRTG. Часть 2 Андрей Дугин №6

Содержание журнала за 2009 год

Мониторинг состояния температурных датчиков с помощью протокола SNMP Вадим Шпурик №10
Настраиваем Exchange 2007 для отправки электронной почты на внешние адреса Сергей Крутилин №3
Настраиваем хранение логов в базе данных MySQL Сергей Крутилин №5
Настройка Webacula. Веб-интерфейс к Bacula Сергей Яремчук №12
Новые методы защиты и управления информацией Сергей Соломатин №6
Обзор Open Source ECM-системы Alfresco Сергей Яремчук №3
Обзор бесплатных почтовых серверов для операционных систем семейства Windows Алексей Бережной №2
Обзор операционной системы gNewSense GNU/Linux 2.2 Deltah Игорь Штомпель №5
Обзор проекта Gnash Игорь Штомпель №6
Оборудование Cisco для «самых маленьких» Сергей Крутилин №4
Один UPS на двоих. FreeBSD в домене Windows Рашид Ачилов №12
Оптимизируем PPD-файлы Иван Коробко №1
Организуем систему резервного копирования для малого и среднего офиса Алексей Бережной №6
Осваиваем нововведения языка сценариев Windows PowerShell 2.0 Василий Гусев №7
Основные изменения в WAIK для Windows Server 2008 R2/7 Сергей Яремчук №4
Офисная АТС Samsung: подключаем GSM-шлюз Рашид Ачилов №2
Перенос профиля пользователя в Windows XP Professional Edition и Windows 2000 Professional Рамиль Айзятуллен №1
После катастрофы. Теория и практика восстановления Exchnage Server Михаил Даньшин №12
Построение каталога сервисов Александр Башкиров №6
Почтовый клиент Alpine Игорь Штомпель №4
Проводим реализацию тонкого делегирования прав в Active Directory Вадим Андросов №7
Продукты для виртуализации. Возможности свободного программного обеспечения Игорь Штомпель №11
Расширение возможностей при работе с сетевыми хранилищами NETGEAR ReadyNAS Алексей Бережной №12
Резервирование и восстановление объектов Active Directory в Windows Server 2008/2008R2 Сергей Яремчук №6
Решаем проблему внезапной блокировки учетной записи Михаил Даньшин №11
Решение давно наболевшей проблемы Денис Староверов №6
Сдаем бухгалтерскую отчётность в электронном виде Максим Лобов №1
Сервер отчётов для малого бизнеса Александр Федяшов №2
Сетевая версия SQL под Linux. Перевод серверной части «1С:Предприятие 7.7» Александр Гернгросс, Максим Лобов №11
Синхронизируем данные между компьютерами с помощью сервисов сетевого хранения Виталий Банковский №4
Сисадмин должен быть ленив. DHCP и динамический DNS Сергей Супрунов №8
Система видеоконференций OpenMeetings Сергей Яремчук №6
Собираем свой дистрибутив с Calculate Linux Scratch Сергей Яремчук №12
Создаём шлюз с системой учёта трафика на слабом компьютере Николай Емашев №3
Строим сеть на Calculate Directory Server Сергей Яремчук №8
Такие разные пароли Иван Квасников №3
Терминальные службы. Установка, печать и балансировка нагрузки Станислав Шпак №9
Технология Oracle Streams. Настраиваем потоки данных, экономим время и деньги Антон Пищулин №12
Тратим меньше, спим больше с VMware Sphere 4.0. Часть 2 Виталий Банковский №12
Тратим меньше, спим больше с VMware Sphere 4.0: установка Виталий Банковский №7
Удаленный аппаратный доступ к серверам Игорь Калинин №8
Удобно, безопасно, недорого. Управление мобильными устройствами на предприятии Алексей Ватутин №12
Управление корзиной. Новый сервис в Active Directory Иван Коробко №8
Управляем объектами в Active Directory. Часть 4 Иван Коробко №1
Устанавливаем Windows XP с помощью System Center Configuration Manager 2007 R2 Алексей Тараненко №5
Устанавливаем и настраиваем службу управления ключами в Windows Server 2008/Windows Vista Иван Квасников №3
Устанавливаем связку Squid + squidGuard + c-icap Олег Палухин №3
Учет компьютеров с Hardware Inspector Сергей Унагаев №11
Учет оборудования с OCS Inventory NG и GLPI Сергей Яремчук №5
Черный экран тишины. Десять способов, как избавиться от него Никита Панов №11

Безопасность №

«Доктор Веб» дал старт первому российскому антивирусу для Mac 0S X Валерий Ледовской №6
Cisco IDS/IPS. Безопасная настройка Андрей Дугин №8
Dr.Web CureNet!. От идеи до воплощения Павел Плотников №12
Dr.Web для Mac OS X. Как рождался интерфейс Павел Плотников №10
iQ.Suite – экономичное решение безопасности для Lotus Domino и Microsoft Exchange Алексей Демин, Ирина Абалихина №2
OSSEC-HIDS. Установка и настройка Сергей Яремчук №10
Shorewall: Iptables с человеческим лицом Валентин Синицын №2
Безопасная работа с сессиями в PHP Антон Гришан №2
Внедряем смарт-карты в домене Станислав Шпак №2
Выбираем антивирус для небольшой сети Сергей Яремчук №4
Защита данных с помощью Active Directory Rights Management Services Андрей Бирюков №6

Содержание журнала за 2009 год

Как спасти пирожки? AppLocker: укрепляем безопасность сети Вадим Поданс №11
Настройка интернет-шлюза с авторизацией через AD по протоколу Kerberos Дмитрий Нестеркин №11
Новые возможности Nmap 5.00 – программы для исследования безопасности сетей Игорь Штомпель №9
Практический инсайд Олег Кузьмин №10
Средства обеспечения безопасности в Windows Vista. Часть 1 Андрей Бирюков №2
Средства обеспечения безопасности в windows Vista. Часть 2 Андрей Бирюков №3
Теневое копирование в Windows Server 2003 Александр Емельянов №2
Тотальная защита локальных сетей Вячеслав Медведев №4
Управляем доступом к ресурсам домена на основе Windows Server Вадим Андросов №9
Шифруемся на лету при помощи TrueCrypt Александр Емельянов №7

Сети №

WebVPN на базе Cisco IOS Иван Панин №7
Гость из будущего. Протокол IPv6 – новая версия Илья Рудь №11
Корпоративные VPN на базе Cisco Иван Панин №6
Обзор технологии Geneva. Построение распределенных гетерогенных систем Андрей Бирюков №11
Эффективный инструмент для создания единой корпоративной сети Иван Панин №8

Веб №

Native Client – Rich Internet Applications от Google Кирилл Сухов №2
Open Web Tools Directory Игорь Штомпель №10
Как работает suexec Андрей Шетухин №7
Портал в стиле Web 2.0 Александр Башкиров №5
Создаем свой YouTube. Как обработать видео средствами веб-сервера Александр Майоров №12
Ускоряем загрузку сайта, минимизируя количество HTTP-запросов Антон Гришан №7

Программирование №

DBMS_SHEDULER. Запуск последовательных и зависимых заданий Антон Пищулин №10
IPython как инструмент системного администратора Сергей Супрунов №3
Python: сложные аспекты Дмитрий Васильев №5
Знакомьтесь, Erlang! Основы языка программирования Дмитрий Васильев №8
Компонентная модель EJB. Преимущества и недостатки Арсен Ибрагимов №12
Обманчивая простота языка BF: генезис или мутация? Алексей Вторников №2
Основные грани Ruby. От главных конструкций до приложения Дмитрий Васильев №10
Основы Spring Андрей Уваров №9
Пишем первые модули на Erlang Дмитрий Васильев №9
С новым Python-3000! Сергей Супрунов №2
Язык Python. Дополнительные типы коллекций Сергей Супрунов №10

Веб-программирование №

JavaFX – Reach Internet Application от Sun. Прощай, унылый Swing? Кирилл Сухов №4
JQuery: магия JavaScript Александр Слесарев №1
WIMP – Windows, IIS, MySQL, PHP Кирилл Сухов №3
WinBinder PHP. Создаём GUI-интерфейс за 2 клика Александр Майоров №1
Автоматическая загрузка объектов в PHP Антон Гришан №3
Доступ к данным на основе хранимых процедур в веб-приложениях Антон Гришан №1
Почему стоит использовать механизм обработки исключений в PHP Антон Гришан №2
Приёмы минификации в веб-приложениях Антон Гришан №4

Администрирование «1С» №

«1С:Предприятие 8». Управляемое приложение. Бета-версия Альберт Балаков №3
Используем универсальные отчеты и обработки в «1С:Предприятии 8» Альберт Балаков №1
Обновление конфигурации Андрей Луконькин №4
Очередное собрание ошибок Андрей Луконькин №6
Работаем с FTP-сервером из «1С» Андрей Луконькин №5
Решаем проблемы с установкой и удалением драйвера защиты Андрей Луконькин №2
Управление базами данных «1С» 7.7 при помощи групповых политик Владимир Борисов №5

Изучаем «1С» №

Библиотека стандартных подсистем. Обсуждаем плюсы и минусы новшества Андрей Луконькин №10
Углубляемся в код управляемого приложения Андрей Луконькин №8
Управляемое приложение. Первые осторожные шаги Андрей Луконькин №7
Электронная проходная своими руками Андрей Луконькин №9

Кафедра №

Вызов XXI века: каким быть программному обеспечению? Игорь Штомпель №12
Вычислительная модель. Введение в системное программирование Алексей Барабанов №12

IT-управление №

Ищем сотрудников или партнеров? Артем Черневский №9
Расчет на салфетке. Как продать IT-проект с помощью математики Дмитрий Бутянов №8
Управляем эффективностью компании. Акценты сместились, приоритеты уточнились Виталий Верещагин №11

Человек номера №

Жизнь в стиле хокку Оксана Родионова №4
Кликнем мышкой Оксана Родионова №3
Приключения продолжаются... Оксана Родионова №5
Романтик Оксана Родионова №6

Гость номера №

В чем секрет популярности Linux User Group из Пекина? Антон Борисов №5

Содержание журнала за 2009 год

Вячеслав Калошин: «Всегда ищи точки приложения своих сил!» Агунда Алборова №10
Михаил Кристев: «Кризис – не время для уныния» Алексей Алексеев №8
Оксана Глущенко: «ИТ-ландшафты: бизнес-эволюция или бизнес-революция?» №7
Павел Бетсис: «Отказаться от инноваций – значит перестать развиваться» Галина Положевец №11
Свободный полет Оксана Родионова №9

Человек-легенда №
Понять Билла Михаил Гаков №7
Сетевой книгоноша Владимир Гаков №9
Харизматик-искуситель Владимир Гаков №8

Закон есть закон №
До часа «Х» – меньше 100 дней. Что делать с персональными данными? №10
Перегретая тема. И снова о персональных данных №12
Сложно, но можно. Как защитить персональные данные на предприятии? Юлия Штокало №9
Юридический статус виртуальных денег на территории России Юлия Штокало №2

Острый угол №
SOA. Взгляд шире! Это не просто набор стандартов и технологий Эдуард Долгалев №10
SOA: плюсы и минусы. Готовы ли компании к SOA-архитектуре? CNews Analytics /CNA/ №10
Аутсорсинг в вашей компании. Это хорошо или плохо? №11
Взломщик подобен художнику. По уровню защиты мы заметно отстаем от Запада Андрей Соколов №8
Вы хакеров уважаете? №8
Доверяй, но проверяй. Как защититься от мнимых друзей №11
Как вы стали системным администратором? №9
Как на вас и на вашей компании сказался кризис? №7
Киберпреступность – это бизнес. Масштабы бедствий превышают миллионы долларов Алексей Андрияшин №8
Куда приводят мечты? В сети Оксана Родионова №9
Они – не Робин Гуды. Хакеров можно уважать за ум, но не за поступки Михаил Калиниченко №8
Пять аргументов «за». Антикризисные задачи ИТ-аутсорсинга Евгений Щепилов №11
Сисадмины и хакеры: свои среди чужих, чужие среди своих Алексей Ремизов №8
Скупой платит дважды. Защитив всего 99,9% своих ресурсов, компания рискует быть атакованной Кирилл Керценбаум №8
Цари природы, энергобатарейки или биокомпьютеры Андрей Погодин №7
Это не панацея от бед! О чем забывают при создании SOA-систем Владимир Энгельс №10

События №
Все под контролем. Windows Server 2008 R2 экономит и деньги, и время Полина Гвоздь №12
Праздник, который придумал Тед Илья Александров №7
Слет сисадминов. И вновь продолжается бой №8
Углубляем знания с Cisco Expo Learning Club Андрей Бирюков №7
Форум Cisco Expo-2009 переписал собственные рекорды №12
Юбилейная конференция Cisco Expo. Главное ИТ-событие года в России №10

Репортаж №
«Айдеко» представила интернет-шлюз Itleco ICS 3.0 Дмитрий Шурупов №6
Sun Tech Days 2009: кризис не помеха Кирилл Сухов №5
В Москве прошел пятый Форум по открытому коду Дмитрий Шурупов №6
В Москве прошла конференция системных администраторов RootConf 2009 Дмитрий Шурупов №4
Виртуализация от Citrix – новый взгляд на привычные вещи Алексей Бережной №4
Новые беспроводные решения от Cisco Андрей Бирюков №6
Платформа 2009. Определяя будущее Андрей Бирюков №1
Слон против дельфина Кирилл Сухов №2

Из личного опыта №
Лабораторная работа: исследование уязвимостей с помощью Metasploit Framework Павел Троицкий №1

Семь лет с нами №
Алексей Барабанов: «Кто «хозяин» компьютеров? Сисадмин!» №10
Павел Александров: «У «Системного администратора» нет альтернативы» №9
Павел Закляков: «Нужно, чтобы после прочтения человек бежал к компьютеру» Оксана Родионова №7
Сергей Анциферов: «Ваша изюминка – «болты и гайки» №11
Сергей Яремчук: «Сисадмин» – это мой формат» №8

Ретроспектива №
Компьютер, опередивший время Дмитрий Мороз №2
Компьютерный парк юрского периода. Его прообраз задумал испанский монах в XIV веке Владимир Гаков №9
Лидер всегда щедр. За что конкуренты любили Sun? Илья Александров №10
Мал, да удал: мини-комьютер PDP-8 Алексей Вторников №4
Персональное дело IBM. Вся правда об империи «Голубого гиганта» Владимир Гаков №11
Тотальная мобилизация. Третья попытка Пола Галвина Владимир Гаков №12
Шедевр рекламного искусства Дмитрий Мороз №3

Творчество админа №
POWER OFF Станислав Шпак №12
Игра Станислав Шпак №8
История одного знакомства Станислав Шпак №7
Контейнер Станислав Шпак №4
Последние минуты Станислав Шпак №5
Пункт 4.11 Станислав Шпак №3

Редакционнная подписка
для физических лиц

> Вы можете оформить подписку только на **российский адрес.**
> При заполнении квитанции обязательно **разборчиво укажите фамилию, имя, отчество полностью, почтовый индекс и адрес получателя (область, город, улица, номер дома, номер квартиры), контактный телефон.**
> Журнал высылается почтой заказной бандеролью только после поступления денег на расчетный счет и **копия заполненного и оплаченного бланка, отправленная в редакцию по факсу: (495) 628-8253, (доб. 120) или на emil: subscribe@samag.ru**

ИЗВЕЩЕНИЕ

ООО "С 13" Форма № ПД-4
ИНН 7708654814 / КПП 770801001
Р. сч. 40702810300080001868 К. сч. 30101810100000000787
ОАО «УРАЛСИБ» г. Москва БИК 044525787
Коды: по ОКПО 84027582, по ОКОПФ 65

Вид платежа: Редакционная подписка на журнал "Системный администратор" за 2010 г.

01	02	03	04	05	06	07	08	09	10	11	12
X	X	X	X	X	X	X	X	X	X	X	X

Дата _____ Сумма платежа: 2400 руб. 00 коп.

Информация о плательщике:

_____
(Ф. И. О. почтовый индекс, адрес и телефон)

_____

_____

Подпись _____

Кассир

КВИТАНЦИЯ

ООО "С 13" Форма № ПД-4
ИНН 7708654814 / КПП 770801001
Р. сч. 40702810300080001868 К. сч. 30101810100000000787
ОАО «УРАЛСИБ» г. Москва БИК 044525787
Коды: по ОКПО 84027582, по ОКОПФ 65

Вид платежа: Редакционная подписка на журнал "Системный администратор" за 2010 г.

01	02	03	04	05	06	07	08	09	10	11	12
X	X	X	X	X	X	X	X	X	X	X	X

Дата _____ Сумма платежа: 2400 руб. 00 коп.

Информация о плательщике:

_____
(Ф. И. О. почтовый индекс, адрес и телефон)

_____

_____

Подпись _____

Кассир

Подписные индексы:

20780*
+ диск с архивом статей 2009 года

81655**
без диска

по каталогу агентства «Роспечать»

88099*
+ диск с архивом статей 2009 года

87836**
без диска

по каталогу агентства «Пресса России»

* Годовой
** Полугодовой
*** Диск вкладывается в февральский номер журнала, распространяется только на территории России